Multimedia Tools and Applications
https://doi.org/10.1007/s11042-019-08350-1

Image representation using complete multi-texton histogram

Belal Khaldi[1] · Oussama Aiadi[1] · Kherfi Mohammed Lamine[2]

Received: 24 February 2019 / Revised: 7 August 2019 / Accepted: 9 October 2019 /
[redacted]

Abstract
Texture is a fundamental aspect which is often used to represent the visual content of images. Psychologists stated that color and texture have a close relationship via fundamental micro-structures called textons. Textons are considered as atoms for pre-attentive human visual perception. Based on texton theory, many literature works have tried to develop efficient models for image recognition. In this paper, we put forward a texton-based method, called Complete Multi-Texton Histogram (CMTH), that has the ability to discriminate both texture and non-texture color images. CMTH incorporates information about the color, edge orientation and texton distribution within the image. The proposed CMTH has been extensively examined on five publicly available datasets. Three of these datasets were intended to evaluate texture discrimination, namely: Vistex, Outex, and Batik whereas the two others were intended to evaluate heterogeneous image discrimination, namely: Corel10K and UKBench. The proposed method has been evaluated via image classification and retrieval tasks. The obtained results have shown that our proposed descriptor significantly outperforms the state of the art methods in both classification and retrieval.

Keywords Texton theory · Texture perception · Image representation · Features extraction

1 Introduction

Image features are numerical measurements (e.g., values, vectors, matrices, etc.) extracted from an image to describe its visual content. These features could be roughly categorized into three main categories which are color [34], texture [14] and shape [60]. Color histogram [49] is a very simple, yet, an effective and widely-used feature. However, it surfers from

✉ Belal Khaldi
khaldi.belal@univ-ouargla.dz

[1] Dept. Computer Science and Information Technologies, University of Kasdi Merbah Ouargla (UKMO), Ghardaia Road, BP.511, 30 000, Ouargla, Algeria

[2] Laboratoire de recherche en Mathématiques et Informatique Appliquées (LAMIA), 3351, Boulevard des Forges, C.P. 500, Trois-Rivières, G9A 5H7, Canada

the lack of information about the spatial distribution of colors. On the other hand, texture features provide important information about the primitives that constitute a texture and their spatial distribution.

Texture is a very useful cue that can be used to discriminate images. In literature, several approaches have been proposed to represent texture, namely: statistical, frequency-based and structural approaches. Statistical features include co-occurrence matrices [13, 20, 23–25, 54], Weber Local Descriptor (WLD) and its derivations [5, 6], Local Binary Pattern (LBP) and its derivations [3, 35, 44, 47], autocorrelation-based [26, 53, 59] and registration-based [33, 56] features. Structural features include primitive measurement-based [27, 57], edge-based [40, 62], skeleton-based [38, 48] and morphological operation-based [9, 12, 50] features. Frequency-based features include spatial domain filtering [1, 7, 37] , frequency domain analysis [32] and joint spatial-frequency methods [39, 58]. Texture features have widely been exploited in variety of applications of image [2, 28] and video analysis [51, 52, 61].

Deep learning, precisely Convolutional Neural Networks (CNN), has emerged as an alternative paradigm of handcrafted image-features tuning. Layer outputs in such networks can be considered as features that are generic and, to some extent, independent of the classification/retrieval task. However, CNN strongly depends on the training dataset and suffers from the problem of high computation and data storage. Additionally, a considerable human effort is required in the selection of images suitable for building the training set which makes it far from ideal [42]. According to Bu et al. [4], texture features extracted from the fully connected (FC) layer of a CNN are not suitable for texture representation. This issue has been attributed to the regularity of texture patterns that can't be represented using an orderless approach(i.e., FC layer).

About 37 years ago, Julesz [15] introduced the concept of texton. After a series of hard works trying to understand how the human vision discriminates textures, Julesz concluded: "The many indistinguishable texture pairs having identical second-, but different third- and higher-order statistics, led to the conjecture that globally the preattentive texture discrimination system cannot process statistical parameters of third- or higher-order. Thus in cases when iso-second-order textures yield discrimination this must be based on local conspicuous features called textons". Textons refer to fundamental micro-structures in images (and videos), which are considered as the atoms of pre-attentive human visual perception [15]. Textons come in the form of blobs attributed with color, length and orientation (i.e. lines, elongated blobs, and dots) [15–17]. More details and mathematical models of textons could be found in [63].

Texton theory has been successfully applied in various fields including image retrieval [2, 28, 29, 31], classification [11, 21, 55], and others [36, 45]. Moreover, many researchers have tried to figure out a way to improve texture representation using textons. In this paper, we propose a feature called Complete Multi-Texton Histogram(CMTH). This feature integrates both the pros of two recent works, the one presented in [21] and the other one in [31]. CMTH provides: (a) a complete representation that takes into account all possible types of textons, (b) rich information about colors, textures and edges. To prove its effectiveness, the proposed feature has been evaluated against numerous other state of the art features.

This paper is organized as follows: in Section 2, we introduce and discuss works that are highly related to ours. Section 3 presents our proposed improved feature CMTH. In Section 4 we conduct evaluations and discuss the results. Finally, we draw some conclusions at the end of this paper.

2 Related work

This section is devoted for reviewing works that are most relevant to ours. In fact, we can list four highly related works which are Texton Co-occurrence Matrix (TCM) [29], Multi-Texton Histogram(MTH) [31], Complete Texton Matrix(CTM) [21] and Noise Resistant Fundamental Units of Complete Texton Matrix(NRFUCTM) [22].

2.1 Texton co-occurrence matrix(TCM)

TCM [29] has originally been proposed to measure the spatial correlation between textons in a given texture image. It codifies the information about the spatial correlation between textons in the form of a co-occurrence matrix. In this paper, the authors were inspired by the work of Haralick on Gray-Level Co-occurrence Matrix(GLCM) [13]. However, instead of calculating the appearance frequency of different gray levels, TCM considers the appearance frequency of different edges and textons.

For a given color image *I*, the main principle of extracting the TCM from this image can be summarized in the following two phases.

2.1.1 Extracting edge orientation matrix

Texture could be recognized, by the human eye, as repetitive edges. Therefore, the authors suggested integrating information about edge orientations and gradients in their proposed feature. For each pixel *p* located at the coordinates *(x,y)* within the image, three measures are separately calculated for each channel of the RGB color space: edge horizontal orientation g_{xx}, edge vertical orientation g_{yy} and edge combined orientation g_{xy}. Such orientations can simply be calculated using Sobel operator.

In order to calculate the gradient of a given pixel, orientation and orientation magnitude are needed. For a given pixel *p(x,y)* (1) is used to extract orientation, whereas, (2) and (3) respectively are used to calculate the magnitude on one direction and its perpendicular direction.

$$\theta\left(x,y\right)=\frac{1}{2}arcttan\left(\frac{2g_{xy}}{g_{xx}-g_{yy}}\right) \tag{1}$$

$$G_1\left(x,y\right)=\sqrt{\frac{1}{2}\left[\left(g_{xx}+g_{yy}\right)+\left(g_{xx}-g_{yy}\right).F\left(x,y\right)\right]} \tag{2}$$

$$G_2\left(x,y\right)=\sqrt{\frac{1}{2}\left[\left(g_{xx}+g_{yy}\right)+\left(g_{xx}-g_{yy}\right).F'\left(x,y\right)\right]} \tag{3}$$

such that

$$F\left(x,y\right)=cos\left(\theta\left(x,y\right)\right)+2g_{xy}sin\left(2\theta\left(x,y\right)\right) \tag{4}$$

$$F'\left(x,y\right)=cos\left(\theta\left(x,y\right)+\frac{\pi}{2}\right)+2g_{xy}sin\left(2\theta\left(x,y\right)+\frac{\pi}{2}\right) \tag{5}$$

Using the results obtained by the previous process, two new maps will be generated, the first one holds the maxima of orientation magnitudes (i.e., $Max(G_1, G_2)$) and the other one holds the minima of the orientation magnitudes(i.e., $Min(G_1, G_2)$). Finally, a co-occurrence matrix is extracted from each produced map.

2.1.2 Extracting texton matrix

This phase could be summarized in the following steps.

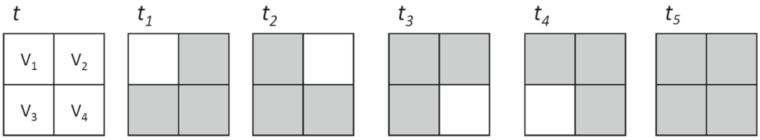


Fig. 1 Five special types of textons: t 2x2 grid; the possible textons are: t_1; t_2; t_3; t_4 and t_5 [29]

Step 1. The colors of the image *I* are quantified into 256 bins, which produces a new image *C* with a single color channel.
Step 2. For each texton t_i from the five predefined templates of textons $\{t_1,t_2,t_3,t_4,t_5\}$ (Shown in Fig. 1).
Step 3. pass the texton t_i (2x2 grid), from left-to-right and from top-to-bottom, over the image looking for matching.
Step 4. If t_i matches some 2x2 grid in the image, then this grid remains the same. Otherwise, all values of the grid will be reset to 0.
Step 5. For the next texton from $\{t_{1_i},t_2,t_3,t_4,t_5\}$, go back to *Step 3*.
Step 6. Combine the five produced maps (i.e., one from each texton) to get a final image and extract the texton co-occurrence matrix from it.

After having the three co-occurrence matrices extracted (i.e., maximum orientation, minimum orientation and texton co-occurrence matrices), a set of statistical features are drawn from them, namely: energy, homogeneity, contrast and entropy. Therefore, each image is represented with a TCM feature vector of length 12.

2.2 Multi-texton histogram(MTH)

Two years after their original work on TCM [29], Lie et al. have proposed an improved version which is called Multi-Texton Histogram(MTH) [31]. In contrast to TCM, which was devoted to represent texture, MTH has been proposed to represent heterogeneous images. It has the advantages of both co-occurrence matrix and color histogram by holding information about the spatial correlation of colors and texture orientations.

Similarly to TCM, MTH passes a set of textons (illustrated in Fig. 2), one at a time, on a given image *M* then combines the produced maps in order to generate a final texton image *T*.

The authors have argued that texton map doesn't seem to be enough. They, therefore, have produced another map that represents texture orientations. According to [29], information about texture orientations provide a semantic description of the image, which makes it a

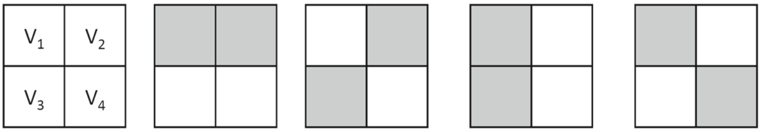


Fig. 2 Four special types of textons [31]

very useful cue. For gray-level images, a texture-orientation map could be simply produced using (6)

$$\theta = \arctan \frac{g_y}{g_x} \tag{6}$$

such that g_x and g_y are two maps produced by applying the Sobel operator along the horizontal and vertical directions, respectively.

Regarding color images, applying Sobel operator after converting them into gray-level space isn't efficient. This is because a great amount of information will be lost as a result of converting the image from color to gray-level space. Lie et al. suggested an alternative method to extract orientations from a color image using (7).

$$\theta = \arccos \frac{a.b}{|a|.|b|} \tag{7}$$

where $a(R_x, G_x, B_x)$ and $b(R_y, G_y, B_y)$ are two 3-Dimensional gradient maps for x and y directions, respectively. As instance, R_x is the gradient map obtained from the Red channel of the RGB color space. The norms $|a|$ and $|b|$, and the dot product $a.b$ are calculated using (8), (9) and (10), respectively.

$$|a| = \sqrt{R_x^2 + G_x^2 + B_x^2} \tag{8}$$

$$|b| = \sqrt{R_y^2 + G_y^2 + B_y^2} \tag{9}$$

$$a.b = R_x.R_y + G_x.G_y + B_x.B_y \tag{10}$$

After having the orientation image produced, it has been quantified uniformly into 18 orientations.

For a given image I, let the RGB color space be quantified into 64 bins which yields a new image T. Let P_1 (x_1, y_1) and P_2 (x_2, y_2) be two neighboring pixels in the image T separated by a given distance D $(\triangle x, \triangle y)$. T (P_1) and T (P_2) are two neighboring colors in the image T. θ (P_1) and θ (P_2) are two neighboring orientations in the orientation map θ. Let's denote the number of co-occurring orientations and textons, respectively, by N and $\overline{N}$. Then, the MTH can be calculated using (11) and (12).

$$MTH\,(T\,(P_1)) = N\{\theta\,(P_1) = \theta\,(P_2) = v\} \tag{11}$$

$$MTH\,(\theta\,(P_1)) = \overline{N}\{T\,(P_1) = T\,(P_2) = w\} \tag{12}$$

Since the color space was quantified into 64 bins and the orientations into 18, the length of the original MTH is 82.

A recent study by Raza et al. [46] opted for an identical approach with a slight modefication for image retrieval. Maulani et al. [41] have conducted an experimental study for classifying Indonesian Batik using MTH. Satisfying results have been reported in this study.

2.3 Complete texton matrix (CTM)

Based on the two former works, Kumari et al. [21] have proposed another alternative for representing heterogeneous images using textons. They tried to integrate more information in their representation by considering more types of textons. Instead of four or five textons as in TCM and MTH, the CTM is extracted using 11 textons. However, Kumari et al. have restricted themselves to textons only. In other words, the CTM does not include any information about edge orientations or gradients.

The main principle of the CTM is very simple. The image is firstly converted into gray-level space then quantified into 8 bins. As a second step, the image is divided into 2 × 2 non-overlapping grids then matched with each texton separately. The result of this matching process is 11 new texton maps. Using these texton maps, they produced a 2D (i.e., 8 × 11) matrix CTM, such that $CTM\,(g, T)$ represents the appearance frequency of a texton T having the gray level g. This matrix is used later-on as an image descriptor where it showed a high performance compared to other features.

2.4 Noise resistant fundamental units of complete texton matrix (NRFUCTM)

In their recent work, kumari et al. [22] have proposed reducing the noise from the image before extracting the *CTM*. They firstly transform the original image into Local Binary Code (LBP) map. A floating window is a 3 × 3 pixels neighborhood where the absolute gray level intensity of the central pixel Z_c and a neighboring pixel Z_i falls within the range of a threshold t. A non-Uniform LBP (NULBP) floating window may be transformed into ULBP, by complementing one or c number of floating pixels starting from the least significant bit (LSB) . This process produces a Noise Resistant Fundamental Texture (NRFT). Figure 3 shows how a ULBP could be extracted from a floating window.

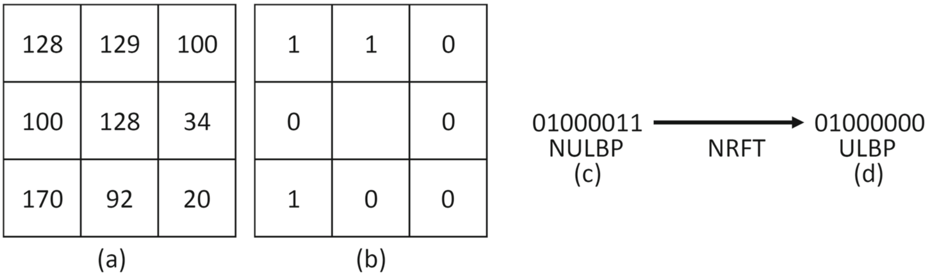
After having the NRFT image derived, they extract a CTM from it (the result is NRFUCTM map). Then, GLCM [13] features(i.e., Homogeneity, Energy, Contrast and Correlation) have been extracted with three distance factors (d=2, 3 and 4) and four orientations ($0^o, 45^o, 90^o and 135^o$).

In the next section, we will put forward a new feature in which we integrate the pros of the former works.

3 Complete multi-texton histogram

After a detailed analysis of TCM, MTH, CTM and NRFUCTM features, we noticed that, in spite of the high performance of those features, they suffer from some serious drawbacks. TCM on one hand suffers from the following:

- It has been proposed to be used with texture only, which makes it unsuitable to represent heterogeneous images.
- Reducing the TCM co-occurrence matrix into third-order moments (i.e. energy, correlation, contrast and homogeneity) causes the loss of many useful information [31].



128	129	100
100	128	34
170	92	20

(a)

1	1	0
0		0
1	0	0

(b)

Fig. 3 A simple NRFT scheme that transforms NULBP into ULBP where **a** is a floating window, **b** is the LBP pattern, **c** is the NULBP binary code and **d** is a ULBP obtained by complementing the two least significant bits of the NULBP binary code

MTH, on the other hand, has tried to resolve some of TCM's issues. However:

- In MTH, two neighboring pixels are supposed to be in the same orientation if they have the same color, and vice versa. Nevertheless, such an assumption doesn't always hold.
- It opts for a poor variation of textons. This leads other types of textons that are present in image to be ignored.

CTM and NRFUCTM have considered using a full variation of 2 × 2 grid textons. Yet, they have integrated another weakness. CTM and NRFUCTM don't include neither gradient, nor edge orientation information. They are restricted only to texton co-occurrence matrix. Such a restriction could yield a weak representation.

In this section, we propose a general representation inspired by the previous features. We combine the information richness provided by TCM and MTH with the completeness provided by CTM and NRFUCTM. As a result of such a combination, a Complete Multi-Texton Histogram(CMTH) is obtained. We use 11 types of textons to produce a histogram that contains information about the spatial distribution of textons, colors and edge orientations. CMTH is:

1. More general than TCM because it can be used with either heterogeneous or texture images.
2. More complete than MTH because it employs 11 types of textons instead of 4.
3. In contrast to MTH, CMTH consider all possible variations of colors/orientations.
4. Richer in information than CTM and NRFUCTM because it integrates information about edge gradients and orientations.

The procedure of extracting the CMTH from a color image (in RGB space) is schematized in Fig. 4 shows.

The process of extracting CMTH, from a given image *I*, is summarized in the following steps:

Step 1 Detect the edge orientation for each pixel in *I* using (6). We opted for the gradient operator proposed in [31], rather than *Sobel, Robert, LoG, etc.*, because it has shown a better performance. This step produces an orientation map quantified into 9 bins.

Step 2 Quantify the three channels of RGB color space into one channel $C = 9R+3G+B$ (i.e., 27 bins), and then, sub-divide the qunatized image P into non-overlapping 2×2 pixel grids.

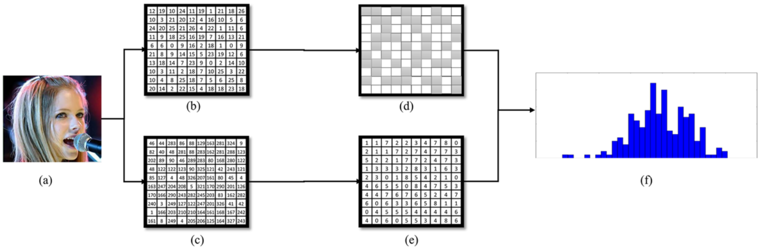


Fig. 4 A general scheme for *CMTH* extraction. **a** is a given image *I*, **b** is the result of qunatizing *I* into 27 bins, **c** is the edge orientation map **d** is the result of passing all types of textons on the quantified image **e** is the quantified orientation map and **f** is the final CMTH histogram

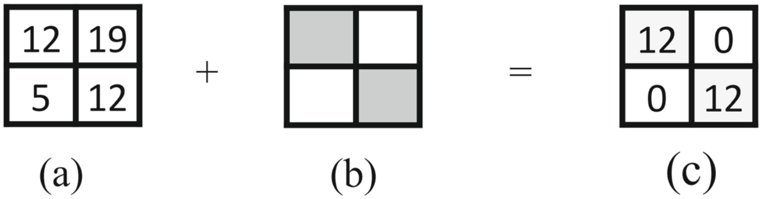


Fig. 5 Passing textons on image grid, **a** is a 2 × 2 grid, **b** is a given texton and **c** is the result of passing the texton on the grid

Step 3 Pass each texton (illustrated in Fig. 6) on the grid image, if it matches, then the matched grid pixels remain the same; otherwise, reset the pixels to 0. Fig. 5 explains more this step;

Step 4 Combine the obtained 11 texton maps to produce a complete texton map.

Step 5 Extract three co-occurrence matrices from the complete texton map. The first one represents how neighboring pixels have the same color with different orientations (i.e., corners) co-occur (Given by (13)). The second one represents how neighboring pixels having the same orientation with different colors (i.e., edges) co-occur (Given by (14))). And the third one represents how textons co-occur (Given by (15))).

$$MC\left(V\left(P_1\right), V\left(P_2\right)\right) = N\{\theta\left(P_1\right) = \theta\left(P_2\right)\} \tag{13}$$

$$MO\left(\theta\left(P_1\right), \theta\left(P_2\right)\right) = N\{V\left(P_1\right) = V\left(P_2\right)\} \tag{14}$$

$$MT\left(T_1, T_2\right) = N\{T = T_i \wedge T_N = T_2\} \tag{15}$$

where, $V\left(P_i\right)$ is the color value of the pixel P_i, $\theta\left(P_i\right)$ is the orientation of the pixel P_i, P_1 and P_2 are two neighboring pixels. T_i and T_N are two neighboring textons within the map.

Step 6 Finally, reshape each matrix to a vector, then concatenate them to produce a final texton histogram *CMTH*.

Our proposed method analyzes the correlation between neighboring textons, color, and edge orientations using 11 texton types. This color/orientation correlation is represented

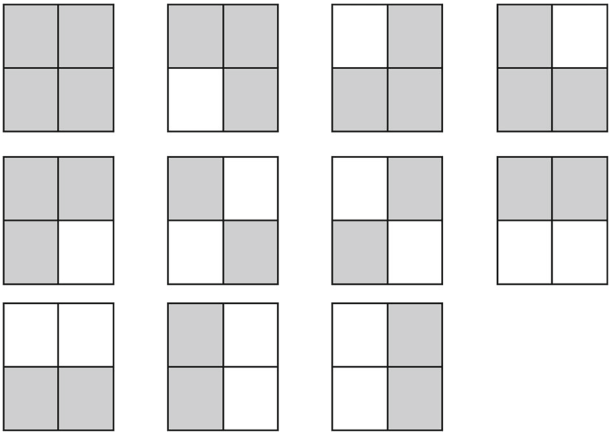


Fig. 6 Eleven special types of textons used to extract CMTH

in a form of a histogram H. The values have been quantified to reduce the size of the feature vector. In the next section, we conduct a detailed experimental evaluation comparing TCM, MTH, CTM, NRFUCTM and our CMTH in classifying/retrieving various texture and heterogeneous images. Furthermore, our descriptor has been evaluated against other well-known descriptors.

4 Experimental evaluation

We have evaluated our *CMTH* for image classification and image retrieval tasks, however, it can be used in different other applications of image processing and pattern recognition. It is worth mentioning that the investigated texton-based methods and the evaluation have been implemented using Matlab 2017b. In this section, we start by explaining the experimental setup, and then conduct experimental evaluations and discuss the results.

4.1 Image classification

To confirm the classification effectiveness of our proposed method, we have opted for four datasets. On one hand, Outex and Vistex datasets are dedicated to evaluate the texture discrimination capability of our method. On the other hand, Corel and UKbench datasets are dedicated to evaluate the proposed method in discriminating heterogeneous images.

As we have stated before, Minarno et al. [41] have used MTH in order to build a Batik classification system where they argue that the results were quite satisfactory. Thus, we have conducted a comparison, using the same Batik dataset and the same configuration, between our proposed CMTH and the baseline results they provided.

Here is a brief description of the used datasets.

- Outext dataset [20]: consists of 11484 images categorized into 29 classes (e.g., barley rice, canvas, chips, and so on).
- Vistex dataset: consists of 830 images categorized, unequally, into 29 classes (e.g., Bark, Brick, Grass, and so on).
- Corel10K [30] dataset: consists of 10000 images categorized into 100 classes (e.g., City, Mountain, Sky, and so on).
- UKbench [43] dataset: consists of 10192 images categorized into a very high number of classes (2548 classes).
- Batik [41] dataset: the dataset consists of 300 images categorized into 50 classes (i.e., 6 images per class).

Representative samples from each dataset are illustrated in Fig. 7.

In order to prevent data overfitting during the training process, we opted for cross validation. K-fold Cross-validation (k=5), with different classifiers, has been used to evaluate the performance of the different features using multiple datasets. To accelerate calculation, feature sizes have been reduced into 12 using Principal Component Analysis(PCA).

Table 1 gives the obtained accuracy/standard deviation by applying KNN classifier, with different values of k, on all datasets.

Table 1 clearly shows that our proposed method remarkably outperforms the others in all datasets and for all values of k. For $k=1$, features like TCM and CTM, which are designed to be more efficient in representing textures, show high performances in classifying Outex and Vistex (i.e., TCM yields 72% with Outex and CTM yields 69% with Vistex). However, these two features show a poor performance in classifying heterogeneous images (i.e., TCM

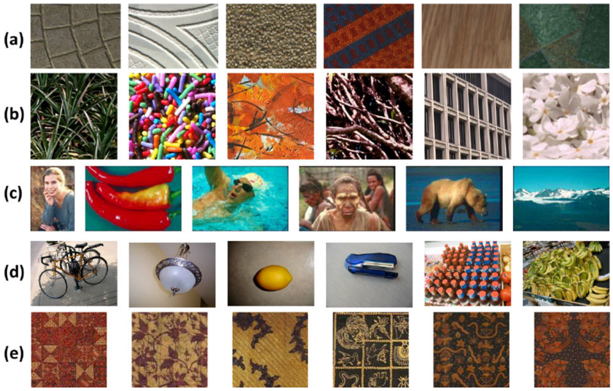


Fig. 7 Representative samples from each dataset. **a** Outex, **b** Vistex, **c** Corel10K, **d** UKbench and **e** Batik

yields 36% with UKbench and CTM yields 21% with Corel10K). In Contrast, MTH which has been proposed to represent heterogeneous images yields better results in classifying heterogeneous images(e.g., 23% with Corel10K) and poor results in classifying texture images (e.g., 51% with Outex).

Table 1 Results obtained by classifying images from each dataset using different values of *k* (KNN's parameter)

		Outex	VisTex	Corel10K	UKbench
	TCM	72.49% ±2.8%	50.12% ±1.6%	16.71% ±0.1%	36.92% ±0.1%
	MTH	51.92% ±2.3%	68.36% ±1.9%	23.68% ±0.1%	50.00% ±0.1%
	CTM	72.31% ±2.8%	69.93% ±2.0%	21.22% ±0.2%	55.42% ±0.1%
	NRFU	36.72% ±1.7%	28.85% ±1.1%	5.38% ±0.0%	6.50% ±0.05%
k=1	CMTH	78.40% ±3.0%	83.57% ±2.1%	31.76% ±0.1%	64.42% ±0.1%
	TCM	76.42% ±3.0%	44.08% ±1.4%	16.29 % ±0.1%	28.67 % ±0.1%
	MTH	52.30% ±2.4%	60.75% ±1.8%	22.68 % ±0.2%	40.92 % ±0.1%
	CTM	70.18% ±2.8%	60.14% ±2.0%	20.02 % ±0.2%	44.92 % ±0.1%
	NRFU	36.89% ±1.8%	27.78% ±1.2%	5.26 % ±0.0%	5.83 % ±0.0%
k=3	CMTH	76.42% ±3.0%	72.83% ±2.1%	30.89 % ±0.2%	55.75 % ±0.1%
	TCM	68.41 % ±2.7%	44.08 % ±1.5%	17.51 % ±0.1%	28.08 % ±0.1%
	MTH	52.68 % ±2.5%	58.45 % ±1.9%	25.16 % ±0.2%	37.58 % ±0.1%
	CTM	68.93 % ±2.9%	56.76 % ±2.0%	20.97 % ±0.2%	41.58 % ±0.1%
	NRFU	39.49 % ±2.1%	28.50 % ±1.3%	6.00 % ±0.08%	6.08 % ±0.0
k=5	CMTH	74.83 % ±3.0%	69.20 % ±2.1%	33.16 % ±0.2%	51.25 % ±0.1%
	TCM	65.75 % ±2.7%	41.55 % ±1.5%	18.90 % ±0.2%	25.17 % ±0.1%
	MTH	52.13 % ±2.5%	54.59 % ±1.9%	26.88 % ±0.2%	31.42 % ±0.1%
	CTM	65.60 % ±2.8%	51.69 % ±2.0%	21.78 % ±0.2%	37.50 % ±0.2%
	NRFU	40.83 % ±2.2%	26.33 % ±1.4%	6.39 % ±0.0%	6.17 % ±0.0%
k=9	CMTH	71.72 % ±3.0%	64.37 % ±2.0%	34.06 % ±0.2%	43.83 % ±0.1%

Regarding the parameter *k* of KNN classifier, we can roughly say that the best value is *k=1* with some exceptions. TCM's accuracy jumps to 76% with Outex dataset when k equals 3, and then falls down again. Unlike the other methods, CTM shows a slight improvement in the classification correct rate of Corel10K.

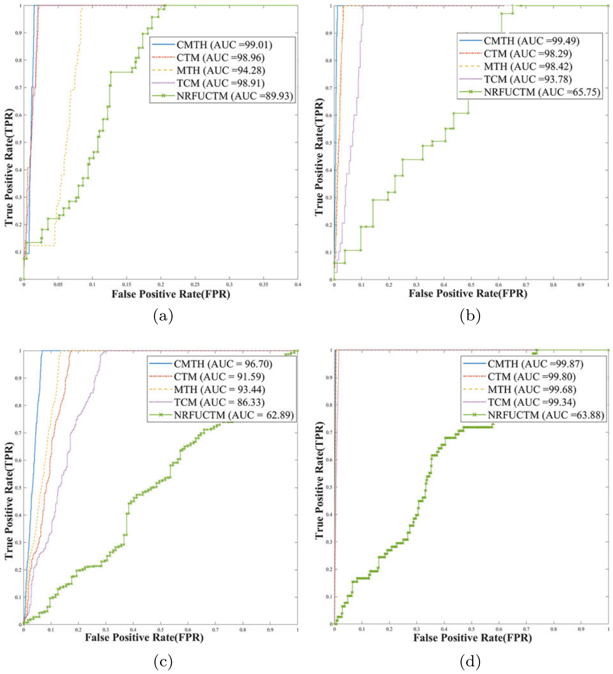
To have a clear idea about the performance gained by each feature, we plot the Receiver Operator Characteristic (ROC) curves. To have ROC extracted, we have to firstly calculate the True Positive Rate (TPR) and False Positive Rate (FPR) for the 4 folds of the experiments. These two metrics are dedicated for evaluating binary classification (positive/negative). Therefore, we transform our data into binary by, considering each time, one class as positive and the rest as negative. The TPR and FPR are calculated by:

$$TPR\,(\%) = \frac{positivetestingimagescalssefiedasPositive}{Poistivetestingimages} \times 100$$

$$FPR\,((\%) = \frac{negativetestingimagescalssefiedasPositive}{NumberofPoistivetestingimages} \times 100$$

Figure 8 presents the ROC obtained from all the involved features. Area Under the Curve (AUC) has been also calculated and presented in Fig. 8. AUC aims to confirm that the probability of classifying a randomly chosen positive instance is higher than a randomly chosen negative one.

From Fig. 8, it appears that our CMTH outperforms all the others in all datasets especially with UKbench where the model is trained with few samples(i.e. 3 images). This means that our method is more able to classify a random positive sample than a random negative one. By using KNN classifier, we can say, with certainty, that our method highly outperforms the others. But does it keep this performance with other classifiers?



(a)

(b)

(c)

(d)

Fig. 8 ROC and AUC yielded using KNN in **a** Outex dataset **b** Vistex dataset, **c** Corel10K dataset and **d** UKbench dataset

To answer this question, we have reconduct the evaluation using other classifiers namely, Support Vector Machine(SVM), Discriminant Analysis Classifier(DAC), Decision Tree(DT) and Naive Bayes(NB). The obtained results are given in Table 2.

Table 2 clearly shows that almost all the texton-based features have lost performance because of using the classifiers *DAC*, *SVM*, *DT* and *NB*. However, Our proposed *CMTH* keeps outperforming the others.

Among all features, *NRFUCTM*, which represents the texture as second order statistics, seems to be the least efficient. This confirms Julesz's texton theory which states that second-order statistics are not always identical to the difference of two textures [15, 17]. Hence, limiting features to only second order statistics in describing image content may not yield satisfactory representation of textures. On the other hand, *MTH* and *CTM* have overcome this limit by incorporating first order statistics. However, these two last features have yielded similar results because *MTH* doesn't consider all texton types, whereas, *CTM* totally ignores the orientation of the edges within a texture which is a fundamental cue in representing texture. CMTH, which incorporates information about edge orientations, first order statistics, second order statistics and considers using all texton types, yields far better results that sometimes exceed 100% improvement (e.g., classifying Corel10K with KNN *k=1*, CMTH = 31%, TCM=16% and NRFUCTM = 5%).

In thier recent study, Minarno et la. [41] reported that MTH has yielded satisfactory results in Batik classification. We, hence, compare the baseline they provided with the

Table 2 Accuracy obtained using different classifiers and different features

		Outex	VisTex	Corel10K	UKbench
DAC	TCM	41.96 % ±2.5%	39.73 % ±1.6%	23.01 % ±0.2%	66.17 % ±0.1%
	MTH	33.85 % ±2.4%	41.30 % ±1.4%	25.18 % ±0.2%	61.67 % ±0.1%
	CTM	33.65 % ±1.8%	42.39 % ±1.7%	19.89 % ±0.2%	66.42 % ±0.1%
	NRFU	39.83 % ±2.5%	39.37 % ±1.7%	21.09 % ±0.1%	34.17 % ±0.1%
	CMTH	42.34 % ±2.6%	51.21 % ±1.8%	34.30 % ±0.2%	73.08 % ±0.1%
SVM	TCM	42.95 % ±3.0%	37.08 % ±1.8%	20.58 % ±0.1%	18.67 % ±0.0%
	MTH	45.31 % ±2.6%	64.37 % ±1.5%	33.02 % ±0.1%	42.83 % ±0.0%
	CTM	27.43 % ±1.3%	59.78 % ±1.6%	21.15 % ±0.2%	46.17 % ±0.0%
	NRFU	31.51 % ±1.9%	26.93 % ±1.8%	11.69 % ±0.1%	7.08 % ±0.0%
	CMTH	51.01 % ±1.6%	72.46 % ±1.8%	40.61 % ±0.2%	53.75 % ±0.0%
DT	TCM	64.63 % ±2.5	45.41 % ±1.4%	18.33 % ±0.1%	24.67 % ±0.0%
	MTH	47.70 % ±2.3%	48.43 % ±1.5%	17.23 % ±0.1%	23.42 % ±0.0%
	CTM	67.38 % ±2.7%	54.71 % ±1.8%	19.69 % ±0.1%	27.08 % ±0.0%
	NRFU	60.81 % ±2.6%	38.16 % ±1.4%	15.96 % ±0.1%	12.58 % ±0.0%
	CMTH	64.45 % ±2.8%	58.57 % ±1.6%	22.94 % ±0.1%	24.25 % ±0.0%
NB	TCM	37.24 % ±1.7%	39.98 % ±1.4%	23.62 % ±0.2%	37.08 % ±0.07
	MTH	25.54 % ±1.1%	44.20 % ±1.5%	24.68 % ±0.2%	43.25 % ±0.0%
	CTM	34.97 % ±1.5%	43.96 % ±1.7%	23.06 % ±0.2%	48.58 % ±0.0%
	NRFU	41.01 % ±1.5%	42.03 % ±1.4%	22.71 % ±0.2%	19.42 % ±0.0%
	CMTH	38.71 % ±1.5%	51.81 % ±1.7%	34.22 % ±0.2%	46.17 % ±0.0%

Table 3 Classification accuracy obtained by classifying Batik dataset using different configurations

			KNN(K=5)	SVM
Train=50%	MTH	4 Textons	57.3%	49.3%
Test=50%		6 Textons	62%	59.3%
	CMTH	11 Textons	90%	89%
Train=60%	MTH	4 Textons	62%	58%
Test=40%		6 Textons	68%	65%
	CMTH	11 Textons	91%	91%
Train=70%	MTH	4 Textons	70%	64%
Test=30%		6 Textons	82%	76%
	CMTH	11 Textons	91%	89%

results we obtained using the same dataset configuration. Table 3 shows the results obtained for Batik classification using MTH and CMTH.

As we can see from Table 3, the proposed CMTH overcomes MTH in all possible configurations. In contrast to MTH, it seems that the accuracy of our CMTH isn't highly affected by the size of the training set. As instance, our system yields 90% accuracy when the classifier is trained with 150 images and almost the same accuracy (i.e., 91%) when it is trained with 100 images. Hence, we can say that the CMTH is richer with information than other features, and the classifiers could therefore be trained with a small number of images.

4.2 Image retrieval

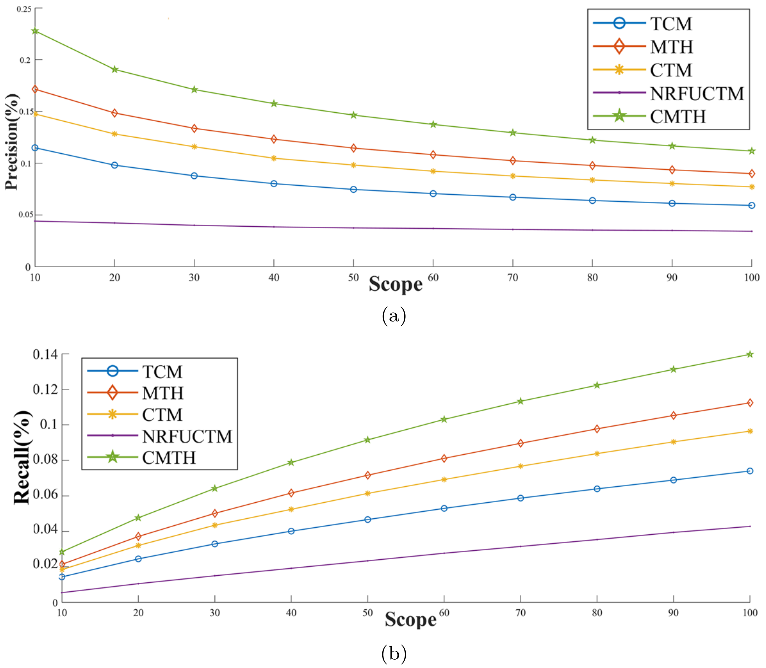
Image retrieval is one of the most frequently used applications to test descriptors viability. In this study, two metrics have been adopted to evaluate the retrieval performance of each descriptor, namely: precision and recall. Precision and recall are given by the formulas in (16) and (17) respectively. The obtained results are illustrated by Fig. 9

$$Precision = \frac{Number\ of\ retrieved\ relevant\ images}{Number\ of\ retrieved\ images} \tag{16}$$

$$Recall = \frac{Number\ of\ retrieved\ relevant\ images}{Number\ of\ relevant\ images\ in\ the\ dataset} \tag{17}$$

From Fig. 9, it appears that our method significantly outperforms the TCM, MTH, CTM and NRFUCTM for all scopes in terms of precision and recall. We also observe that as the scope increases from 10 to 100, the precision declines while the recall increases. This phenomena is known as precision-recall tradeoff. Although CTM is more recent, it appears that the MTH outperformed it. This can be attributed to the edge orientation information that are included in MTH but not in CTM. As it has been stated by [29], information about edge orientations could provide a semantic description of the image. The CMTH on the other hand, outperforms these two methods because it integrates the pros of the both. Since they have been proposed only for texture images, TCM and NRFUCTM have yielded poor results in retrieving images from the Corel10K.

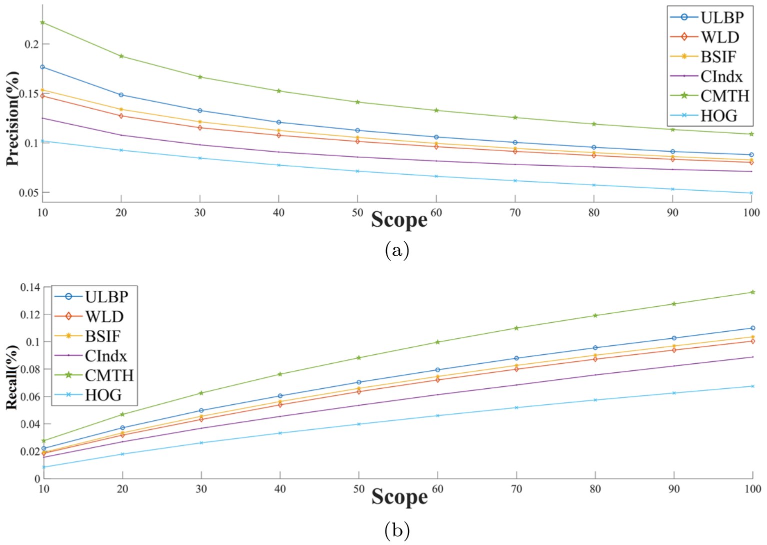
To prove the superiority of the proposed method, we have involved other widly-knwon and powerful features in the comparison. We have compared the CMTH with other well-known features, that have shown high performances, namely: Uniform Local Binary



(a)

(b)

Fig. 9 The yielded **a** precision and **b** recall by the descriptors TCM, MTH, CTM, NRFUCTM and CMTH



(a)

(b)

Fig. 10 The yielded **a** precision and **b** recall using the descriptors ULBP, WLD, HOG, BSIF, CIndex and CMTH

Table 4 Time (in seconds) consumed by each method for extracting a descriptor of one image with a size of 1024 × 1024 pixels

TCM	MTH	CTM	NRFUCTM	CMTH
0.59±0.01	0.26±0.01	0.41±0.01	1.68±0.01	0.23±0.02
ULBP	**WLD**	**BSIF**	**HOG**	**CIndx**
0.14±0.01	0.79±0.02	0.10±0.01	0.13±0.02	8.90±0.07

Pattern(ULBP) [44], Binarized Statistical Image Features(BSIF) [18], Histograms of Oriented Gradients(HOG) [8], Weber Local Descriptor(WLD) [5] and CIndex [10]. Figure 10 illustrates the precision/recall yielded by each feature using heterogeneous image dataset *Corel10K*.

As illustrated in Fig. 10, the proposed CMTH remarkably outperforms all the other features. ULBP, WLD and BSIF are powerful features that are widely used in applications, but they deal only with gray-level images. Such a weakness causes a degradation in their precision and recall when dealing with color images. Additionally, some of these feature descriptors,namely WLD and HOG, focus on orientation information whereas the others, namely ULBP, CIndx, BSIF, focus on the appearance frequency of some specific patterns (or Textons). More details and a comparative study between these descriptors could be found in [19]. In contrast, the high performance our method exhibits could be attributed to a set of factors which are: taking into account (a) color information, (b) orientation information, (c) information about the local and the global distribution of colors, and (d) textons distribution information.

As a final experiment, we evaluate the time consumed by each method to extract a descriptor of an image with a size of 1024 × 1024 pixels. The results have been presented in Table 4.

From Table 4, it appears that NRFUCTM consumes the highest time among other texton-based methods due to the heavy preprocessing step of finding and uniformig the non-uniform patterns. Our method, in the other hand, achieves a time of 0.23 second which is the second best time compared to the other texton-based methods. MTH consumes more time than CMTH because the former needs to verify, at pixel level, a constraint that links orientations to pixels colors. It should be mentioned that the consumed time of the the textons-based methods could highly be improved by using a more efficient language such c++ rather than Matlab.

5 Conclusion

In this paper, we have put forward a new descriptor that is able to represent texture and heterogeneous color images. Because it incorporates information about color, edge orientation and employs all types of textons, our method has the discrimination power of color, texture and shape features. Extensive experiments were conducted both in the context of image retrieval and classification using four datasets. Two of these datasets were meant to evaluate the ability of the proposed descriptor in texture representation and the other two to evaluate its ability in heterogeneous image representation. Experimental evaluation has shown that our proposed CMTH yields far better results than other related descriptors.

References

1. Ade F (1983) Characterization of textures by eigenfilters. Signal Process 5(5):451–457
2. Bala A, Kaur T (2016) Local texton xor patterns: a new feature descriptor for content-based image retrieval. Eng Sci Technol Int J 19(1):101–112
3. Banerji S, Verma A, Liu C (2011) Novel color lbp descriptors for scene and image texture classification. In: 15th International conference on image processing, computer vision, and pattern recognition, Las Vegas, Nevada. Citeseer, pp 537–543
4. Bu X, Wu Y, Gao Z, Jia Y (2019) Deep convolutional network with locality and sparsity constraints for texture classification. Pattern Recogn
5. Chen J, Shan S, He C, Zhao G, Pietikainen M, Chen X, Gao W (2010) Wld: a robust local image descriptor. IEEE Trans Pattern Anal Mach Intell 32(9):1705–1720
6. Chen J, Shan S, Zhao G, Chen X, Gao W, Pietikainen M (2008) A robust descriptor based on weber's law. In: IEEE conference on computer vision and pattern recognition, 2008. CVPR 2008. IEEE, pp 1–7
7. Coggins JM, Jain AK (1985) A spatial filtering approach to texture analysis. Pattern Recogn Lett 3(3):195–203
8. Dalal N, Triggs B (2005) Histograms of oriented gradients for human detection. In: IEEE Computer society conference on computer vision and pattern recognition, 2005. CVPR 2005, vol 1. IEEE, pp 886–893
9. Dougherty ER (1994) Morphological segmentation for textures and particles. Digital Image Process Methods 42:43–102
10. Florindo JB, Landini G, Bruno OM (2016) Three-dimensional connectivity index for texture recognition. Pattern Recogn Lett 84:239–244
11. Gangeh MJ, Sørensen L, Shaker SB, Kamel MS, De Bruijne M, Loog M (2010) A texton-based approach for the classification of lung parenchyma in ct images. In: International conference on medical image computing and computer-assisted intervention. Springer, pp 595–602
12. Goutsias J, Batman S (2000) Morphological methods for biomedical image analysis. Handbook Med Imaging 2:175–272
13. Haralick RM, Shanmugam K et al (1973) Textural features for image classification. IEEE Trans Syst Man Cybern (6): 610–621
14. Jalaja K, Bhagvati C, Deekshatulu BL, Pujari AK (2005) Texture element feature characterizations for cbir. In: 2005 IEEE International geoscience and remote sensing symposium, 2005. IGARSS'05. Proceedings, vol 2. IEEE, pp 4–pp
15. Julesz B (1981) Textons, the elements of texture perception, and their interactions. Nature 290(5802):91
16. Julesz B (1981) A theory of preattentive texture discrimination based on first-order statistics of textons. Biological Cybern 41(2):131–138
17. Julesz B, Bergen JR (1987) Human factors and behavioral science: Textons, the fundamental elements in preattentive vision and perception of textures. In: Readings in computer vision. Elsevier, pp 243–256
18. Kannala J, Rahtu E (2012) Bsif: binarized statistical image features. In: 2012 21st International conference on pattern recognition (ICPR). IEEE, pp 1363–1366
19. Khaldi B, Aiadi O, Kherfi ML (2019) Image classification using texture features and support vector machine (svm). In: Proc. of the 2nd International conference on artificial intelligence and information technology. Ouargla, Algeria, vol 2. University Kasdi Merbah, Ouargla
20. Khaldi B, Kherfi ML (2016) Modified integrative color intensity co-occurrence matrix for texture image representation. J Electron Imaging 25(5):053007
21. Kumari YS, Kumar VV, Satyanarayana Ch (2017) Texture classification using complete texton matrix. Int J of Image Graphics Signal Process 9(10):60
22. Kumari YS, Kumar VV, Satyanarayana Ch (2018) Classification of textures based on noise resistant fundamental units of complete texton matrix. Int J Image Graphics Signal Process 10(2)
23. Laws KI (1980) Textured image segmentation. Technical report University of Southern California Los Angeles Image Processing INST
24. Ledoux A, Losson O, Macaire L (2015) Texture classification with fuzzy color co-occurrence matrices. In: 2015 IEEE International conference on image processing (ICIP). IEEE, pp 1429–1433
25. Li Q, Shi Z (2010) Texture image retrieval using compact texton co-occurrence matrix descriptor. In: Proceedings of the international conference on multimedia information retrieval. ACM, pp 83–90
26. Lin H-C, Wang L-L, Yang S-N (1997) Extracting periodicity of a regular texture based on autocorrelation functions. Pattern Recog Lett 18(5):433–443
27. Lin H-C, Wang L-L, Yang S-N (1999) Regular-texture image retrieval based on texture-primitive extraction. Image Vis Comput 17(1):51–63

28. Liu G-H, Li Z-Y, Zhang L, Yong Xu (2011) Image retrieval based on micro-structure descriptor. Pattern Recogn 44(9):2123–2133
29. Liu G-H, Yang J-Y (2008) Image retrieval based on the texton co-occurrence matrix. Pattern Recogn 41(12):3521–3527
30. Liu G-H, Yang J-Y, Li ZY (2015) Content-based image retrieval using computational visual attention model. Pattern Recogn 48(8):2554–2566
31. Liu G-H, Zhang L, Hou Y-K, Li Z-Y, Yang J-Y (2010) Image retrieval based on multi-texton histogram. Pattern Recogn 43(7):2380–2389
32. Lohmann AW (1993) Image rotation, wigner rotation, and the fractional fourier transform. JOSA A 10(10):2181–2186
33. Lopez F, Acebron F, Valiente J, Perez E (2001) A study of registration methods for ceramic tile inspection purposes. Proceedings of the IX Spanish symposium on pattern recognition and image analysis 1:145–150
34. Lu T-C, Chang C-C (2007) Color image retrieval technique based on color features and image bitmap. Information Processing & Management 43(2):461–472
35. Lui L, Zhao L, Long Y, Kuang G, Fieguth P (2012) Extended local binary patterns for texture classification. Image Vis Comput 30(2):86–99
36. Malik J, Belongie S, Shi J, Leung T (1999) Textons, contours and regions: Cue integration in image segmentation. In: The proceedings of the seventh ieee international conference on computer vision, 1999. IEEE, pp 918–925
37. Malik J, Perona P (1990) Preattentive texture discrimination with early vision mechanisms. JOSA A 7(5):923–932
38. Maragos P, Schafer R (1986) Morphological skeleton representation and coding of binary images. IEEE Transactions on Acoustics, Speech, and Signal Processing 34(5):1228–1244
39. Mehrotra R, Namuduri KR, Ranganathan N (1992) Gabor filter-based edge detection. Pattern Recogn 25(12):1479–1494
40. Mikolajczyk K, Zisserman A, Schmid C (2003) Shape recognition with edge-based features. In: British machine vision conference (BMVC'03), vol 2, pp 779–788. The British Machine Vision Association
41. Minarno AE, Maulani AS, Kurniawardhani A, Bimantoro F (2018) Comparison of methods for batik classification using multi texton histogram. Telkomnika 16(3)
42. Nanni L, Ghidoni S, Brahnam S (2017) Handcrafted vs. non-handcrafted features for computer vision classification. Pattern Recogn 71:158–172
43. Nister D, Stewenius H (2006) Scalable recognition with a vocabulary tree. In: Computer society conference on computer vision and pattern recognition, vol 2. IEEE
44. Ojala T, Pietikainen M, Maenpaa T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans Pattern Anal Mach Intell 24(7):971–987
45. Prakash MJ, Kezia S, Prabha IS, Kumar VV (2013) A new approach for texture segmentation using gray level textons. International Journal of Signal Processing, Image Processing and Pattern Recognition 6(3):81–90
46. Raza A, Nawaz T, Dawood H, Dawood H (2019) Square texton histogram features for image retrieval. Multimed Tools Appl 78(3):2719–2746
47. Satpathy A, Jiang X, Eng H-L (2014) Lbp-based edge-texture features for object recognition. IEEE Trans Image Process 23(5):1953–1964
48. Shapira M, Rappoport A (1995) Shape blending using the star-skeleton representation. IEEE Comput Graph Appl 15(2):44–50
49. Swain MJ, Ballard DH (1991) Color indexing. Int J Comput Vision 7(1):11–32
50. Thiran J-P, Macq B (1996) Morphological feature extraction for the classification of digital images of cancerous tissues. IEEE Trans Biomed Eng 43(10):1011–1020
51. Tiwari D, Tyagi V (2016) A novel scheme based on local binary pattern for dynamic texture recognition. Comput Vis Image Underst 150:58–65
52. Tiwari D, Tyagi V (2017) Dynamic texture recognition using multiresolution edge-weighted local structure pattern. Comput Electrical Eng 62:485–498
53. Toyoda T, Hasegawa O (2005) Texture classification using extended higher order local autocorrelation, pp 131–136
54. Vadivel A, Sural S, Majumdar AK (2007) An integrated color and intensity co-occurrence matrix. Pattern Recogn Lett 28(8):974–983
55. Varma M, Zisserman A (2005) A statistical approach to texture classification from single images. International J Comput Vision 62(1-2):61–81
56. Venot A, Lebruchec JF, Roucayrol JC (1984) A new class of similarity measures for robust image registration. Comput Vision Graphics Image Process 28(2):176–184

57. Wang S, Velasco FRD, Wu AY, Rosenfeld A (1981) Relative effectiveness of selected texture primitive statistics for texture discrimination. IEEE Trans Syst Man Cybern 11(5):360–370
58. Weldon TP, Higgins WE, Dunn DF (1996) Efficient gabor filter design for texture segmentation. Pattern Recogn 29(12):2005–2015
59. Wood EJ (1990) Applying fourier and associated transforms to pattern characterization in textiles. Text Res J 60(4):212–220
60. Wu Y, Wu Y (2009) Shape-based image retrieval using combining global local shape features. In: 2nd International congress on image and signal processing, 2009. CISP'09. IEEE, pp 1–5
61. Zhao G, Pietikainen M (2007) Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Trans Pattern Anal Mach Intell (6): 915–928
62. Zhou XS, Huang TS (2001) Edge-based structural features for content-based image retrieval. Pattern Recogn Lett 22(5):457–468
63. Zhu S-C, Guo C-E, Wang Y, Xu Z (2005) What are textons? Int J Comput Vis 62(1-2):121–143

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Belal Khaldi received his BS, MSc and PhD degrees in computer science from the Université Kasdi Marbah, Algeria, in 2010, 2012 and 2016, respectively. He is a professor at the Department of Computer Science and Information Technology, Université Kasdi Marbah, Algeria. His research interests include image representation, feature extraction, computer vision and machine learning.

Oussama Aiadi received his BS, MSc and PhD degrees in computer science from the Université Kasdi Marbah, Algeria, in 2011, 2013 and 2016, respectively. He is a professor at the Department of Computer Science and Information Technology, Université Kasdi Marbah, Algeria. His research interests include biometrics, computer vision applications and machine learning.

Kherfi Mohammed Lamine received his BS degree in computer science from the Institut National d'Informatique, Algeria, and his MSc and PhD degrees in computer science from the Université de Sherbrooke, Canada. He is a professor at the Department of Computer Science and Information Technology, Université Kasdi Marbah, Algeria. His research interests include image andmultimedia retrieval, computer vision, image processing, machine learning, and web-based applications.