Computer Vision and Image Understanding 114 (2010) 981–989

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

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

Histogram similarity measure using variable bin size distance

Yu Ma [a], Xiaodong Gu [b], Yuanyuan Wang [b,*]

[a] State Key Laboratory of ASIC & System, Fudan University, Shanghai 200433, China
[b] Department of Electronic Engineering, Fudan University, Shanghai 200433, China

ARTICLE INFO

Article history:
Received 25 January 2010
Accepted 24 March 2010
Available online 29 March 2010

Keywords:
Variable bin size distance (VBSD)
Histogram similarity
Histogram distance
Image retrieval

ABSTRACT

In order to improve the performance of bin-by-bin distances, this paper proposes variable bin size distance (VBSD) as the histogram similarity measure. It calculates the histogram distance in a fine-to-coarse way, and can be considered as a cross-bin extension for bin-by-bin distances. The VBSD can be used to measure the similarity of multi-dimensional histograms, and is insensitive to both the histogram translation and the variation of histogram bin size. Experimental results show that the variable bin size distance performs better than bin-by-bin distances in the image retrieval applications.

1. Introduction

Histogram is an important statistical characteristic of data. In different image processing applications, the histogram is often used as the main feature to represent the distribution of the intensity, color, and texture parameters of images. As a statistical characteristic, the histogram is not sensitive to both translation and rotation of objects. Meanwhile, it is a standardized and compressed data storage type that can save much space. With these advantages, the histogram is widely used in image segmentation, registration, tracking, and especially in the image retrieval field that involves large amount of data [1].

In all kinds of applications, the distance between two histograms is the main evaluation to measure the similarity or dissimilarity of their corresponding statistical properties, where a larger distance means a lower similarity. Histogram distances can be divided into two categories. The bin-by-bin distances only compare corresponding histogram bins, while the cross-bin distances compare both corresponding bins and non-corresponding bins. These two categories have their respective advantages and disadvantages. The bin-by-bin distances are widely used in measuring similarities for large amount of data, for they are straightforward and easy to compute. However, they ignore correlations between neighboring bins, thus are sensitive to both the histogram translation and the change of the bin size. Although certain kinds of histogram shift, such as those caused by the noise, can be avoided at the stage of computing histograms [2], there are still many kinds of histogram shift that remain and influence the calculation of histogram distances. In contrast, the cross-bin distances take the correlations into account and represent the similarities more comprehensively. Among all the cross-bin distances, the Earth Mover's Distance (EMD) is recognized as the best feature distance measure [3,4]. The calculation of EMD requires solving an optimization problem, affecting the storage and computational efficiency. Thus the application of EMD for large database is constrained. In another view of dimensionality, general cross-bin distances except EMD are enough to solve most of the one-dimensional histogram problems. However, they can do nothing to multi-dimensional histograms. The EMD can measure the similarity of multi-dimensional histograms but it becomes very time-consuming when the dimensionality increases. Recently, several data preprocessing methods, such as quantization and clustering, have been proposed to facilitate the histogram distance computation especially for EMD [5–7], but they require prior information thus lack adaptability for unknown data. As a result, the bin-by-bin distances are still the primary means for the histogram similarity measure in multi-dimensional situations.

To improve the distance measure for multi-dimensional histograms, we propose the variable bin size distance (VBSD) measure. This measure achieves the effect of cross-bin distances by using the computation of bin-by-bin distances. It can be used to measure the similarity of high-dimensional histograms, for which cross-bin distances are powerless. We test this measure in image retrieval experiments to demonstrate its advantage.

This paper is organized as follows. In Section 2, the VBSD algorithm is elaborated and this distance is validated to be a metric. In Section 3, the performance of VBSD is compared with the bin-by-bin distances by image retrieval experiments. Finally, the

* Corresponding author. Fax: +86 21 65643526.
E-mail addresses: mayu@fudan.edu.cn (Y. Ma), xdgu@fudan.edu.cn (X. Gu), yywang@fudan.edu.cn (Y. Wang).

1077-3142/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.cviu.2010.03.006

characteristics of VBSD are discussed in Section 4 and conclusions are given in Section 5.

2. The variable bin size distance (VBSD) for histogram similarity measure

2.1. Histogram distances

The bin-by-bin distances compare corresponding histogram bins using different measure spaces. There are four types of bin-by-bin distances often used in measuring histogram similarity: L_1 distance, L_2 distance, χ^2 statistics, and Jeffrey divergence. They are labeled as D_{L1}, D_{L2}, $D_{\chi 2}$, and D_{Je}, respectively in the following text. If two histograms H and G with N bins are defined as $H = \{h_y\}$ and $G = \{g_y\}$, where the bin index $y \in \{1, 2, \ldots, N\}$, then four kinds of distances can be formulized as:

$$D_{L1}(H,G) = \sum_y |h_y - g_y| \tag{1}$$

$$D_{L2}(H,G) = \left(\sum_y |h_y - g_y|^2\right)^{1/2} \tag{2}$$

$$D_{\chi 2}(H,G) = \sum_y \frac{|h_y - g_y|^2}{2(h_y + g_y)} \tag{3}$$

$$D_{\mathrm{Je}}(H,G) = \sum_y \left(h_y \log\frac{h_y}{g_y} + g_y \log\frac{g_y}{h_y}\right) \tag{4}$$

These distance measures are defined by different theories and have individual characteristics. They only compare corresponding histogram bins and ignore the correlations between neighboring bins. Thus a slight translation of histograms might cause apparent change of the histogram distance. Moreover, these kinds of distances are sensitive to the bin size of histograms. In a refined binning mode, similar features may be separated in different bins and recognized as different, while in a coarse binning mode, the similarity of histograms are hard to represent effectively.

In contrast, the cross-bin distances overcome these shortcomings and make a more reasonable measure for the histogram similarity. There are several forms of cross-bin distances, e.g. the quadratic form distance [8] and the match distance [9]. The quadratic form distance uses a similarity matrix to reflect the similarity relationships across bins and compute the distance by matrix multiplication. However, in a majority of applications it works worse than bin-by-bin distances for overestimating the correlations across bins. The match distance is another form of cross-bin distance which calculates the cost of mapping two histograms. It works well in the similarity measure for low-dimensional histograms but is hard to be used in high-dimensional ones, because the mapping in high dimensional spaces requires complicated graph matching algorithms [10]. As a special kind of match distance, the Earth Mover's Distance (EMD) is recognized as the best feature distance measure. In the EMD, two features are considered as the 'earth' and the 'holes', respectively so that the distance measure problem is transformed into the earth moving problem, where the minimum cost of moving all the 'earth' into the 'holes' is calculated. The EMD method both considers the cross-bin correlations and reduces the sensitivity to bin size, resulting in good performance for feature distance measure. Besides, it is also powerful in solving the partial match problems. The main limitation fact of EMD is its high complexity and huge computation for solving the linear programming problem, especially when the dimensionality of feature space is high. In measuring histogram distances, the solving matrix is very huge when the bin number is large, e.g. for histograms with 100 bins the solving matrix size is 100-by-10,000. Although there have been improved methods to reduce the computation and obtain approximate performance [11,12], the optimization algorithms themselves are still complex. In many large-database applications such as image retrieval, high-dimensional histograms and the refined binning might result in large number of bins, preventing the EMD from exploiting its advantages. Besides these limitations, all these cross-bin distance measures, including the EMD, also have to face the problem of bin size selection as mentioned above.

2.2. The variable bin size distance (VBSD)

For high-dimensional data, simple and direct computation is required, and correlations across bins also need to be considered. We propose a variable bin size distance (VBSD) measure to resolve these conflicts. The VBSD is based on bin-by-bin distances and can achieve the effect of cross-bin distances. It could be considered and denoted as a cross-bin extension of corresponding bin-by-bin distances, e.g. 'the VBSD for L_1 distance'.

The main idea of VBSD is concise and comprehensible. Firstly, the bin-by-bin distance for the most refined bin size is computed as the first sub-distance. After that the intersection part of two histograms is subtracted from each histogram. Secondly, the remained two histograms are changed into coarser histograms with an increased bin size. The bin-by-bin distance is computed again for current bin size to get the second sub-distance, and so does the intersection part subtracted. This process continues until the bin size is large enough. Finally, the fine-to-coarse bin-by-bin sub-distances are obtained and a weighting function is used for all these sub-distances to get a summation. This summation, named as 'variable bin size distance', is used to represent the distance between the two histograms.

The detailed calculation process is elaborated as follows, where the sparse matrix is used to represent the histograms for convenient acquisition and storage of high-dimensional histograms. Suppose two histograms H_0 and G_0 are symbolized in the form of:

$$H_0 : \{(a_{1n}, a_{2n}, \ldots, a_{kn}), h_0(a_{1n}, a_{2n}, \ldots, a_{kn})\} \tag{5}$$

$$G_0 : \{(a_{1n}, a_{2n}, \ldots, a_{kn}), g_0(a_{1n}, a_{2n}, \ldots, a_{kn})\} \tag{6}$$

where the first part in the braces means the index and the second part means the value. The label k is the dimensionality, a_{in} ($i \in \{1, 2, \ldots, k\}$) is the position index for the nth bin with non-zero value in the ith dimensionality in the histogram space. $(a_{1n}, a_{2n}, \cdots\cdots, a_{kn})$ denotes the position of the nth non-zero bin, while $h_0(a_{1n}, a_{2n}, \cdots\cdots, a_{kn})$ and $g_0(a_{1n}, a_{2n}, \cdots\cdots, a_{kn})$ denote the histogram values correspond to this bin.

Firstly, the bin-by-bin histogram distance for the most refined bin size is computed, denoted as d_0:

$$d_0 = D_x(H_0, G_0) \tag{7}$$

where D_x represents any kind of the bin-by-bin distances, e.g. D_{L1}, D_{L2}, $D_{\chi 2}$, and D_{Je}.

Meanwhile, the intersection part of two histograms is subtracted from each histogram:

$$\begin{cases} H_0' = H_0 - (H_0 \cap G_0) \\ G_0' = G_0 - (H_0 \cap G_0) \end{cases} \tag{8}$$

Then the bin size for each dimension is enlarged, i.e. becoming s times of the current one, and the remained two histograms after subtraction are changed accordingly. Usually the magnification s is selected as two. Suppose H_0' is symbolized as:

$$H_0' : \{(a_{1n}, a_{2n}, \ldots, a_{kn}), h_0'(a_{1n}, a_{2n}, \ldots, a_{kn})\} \tag{9}$$

and the new histogram with new bin size is:

$$H_1 : \{(b_{1p}, b_{2p}, \ldots, b_{kp}), h_1(b_{1p}, b_{2p}, \ldots, b_{kp})\} \tag{10}$$

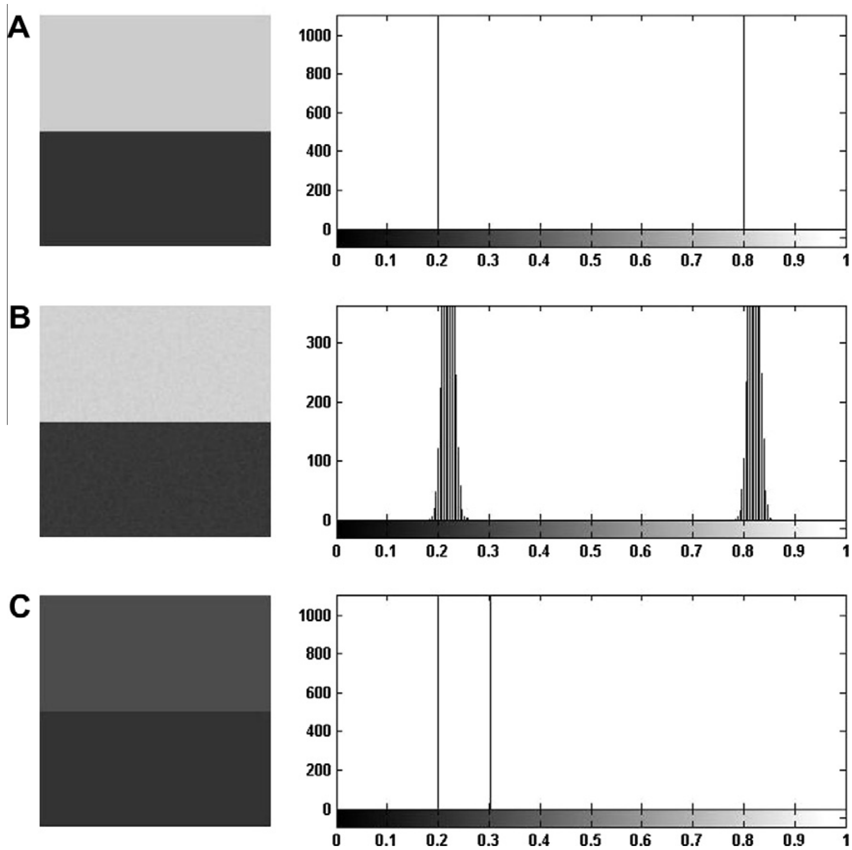

Fig. 1. Three simple-pattern images and their histograms.

where the new position index b_{ip} satisfies:

$$\max_p (b_{ip}) < \frac{\max_n(a_{in})}{s} + 1 \tag{11}$$

The new histogram h_1 is calculated by:

$$h_1(b_{1p}, b_{2p}, \ldots, b_{kp}) = \sum_{c_1, c_2, \ldots, c_k} h'_0(c_1, c_2, \ldots, c_k) \tag{12}$$

where $(c_1, c_2, \ldots, c_k)$ denotes all the possible histogram positions satisfying:

$$s \cdot (b_{ip} - 1) < c_i \leqslant s \cdot b_{ip} \tag{13}$$

Similarly, the new histogram G_1 is computed in the same way. Then the new bin-by-bin sub-distance at this bin size, denoted as d_1, can be computed:

$$d_1 = D_x(H_1, G_1) \tag{14}$$

The same steps are repeated several times until the bin size is large enough, and a series of fine-to-coarse sub-distances are saved as d_2, d_3, $d_4, \ldots, d_{tmax}$.

We use a positive weighting function f to summarize these distances and define the variable bin size distance VD_x corresponding to the distance D_x as:

$$VD_x = f(d_0, d_1, \ldots, d_{tmax}) = \sum_{t=0}^{tmax} (w_t \cdot d_t) \tag{15}$$

where the positive parameter w_t is the weight of d_t. In fact, the simplest form of f is the average function, making VD_x the mean value of all these distances:

$$VD_x = \frac{1}{1 + tmax} \sum_{t=0}^{tmax} d_t \tag{16}$$

It is obvious that if the calculation level *tmax* is set to zero, the VBSD reduces to the bin-by-bin distance. Thus, the VBSD includes the bin-by-bin distance actually and can be considered as an extension.

2.3. The metric space validation of VBSD

To be a metric, a measure has to satisfy the following properties: non-negativity, symmetry, and identity [3]. For any two elements *A* and *B* in the metric space, their distance *D* satisfies:

$$D(\mathrm{A}, \mathrm{B}) \geqslant 0 \tag{17}$$

$$D(\mathrm{A}, \mathrm{B}) = D(\mathrm{B}, \mathrm{A}) \tag{18}$$

$$D(\mathrm{A}, \mathrm{B}) = 0 \text{ if and only if } \mathrm{A} = \mathrm{B} \tag{19}$$

Besides, for any three elements A, B and C in the metric space, their distance relationships satisfy the triangle inequality:

$$D(\mathrm{A}, \mathrm{B}) + D(\mathrm{B}, \mathrm{C}) \geqslant D(\mathrm{A}, \mathrm{C}) \tag{20}$$

The four kinds of bin-by-bin distances mentioned above are metrics which satisfy these properties. Now we have to verify that the VBSD is also a metric. According to the calculation, each d_t is equivalent to D_x so that they satisfy metric properties:

$$d_t(\mathrm{A}, \mathrm{B}) \geqslant 0 \tag{21}$$

$$d_t(\mathrm{A}, \mathrm{B}) = d_t(\mathrm{B}, \mathrm{A}) \tag{22}$$

$$d_t(\mathrm{A}, \mathrm{B}) = 0 \text{ if and only if } \mathrm{A} = \mathrm{B} \tag{23}$$

$$d_t(\mathrm{A}, \mathrm{B}) + d_t(\mathrm{B}, \mathrm{C}) \geqslant d_t(\mathrm{A}, \mathrm{C}) \tag{24}$$

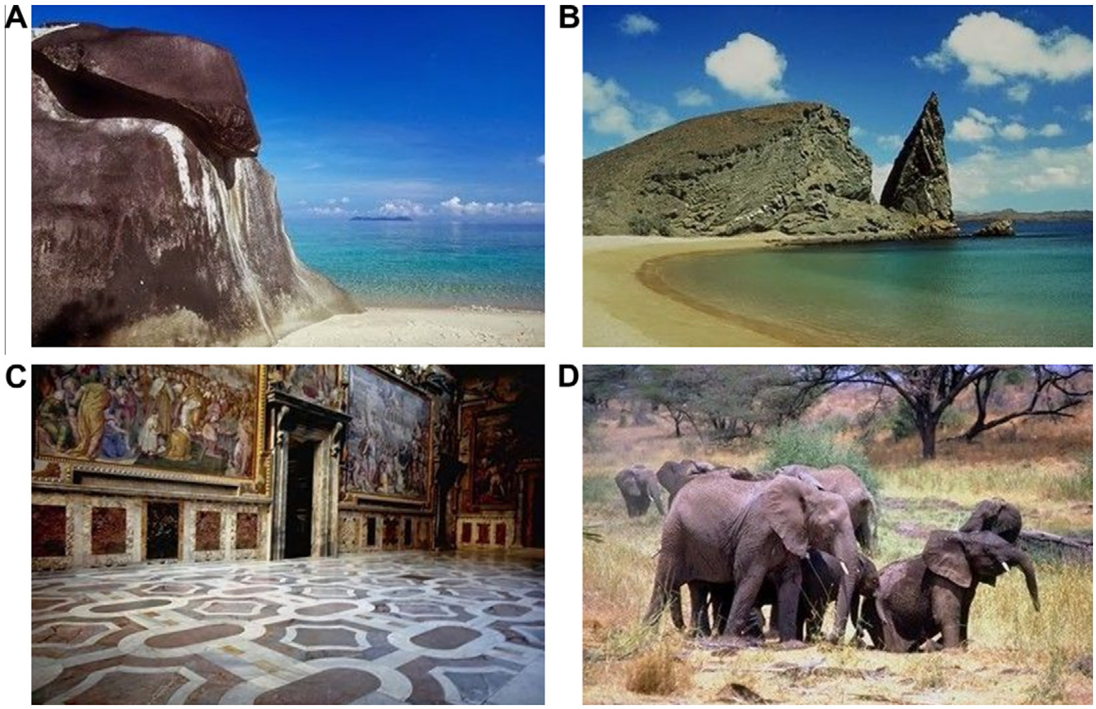

Fig. 2. Four images from the COREL 1000 database.

Accordingly,

1. $VD_x(\mathrm{A},\mathrm{B}) = \sum_{t=0}^{tmax}[w_t \cdot d_t(\mathrm{A},\mathrm{B})] \geqslant 0.$
2. $VD_x(\mathrm{A},\mathrm{B}) = \sum_{t=0}^{tmax}[w_t \cdot d_t(\mathrm{A},\mathrm{B})] = \sum_{t=0}^{tmax}[w_t \cdot d_t(\mathrm{B},\mathrm{A})] = VD_x(\mathrm{B},\mathrm{A}).$
3. If $VD_x(A,B) = 0$ and each w_t is non-zero, then every d_t has to equal to zero. It is apparent that $A = B$.
4. $$\begin{aligned} VD_x(\mathrm{A},\mathrm{B}) + VD_x(\mathrm{B},\mathrm{C}) &= \sum_{t=0}^{tmax}[w_t \cdot d_t(\mathrm{A},\mathrm{B})] + \sum_{t=0}^{tmax}[w_t \cdot d_t(\mathrm{B},\mathrm{C})] \\ &= \sum_{t=0}^{tmax}[w_t \cdot d_t(\mathrm{A},\mathrm{B}) + w_t \cdot d_t(\mathrm{B},\mathrm{C})] \\ &\geqslant \sum_{t=0}^{tmax}[w_t \cdot d_t(\mathrm{A},\mathrm{C})] = VD_x(\mathrm{A},\mathrm{C}) \end{aligned}.$$

Therefore, the VBSD is a metric.

3. The comparison between VBSD and corresponding bin-by-bin distances

The VBSD is calculated from their corresponding bin-by-bin distances so that their performances can be compared in pairs. To illustrate the advantage of the VBSD, we firstly use the simple and straightforward examples to illustrate its characteristics, and then compare their performances in texture image retrieval and color image retrieval.

3.1. Distance comparison

The VBSD considers the correlations between bins, thus it can handle the situation of histogram translation. In this regard, it is similar to cross-bin distances. We use several examples to test this property, including both one-dimensional histograms and multi-dimensional ones.

For the comparison on one-dimensional examples, gray-level histograms are used. There are three simple-pattern images: the first one only consists of dark and bright blocks, the second one is the same as the first one except for being blurred by gaussian noise, and the third one also consists of dark and bright blocks, where one half is the same while the other half is different compared to the first one. The three images and their corresponding gray-level histograms are shown in Fig. 1.

From the visual point, it is obvious that the histogram difference between A and B, which comes from the gaussian noise and can be just seen as a slight shift or diffusion, is smaller than the essential

Table 1
Distance measure comparison for gray-level histograms.

		(A, B)		(A, C)
256 Bins	D_{L1}	1.955	>	1
	D_{L2}	0.729	>	0.707
	$D_{\chi 2}$	0.956	>	0.5
	D_{Je}	1.279	>	0.693
	VD_{L1}	0.955	<	1
	VD_{L2}	0.427	<	0.707
	$VD_{\chi 2}$	0.452	<	0.5
	VD_{Je}	0.605	<	0.693
128 Bins	D_{L1}	1.905	>	1
	D_{L2}	0.750	>	0.707
	$D_{\chi 2}$	0.910	>	0.5
	D_{Je}	1.199	>	0.693
	VD_{L1}	0.805	<	1
	VD_{L2}	0.379	<	0.707
	$VD_{\chi 2}$	0.374	<	0.5
	VD_{Je}	0.497	<	0.693
64 Bins	D_{L1}	1.696	>	1
	D_{L2}	0.743	>	0.707
	$D_{\chi 2}$	0.7546	>	0.5
	D_{Je}	0.983	>	0.693
	VD_{L1}	0.607	<	1
	VD_{L2}	0.307	<	0.707
	$VD_{\chi 2}$	0.273	<	0.5
	VD_{Je}	0.362	<	0.693
32 Bins	D_{L1}	1.262	>	1
	D_{L2}	0.667	<	0.707
	$D_{\chi 2}$	0.503	>	0.5
	D_{Je}	0.646	<	0.693
	VD_{L1}	0.339	<	1
	VD_{L2}	0.193	<	0.707
	$VD_{\chi 2}$	0.143	<	0.5
	VD_{Je}	0.187	<	0.693

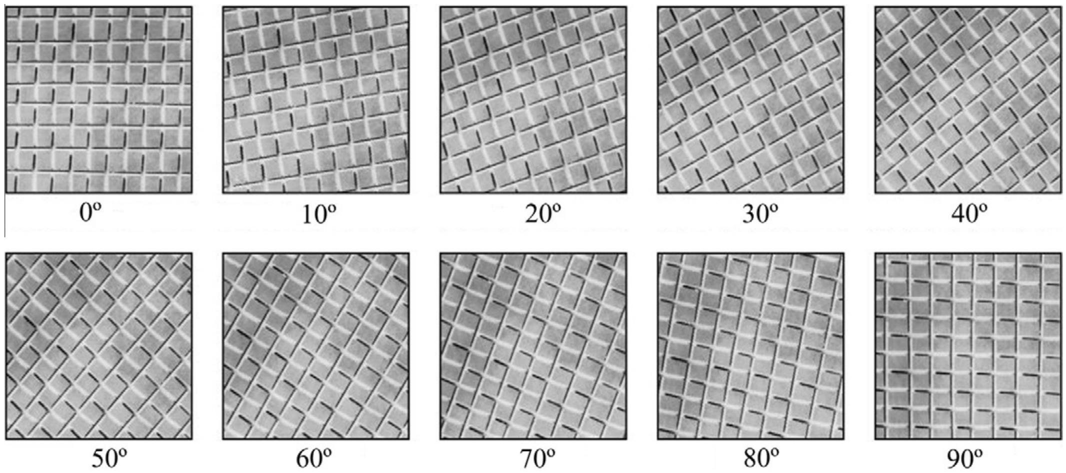

Fig. 3. An example of the rotation results for the texture 'D1'.

difference between histograms A and C. However, the bin-by-bin distances may give the opposite conclusion, i.e. the histogram distance between A and B is larger than the one between A and C. Numerical results for different bin sizes are listed in Table 1, where four kinds of bin-by-bin distances and their corresponding VBSDs are computed from the histogram pairs. The bin numbers are selected as 256, 128, 64, and 32, respectively, leading to different bin sizes. Here, the 'bin number' for the VBSD means the initial bin number in the algorithm. As seen from the results, most of bin-by-bin distances between A and B for different bin sizes are larger than their corresponding ones between A and C. It means that they could not measure the similarity correctly in this situation. In contrast, their respective VBSDs can measure the similarity correctly.

After using synthetic images to show the property of the VBSD, we made an actual comparison on the distances between natural color images as shown in Fig. 2. The images come from the Corel 1000 image database that has been used in many image retrieval applications such as the content-based color image retrieval [13,14].

In Fig. 2, the image 'A' is selected as the reference one, which mainly consists of the blue[1] sky, the cyan seawater, the white cloud, the sienna rock, and the wheat beach. The image 'B' has the similar content and color types to 'A' except that certain colors are lighter or darker. In contrast, the image 'C' is a indoor scene and the image 'D' consists of elephants, grass and trees, which are far different from the image 'A'. For comparison, three-dimensional color histograms of these four images are firstly counted, where the bin numbers for each dimension is set as 64, 32, and 16, respectively. Then all the eight kinds of histogram distances between the pairs (A, B), (A, C), and (A, D) are calculated, with results shown in Table 2. Under the definition of a reasonable subjective, the histogram distance between 'A' and 'B' should be smaller than the ones between 'A' and 'C' or 'D'. However, almost all bin-by-bin distances give the opposite conclusion. On the other hand, their respective VBSDs give the correct answer with only a few exceptions. Results that do not reflect the similarity well are labeled in italic in Table 2.

Table 2
Distance measure comparison for multi-dimensional histograms.

		(A, B)	(A, C)	(A, D)
64 Bins for each dimension	D_{L1}	*1.808*	*1.769*	*1.735*
	D_{L2}	*0.185*	*0.049*	*0.061*
	$D_{\chi 2}$	*0.861*	*0.838*	*0.823*
	D_{Je}	*1.158*	*1.136*	*1.121*
	VD_{L1}	1.216	1.327	1.218
	VD_{L2}	0.221	0.223	*0.214*
	$VD_{\chi 2}$	0.563	0.608	*0.558*
	VD_{Je}	0.756	0.811	*0.744*
32 Bins for each dimension	D_{L1}	*1.723*	*1.520*	*1.546*
	D_{L2}	*0.089*	*0.082*	*0.075*
	$D_{\chi 2}$	*0.805*	*0.665*	*0.697*
	D_{Je}	*1.074*	*0.867*	*0.922*
	VD_{L1}	1.076	1.152	1.248
	VD_{L2}	0.161	0.232	0.237
	$VD_{\chi 2}$	0.488	0.524	0.582
	VD_{Je}	0.652	0.691	0.782
16 Bins for each dimension	D_{L1}	*1.547*	*1.322*	*1.388*
	D_{L2}	*0.161*	*0.138*	*0.138*
	$D_{\chi 2}$	*0.687*	*0.558*	*0.600*
	D_{Je}	*0.888*	*0.721*	*0.787*
	VD_{L1}	0.915	1.051	1.167
	VD_{L2}	0.180	0.271	0.280
	$VD_{\chi 2}$	0.397	0.470	0.535
	VD_{Je}	0.519	0.615	0.715

3.2. The performance of VBSD in image retrieval experiments

A very important application of the histogram distance measure is the image retrieval, i.e. finding relevant images to the reference one in the image database. The advantage of histogram is well reflected in this field especially for large database. The main types of image retrieval contain the texture image retrieval and the color image retrieval, on which many studies focus [15–17]. Here, we test the VBSD measure in these two retrieval tasks. In each experiment, the bin-by-bin distances and their corresponding VBSDs are adopted respectively and their performances are compared.

3.2.1. Texture retrieval based on Gabor features

In the texture retrieval tasks, a very effective feature is the Gabor feature which describes the distribution of different scale-orientation pairs in one texture pattern [18]. The Gabor feature is a two-dimensional statistical feature which can be seen as a two-dimensional histogram. In the following text, it is named as 'Gabor histogram' for convenience. To search for a similar texture pattern to the reference one, it is necessary to calculate Gabor histograms of both the reference texture image and all the candidate texture images, and then compare the distances between each candidate histogram and the reference one. In the measuring of Gabor

[1] For interpretation of color in Fig. 2, the reader is referred to the web version of this article.

histogram distances, the bin-by-bin distances are usually appropriate. However, the Gabor features change with respect to the rotation of texture, making the histogram not rotation invariant any more. When the texture image is rotated, its Gabor histogram shifts in the dimension of orientation. In this situation, the bin-by-bin distances could not reflect the histogram similarity well, so that same textures except for only orientation difference may be recognized as disparate ones. The common solution to this problem is calculating histogram distances for all possible rotations and selecting the minimum one. Besides, rotation-invariant methods have also been proposed such as the one using polar-wavelet texture feature [19]. However, these methods increase the computation greatly and more effective methods are required. According to the characteristics of the VBSD, it has robustness to the histogram translation so that it may reduce the influence of rotation. To test it, we use the VBSD to measure Gabor histogram distances for texture retrieval and compare the results with bin-by-bin measures.

Experiments are based on the Brodatz texture database that contains 111 different types of texture images [20]. The database is expended to a larger 'rotation database' with 1110 texture images that come from original textures with rotations of 0°, 10°, 20°, 30°, 40°, 50°, 60°, 70°, 80°, and 90°, respectively. It means that there are a group of 10 textures coming from each original texture and they are labeled as similar textures. One example of the rotations for the original texture 'D1' is shown in Fig. 3.

After obtaining the rotation database with 1110 texture images, we calculate the Gabor histograms for all the images. The experiments are repeated twice with two different Gabor resolutions: one is six scales by eight orientations as usually used in texture analysis, and the other is eight scales by sixteen orientations for a higher precision. Each member of the database is used as the reference one in turn and its similar texture images are retrieved by comparing histogram distances. The bin-by-bin distances and their corresponding VBSDs are used, respectively, where the VBSD algorithm is only performed on the dimension of orientations for Gabor histograms. For Gabor histograms with six scales and eight orientations, average results for all the reference images are shown in Fig. 4. The retrieval results are represented using the precision–recall curves that are most commonly used in image retrieval. The 'recall' means the percentage of the retrieved relevant images in all the relevant images, while the 'precision' means the percentage of the retrieved relevant images in all the retrieved images. In the figure, four solid lines represent the bin-by-bin distances, while four dashed lines represent the VBSD and the colors reflect the correspondence. Similarly, Fig. 5 shows the average precision–recall curves for the Gabor histograms with eight scales and sixteen orientations. It is obvious from the results that the VBSD measures the texture similarity better than corresponding bin-by-bin distances in the situation of rotation.

3.2.2. Image retrieval based on color histograms

The color histogram is often used in image retrieval tasks because it is a simple way to describe the content of an image. To compare the distance measure effect in color-based image retrieval, the VBSD is tested in the COREL 1000 image database and compared with the bin-by-bin distances. This database contains 10 image categories: 'Africa', 'Beach', 'Building', 'Bus', 'Dinosaur', 'Elephant', 'Flower', 'Horse', 'Mountain', and 'Meal'. It has been used in various studies on image retrieval, especially the content-based image retrieval. Now we just use those images for the histogram-based image retrieval.

For each image, its color distribution in the RGB color space forms a three-dimensional histogram, where the bin number of each dimension determines the histogram resolution. For the bin-by-bin distances, the bin number may have a major influence on the retrieval result, so we test the results for different bin numbers as a comparison. In contrast, the VBSD does not have this problem and only needs to select an initial bin size.

Using each of the 1000 images as the reference image to do the histogram-based retrieval, the average precision-recall curves for L_1 distances are shown in Fig. 6. Four solid lines represent the L_1 distances with bin numbers of 64, 32, 16 and 8 in each dimension, respectively, while the dashed line represents the VBSD. Similarly, Figs. 7–9 show the results for D_{L2}, $D_{\chi 2}$, and D_{Je}, respectively.

According to the average results, there seems to be not large differences when using different bin numbers or VBSD. The VBSD just has a slight advantage over bin-by-bin distances. However, some fact may be overlooked when noticing only the average result. If seen from the results for individual image categories, there are significant differences between categories. Using L_1 distance as an example, the results for different categories are shown in Fig. 10.

From the observation, the retrieval performance of bin-by-bin distances is much affected by bin numbers. For example, a small bin number may work better than a large bin number for several image categories, such as the 'Beach', 'Elephant' and 'Mountain'

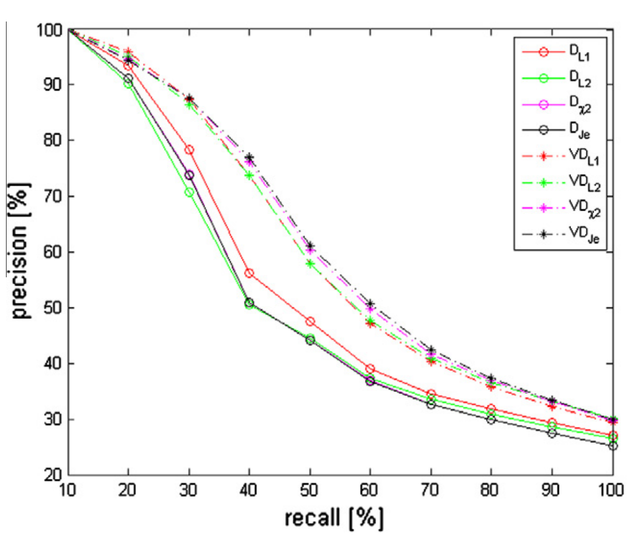

Fig. 4. Average texture retrieval curves for the rotation database, where the Gabor histograms are with six scales and eight orientations.

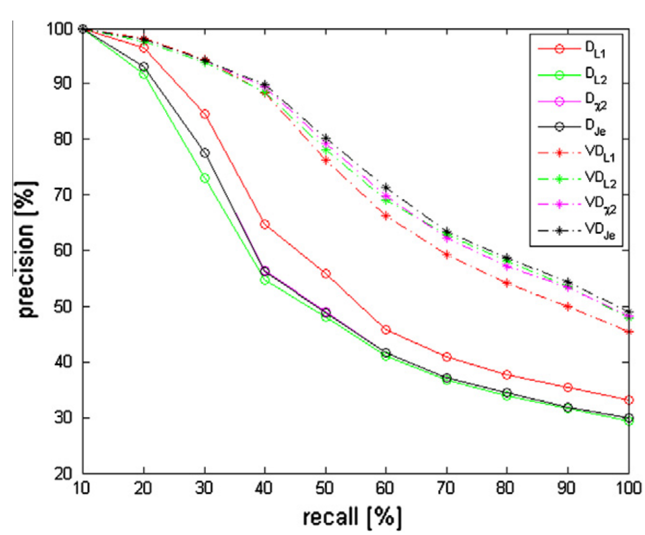

Fig. 5. Average texture retrieval curves for the rotation database, where the Gabor histograms are with eight scales and sixteen orientations.

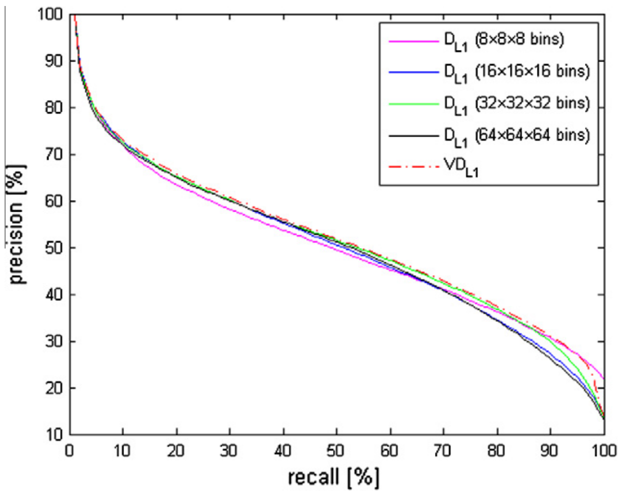

Fig. 6. The average precision–recall curves of histogram-based image retrieval using L_1 distance and corresponding VBSD.

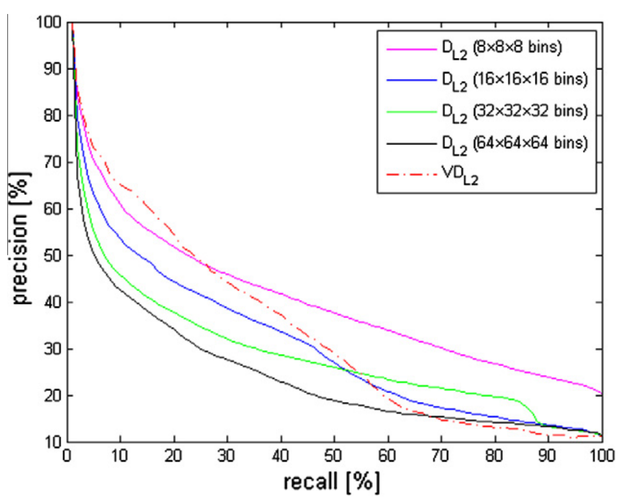

Fig. 7. The average precision–recall curves of histogram-based image retrieval using L_2 distance and corresponding VBSD.

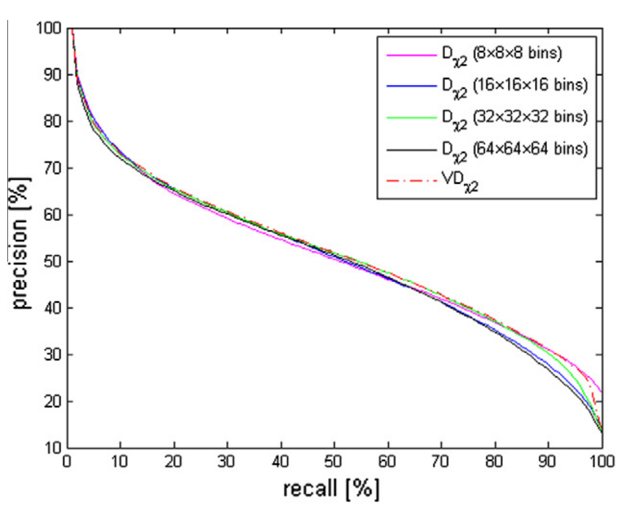

Fig. 8. The average precision–recall curves of histogram-based image retrieval using χ^2 statistics and corresponding VBSD.

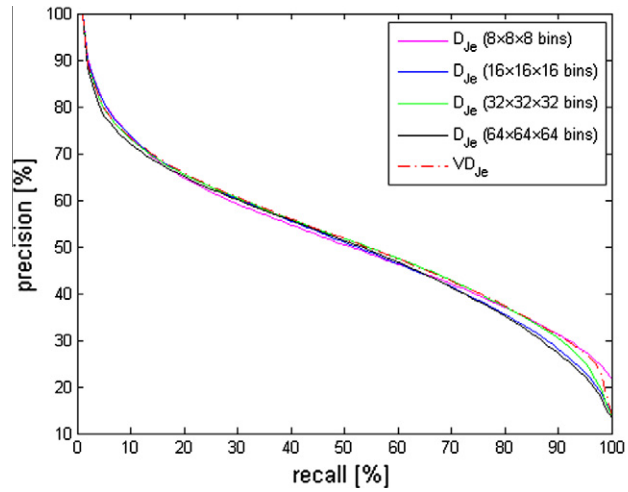

Fig. 9. The average precision–recall curves of histogram-based image retrieval using Jeffrey divergence and corresponding VBSD.

in the database. However, it causes noticeable performance degradation for other categories such as 'Bus', 'Dinosaur', 'Flower', 'Horse', and 'Meal'. This problem makes it difficult to select an appropriate bin number for histogram-based image retrieval when the objective category is unknown. In contrast, the VBSD alleviates the influence of bin number and has a relatively stable performance for different image types. The same conclusion can be achieved for other types of bin-by-bin distances such as L_2 distance, χ^2 statistics, and Jeffrey divergence. For space consideration, we do not list all the results here. Experimental results show that the VBSD has satisfying performance and stableness for different image categories compared to bin-by-bin distances. In practical applications, statistical properties of the database to be searched in are often unknown, so that the VBSD may be more appropriate than bin-by-bin distances that have the problem of selecting bin numbers.

4. Discussion

According to the experiments on the histogram-based texture image retrieval and color image retrieval, we find that the VBSD has certain advantages over bin-by-bin distances. Firstly, it is not sensitive to the histogram translation, which is a frequent situation caused by illumination, noise, or the rotation of texture patterns. Secondly, it is not sensitive to bin numbers because of the fine-to-coarse procedures in its calculation. This property makes it convenient in retrieving in the unknown database. It needs to be emphasized that the VBSD is not equivalent to the simple averaging of the distances under different bin numbers, because the influence under the refined size has been eliminated when calculating the distances under the coarse size. Accordingly, the VBSD is close to the cross-bin distances in a sense. In other words, it achieves the effect of cross-bin distances using the computation of bin-by-bin distances. The average computational cost for the VBSD is about three times as the one for corresponding bin-by-bin distances with the same histogram size. Thus, compared to the complex computation of cross-bin distances such as EMD [21], the time-consuming of the VBSD is more acceptable in image retrieval. Furthermore, the storage cost is also far less than cross-bin distances. These advantages are valuable for the retrieval in large databases.

The VBSD is designed for the distance measure for histograms especially high-dimensional ones, so that it can be used in various applications which need comparing histogram distances, not only the histogram-based retrieval. It can also be used for features

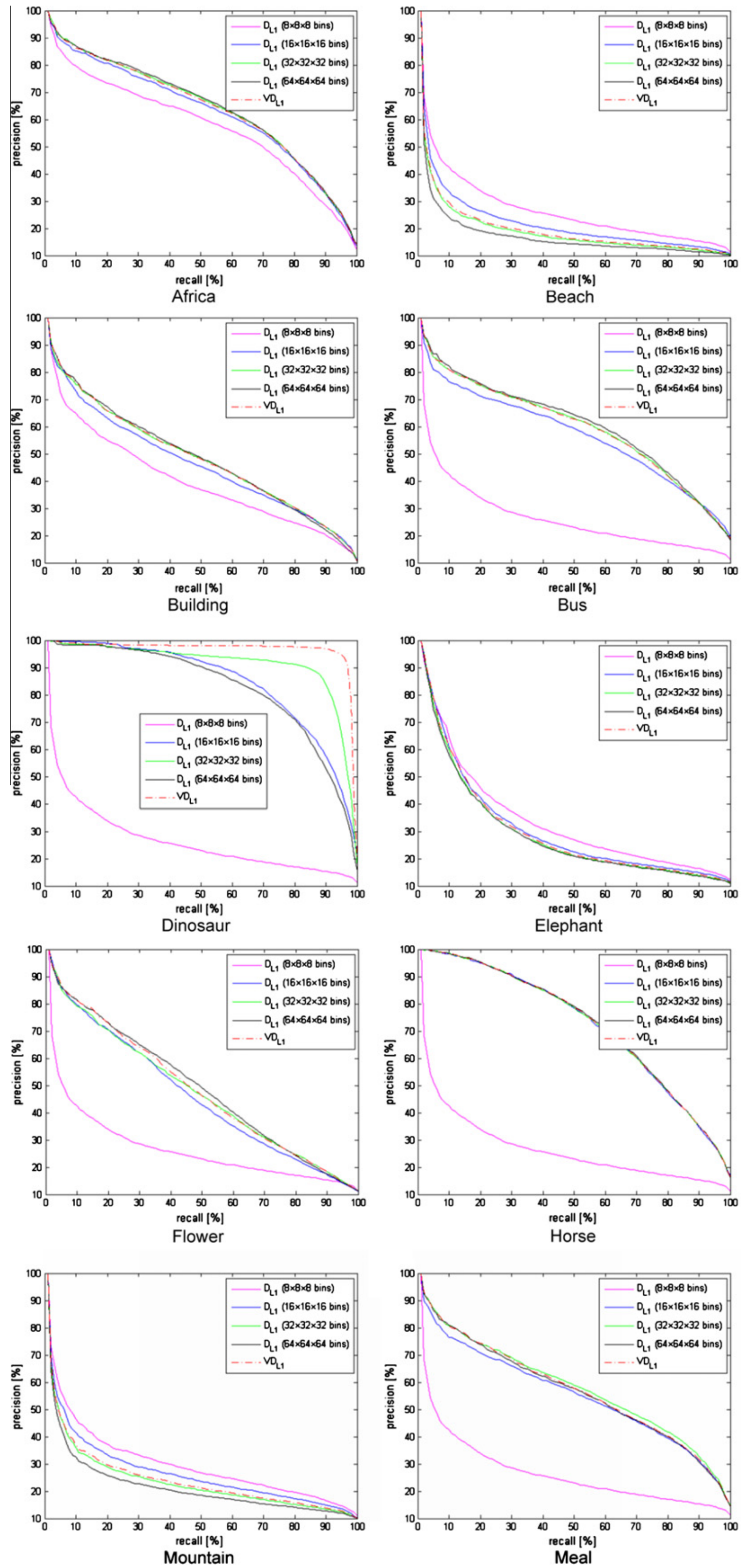

Fig. 10. The retrieval results for different image categories using L_1 distance and corresponding VBSD.

instead of histograms as long as the features can be quantified uniformly. However, this measure is based on bin-by-bin distances thus could do nothing if the features are defined in other ways, such as semantic features, shape descriptors, and fuzzy expressions. For this reason, the advantage of the VBSD in image retrieval is only reflected on the histogram-based ones, instead of the other kinds. In all, the VBSD complements and improves the bin-by-bin distances by incorporating the correlations across bins.

5. Conclusions

In this paper we propose the variable bin size distance measure as a new histogram distance measure. This measure takes the correlations across bins into account and achieves the effect of cross-bin distances using the calculations on bin-by-bin distances. It can be used to measure the similarity of histograms especially the multi-dimensional histograms, and is insensitive to both the histogram translation and the variation of histogram bin size. Meanwhile, the computation and storage costs of the VBSD are less than cross-bin distances like EMD. Experimental results show its better performance over bin-by-bin distances for the image retrieval applications especially for the texture retrieval.

Acknowledgments

This work was supported by the National Basic Research Program of China (No. 2005CB724303), Natural Science Foundation of China (No. 10974035) and the Program of Shanghai Subject Chief Scientist (No. 10XD1400600).

References

[1] R. Datta, D. Joshi, J. Li, J.Z. Wang, Image retrieval: Ideas, influences, and trends of the new age, ACM Comput. Survey 40 (2) (2008) 1–60.
[2] T. Gevers, H. Stokman, Robust histogram construction from color invariants for object recognition, IEEE Trans. Pattern Anal. Mach. Intell. 26 (1) (2004) 113–118.
[3] Y. Rubner, Perceptual Metrics for Image Database Navigation, Stanford University, Stanford, 1999.
[4] Y. Rubner, C. Tomasi, L.J. Guibas, The earth mover's distance as a metric for image retrieval, Int. J. Comput. Vision 40 (2) (2000) 99–121.
[5] F.S. Lim, W.K. Leow, Adaptive histograms and dissimilarity measure for texture retrieval and classification, Proc. Int. Conf. Image Process. 2 (2002) 825–828.
[6] N. Vasconcelos, On the efficient evaluation of probabilistic similarity functions for image retrieval, IEEE Trans. Inform. Theor. 50 (7) (2004) 1482–1496.
[7] S. Jeong, C.S. Won, R.M. Gray, Image retrieval using color histograms generated by gauss mixture vector quantization, Comput. Vision Image Understand. 94 (1) (2004) 44–66.
[8] J. Hafner, H. Sawhney, W. Equitz, M. Flickner, W. Niblack, Efficient color histogram indexing for quadratic form distance functions, IEEE Trans. Pattern Anal. Mach. Intell. 17 (7) (1995) 729–736.
[9] H.C. Shen, A.K.C. Wong, Generalized texture representation and metric, Comput. Vision Graph. Image Process. 23 (2) (1983) 187–206.
[10] M. Werman, S. Peleg, A. Rosenfeld, A distance metric for multidimensional histograms, Comput. Vision Graph. Image Process. 32 (3) (1985) 328–336.
[11] H. Ling, K. Okada, An efficient earth mover's distance algorithm for robust histogram comparison, IEEE Trans. Pattern Anal. Mach. Intell. 29 (5) (2007) 840–853.
[12] S. Shirdhonkar, D.W. Jacobs, Approximate earth mover's distance in linear time, Proc. IEEE Conf. Comput. Vision Pattern Recogn. (2008) 1–8.
[13] J.Z. Wang, L. Jia, G. Wiederhold, Simplicity: semantics-sensitive integrated matching for picture libraries, IEEE Trans. Pattern Anal. Mach. Intell. 23 (9) (2001) 947–963.
[14] L. Jia, J.Z. Wang, Automatic linguistic indexing of pictures by a statistical modeling approach, IEEE Trans. Pattern Anal. Mach. Intell. 25 (9) (2003) 1075–1088.
[15] A. Khotanzad, O.J. Hernandez, Color image retrieval using multispectral random field texture model and color content features, Pattern Recogn. 36 (8) (2003) 1679–1694.
[16] E. Saykol, U. Güdükbay, Ö. Ulusoy, A histogram-based approach for object-based query-by-shape-and-color in image and video databases, Image Vision Comput. 23 (13) (2005) 1170–1180.
[17] Y. Rubner, J. Puzicha, C. Tomasi, J.M. Buhmann, Empirical evaluation of dissimilarity measures for color and texture, Comput. Vision Image Understand. 84 (1) (2001) 25–43.
[18] B.S. Manjunath, W.Y. Ma, Texture features for browsing and retrieval of image data, IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) (1996) 837–842.
[19] C.M. Pun, Rotation-invariant texture feature for image retrieval, Comput. Vision Image Understand. 89 (1) (2003) 24–43.
[20] P. Brodatz, Texture: A Photographic Album for Artists and Designers, Dover, New York, 1966.
[21] T. Hurtut, Y. Gousseau, F. Schmitt, Adaptive image retrieval based on the spatial organization of colors, Comput. Vision Image Understand. 112 (2) (2008) 101–113.