A weighted dominant color descriptor for content-based image retrieval

Ahmed Talib\textsuperscript{a,b,*}, Massudi Mahmuddin\textsuperscript{a}, Husniza Husni\textsuperscript{a}, Loay E. George\textsuperscript{c}

\textsuperscript{a}Computer Science Dept., School of Computing, University Utara Malaysia, 06010 Sintok, Kedah, Malaysia
\textsuperscript{b}IT Dept., Technical College of Management, Foundation of Technical Education, 10047 Bab Al-Muadham, Baghdad, Iraq
\textsuperscript{c}Computer Science Dept., College of Science, Baghdad University, 10071 Al-Jadriya, Baghdad, Iraq

\begin{abstract}
Color has been extensively used in the process of image retrieval. The dominant color descriptor (DCD) that was proposed by MPEG-7 is a famous case in point. It is based on compactly describing the prominent colors of an image or a region. However, this technique suffers from some shortcomings; especially with respect to object-based image retrieval. In this paper, a new semantic feature extracted from dominant colors is proposed. The newly proposed technique helps reduce the effect of image background on image matching decision where an object's colors receive much more focus. In addition, a modification to DCD-based similarity measure is also proposed. Experimental results demonstrate that the proposed descriptor with the similarity measure modification performs better than the existing descriptor in content-based image retrieval application. The proposed descriptor considers as step forward to the object-based image retrieval.
\end{abstract}

\section{1. Introduction}

Image retrieval has become one of the most famous research directions nowadays because it uses to search an image in archive, domain-specific, personal and web image databases. For retrieving color images from multimedia database, low level features and especially the color feature, have been widely used in this regard. This is because color represents the most distinguishable feature compared with other visual features, such as texture and shape. From perspective of feature extraction, color-based image descriptors can be divided into two categories: (i) Global descriptors that consider the whole image to obtain their features, there is no partitioning or pre-processing stage during feature extraction process. The resulted descriptors from this approach are simple and fast but it lacks to spatial color information and high discriminating power. The most famous example about this representation is global color histogram \textsuperscript{[1]}.

(ii) Local descriptors that obtain their features from local regions or partitions of image. This can be done by dividing the image into either fixed size or different size regions. The former type is called fixed partitioning-based approaches and they have more spatial information about colors in the image; an example of this approach is cell color histogram \textsuperscript{[2]}. The latter type is called segmentation-based approaches where the regions of image can be extracted by either segmentation or clustering methods. These descriptors usually have better accuracy than others but introduce more complexity of feature extraction process; examples of this approach are color-based clustering \textsuperscript{[3]} and dominant colors \textsuperscript{[4,5]}.

In addition to the global and local approaches to represent the image, there are other interesting methods that get more attention recently which are local invariant feature-based approaches. These approaches introduced features that are invariant to different image transformation such as translation, scaling, rotation and affine transformation. Salient edges and regions (saliency map) detection are part of these approaches and they are widely used in many computer vision applications including object recognition \textsuperscript{[6]} and image retrieval \textsuperscript{[7]}. Combination of different representation methods (at the feature or rank level) leads to improvement of image retrieval accuracy \textsuperscript{[8]}. Therefore, in this work, combination of global and local features (at feature level) is proposed to enhance color-based image retrieval.

In this respect, MPEG-7 Committee proposed many color, texture and shape descriptors to be used in image and video retrieval \textsuperscript{[9,10]}. Authors in \textsuperscript{[11–13]} maintain that human visual system first helps identify prominent colors in the image and second it processes any other details. The whole process resembles the way humans recognize image from its dominant colors without paying any attention to their distribution. MPEG-7's DCD (MP7DCD) provides compact and effective representations for colors in an image or region of interest \textsuperscript{[9]}. Recently, compactness property of dominant colors representation becomes more attractive for many researchers to reduce size of color descriptors from several hundred bins (histogram-based methods) into few colors (8 colors in MP7DCD) such as the works that have been done in \textsuperscript{[12,14–16]}. 

\begin{keywords}
Dominant color descriptor
MPEG-7
Object- and content-based image retrieval
Semantic feature
Similarity measures
Salient object detection
Background dominance problem
Linear Block Algorithm
\end{keywords}
This compactness is mandatory in specific applications such as web-based image retrieval [17].

In the compact and simplest form, DCD contains the following form:

\[ \text{DCD}(I) = \{(C_i, Pi), i = 1, \ldots, N\} \]

where \( N \) is the number of dominant colors in an image \( I \), \( C_i \) represents the 3-D value of the dominant colors and \( Pi \) represents the percentage of each DC. However, since MP7DCD have certain drawbacks, it has been undergone some enhancements [3, 5]. Most of the previously conducted studies were dedicated to improve the DC extraction process. This is due to the fact that MPEG-7 uses Generalized Lloyd Algorithm (GLA) [18] for color quantization. The latter is characterized by several limitations: (1) it is a time-consuming method; (2) the number of its clusters must be predefined before starting clustering process; and (3) using different initial cluster seeds lead to different results. Hence, a new quantization method, Linear Block Algorithm (LBA), is proposed to speed up the DC extraction [11]. Moreover in [13], code book is proposed for color quantization to reduce the range of colors in an image. According to the researcher’s point of view, the aforementioned enhancement methods of DCD are mainly used to speed up the process. MP7DCD is accurate but it lacks certain semantic information. That is, the prominent colors and their percentages may only lead to retrieve many dissimilar images that share the same biggest DC. Usually, the dissimilarity occurs when the background color of an image has the largest percentage. In other words, most of the images retrieved by DCD contain similar background colors if their percentages are high. However, they differ among each other with respect to the semantic of the color that has the largest percentage (whether it is background or object). This largest percentage of color will make no significant consideration to the other colors of low percentages. As it is the case with the way humans look at the object located at the center of an image, the current work is dedicated to propose a method that gives a weight to each dominant color, depending on its belonging to the salient object or to the borders (background) of an image. Specifying the important colors (object’s colors) from these total dominant colors will enhance image retrieval accuracy. Additionally, this accuracy improvement will effect positively on the methods that can be integrated with dominant colors such as color structure [15, 16] and color correlogram [14, 19]. A new attempt is introduced to give more importance to the object’s DCs of the image in [20]. It is assumed that the lighter color in the image represents the object color and the darker color is the background but this assumption is uncertain for various image contents. Moreover, it depends only on the largest percentage color of the object (only one color) whereas the object may contain many small percentage colors. Additionally, authors in [21, 22] reported that the object in the image may be small or large and almost its percentage in the image is 25%. In other words, if this assumption can be guaranteed, then the existing DCDs that depend on the largest DC can be considered as perfect methods because it is compatible with this assumption.

Besides, MP7DCD’s quadratic similarity measure (QSM) that is used by Deng et al. [5] and Yamada et al. [9] has some drawbacks. In [23, 24], one can see the first simple changes that were made to improve QSM. Authors in [25] also propose a palette histogram similarity measure to solve QSM problem. Moreover in [11], a new similarity measure was proposed to achieve a good performance compared with QSM and all the aforementioned modifications [23–25]. In the present work, a modification will be applied to all the above dis-similarity measures to improve their performance. This modification embeds using mutual color ratio (MCR), which alleviates their dependencies on the biggest DC. In this paper, a semantic feature is added to the DCD to improve its accuracy in an object-based image retrieval application and it is considered as feature level-based solution to the background dominance problem. In addition, MCR is introduced as similarity measure level-based solution to the same problem. Many experiments have been conducted to ensure the efficiency of the proposed method with respect to four quantitative metrics (ARR, ANMRR, MAP and P(10)).

The paper is organized in the following way. Section 2 is concerned with explicating DCDs and the similarity measures. Section 3 is mainly concerned with the proposed semantic DCD and the newly proposed modification that helps improves the similarity measure. Section 4 illustrates the extensive experiment that contains visual and quantitative results. Finally comes the conclusion in Section 5.

2. Related works

Many researches have been done with respect to content-based image retrieval (CBIR). These include the following: Visual-SEEK [26], QBIC [27], Photobook [28], Image-Rover [29], and others. In these studies, several visual (low-level) features, such as color, texture and shape have been used. Out of these low level features, one can get some semantic information from the processed images that used in CBIR. As a case in point is color information, which represents a basic cue for object and scene recognition [30]. Moreover, these studies have varied in their usage of color descriptors [12]. Some, for instance, used global color descriptors whereas others used spatial color descriptors. The former is used to measure the similarity between two images by taking into account both the colors and their percentages in the images, such as color histogram [31–34] and the dominant colors [49–11, 13, 16, 35, 36]. The latter type, on the other flip, measures the similarity between the two images by taking into consideration both the existing colors and their distributions or arrangements in the image, such as color correlogram [37, 38].

Color histogram that is proposed by Swain and Ballard [1] has been extensively used as global color descriptors. It is used to solve translation and rotation invariant problems. Besides, it is characterized by being easily implemented and accurate; particularly with small database size. Accordingly, many enhancements in histogram-based approaches have been achieved as reported in [31–34]. However, such approaches have several drawbacks; the basic one is its dependence on a static quantization method. That is why, it is used to reduce color space to make storage and time more reasonable. Static quantization methods suffer from the low discrimination power. This is because many similar colors may be set to different bins; a matter that makes the similarity measure (\( L_1, L_2 \) or histogram intersection) between the two histogram inefficient.

To solve the static quantization problem in the color histogram, a quadratic similarity distance [31] was proposed. The proposed method is set to compute the similarity between the two images. One can notice that each one of these images has different histogram bins. That is, if \( X \) is the color histogram of the first image with \( N \) bins and \( Y \) is the color histogram of the second image with \( M \) bins, one can write the histogram of both images as in the following form:

\[ X = \{(C_1, W_1^1), (C_2, W_2^1), \ldots, (C_N, W_N^1)\} \quad \text{and} \quad Y = \{(C_1, W_1^2), (C_2, W_2^2), \ldots, (C_M, W_M^2)\} \]

where \( C \) represents the color value and \( W \) is the weight (frequency or percentage) of each color in the image. So, the quadratic distance (\( D_q \)) between these two images can be computed as follows:

\[
D_q(X, Y) = (X - Y)^T A (X - Y)
\]

\[
= \sum_{i=1}^{N} \sum_{j=1}^{M} a_{ij}(W_i^1 - W_j^1)(W_i^2 - W_j^2),
\]

where \( A = [a_{ij}] \) is the color similarity matrix between the bins \( C_i \) and \( C_j \). It has also been noticed that the metric depends on the color similarity of the bins; however, it has some tolerance to the difference between the colors [39]. In [25], the authors show that the quadratic distance has some limitations. For instance, it does not match
human's color perception. Besides, it gives incorrect rank to the retrieved images in some cases.

Due to the aforementioned limitations of histogram and to the fact that humans cannot perceive more than 8 colors [13], extracting the dominant colors (DCs) only from the image represents the best solution in this regard. Consequently, several DC descriptors have been proposed; as cases in point are the following: MPEG-7 DCD [9,4,10,11,13,16,35,36]. The DC that is extracted via using a dynamic quantization that is compact and efficient compared to the other global image descriptors. This is because such a DC requires less time and storage consumption compared to the spatial color descriptors. Hence, the present paper is dedicated to introducing a method that enhances DCD to be more semantic and reduces the gap between it and the spatial color descriptors. However, the method shows no significant increase in the storage but requires more time.

Local invariant features are widely used recently for solving a wide variety of problems, from image matching and the recognition of specific objects to the recognition of object categories [40]. These features characterized by their invariance to image transformation such as translation, rotation, affine and others. Authors in [40] explain that they are widely used not for locality nor for invariance features but rather for their ability to shift to the form that the researcher prefer to use them in. Recently, representing image content in a robust and flexible way is focused by many researches. This is by using of local features effectively to compensate of using semantic-level segmentation where separating object (s) from the background is a very hard problem. Actually, this problem mostly cannot be solved by using low-level features only.

Local invariant features-based method is a task consists of higher-level processing steps to extract relevant information, or at least to be robust to the outliers, in the image. This new way of looking at local features has opened up a whole new range of applications, and moves many steps closer towards cognitive-level image understanding. There are many local features detectors include corner, blob and region detectors that use different features such as contour-based, edge-based, intensity-based, biologically plausible-based, color-based or model-based methods. Salient features (regions or edges) represent one of the outcomes of local features detectors.

Saliency idea has been used in many computer vision algorithms. The early approach of using edge detectors was to extract object descriptions where it depends on the idea that the edges are more significant than other parts of the image. More explicit uses of saliency can be divided into those that concentrate on low-level local features [41], and those that compute salient groupings of low-level features [42]; though some approaches operate at both levels [43]. These salient points are the points on the object which are almost unique. Many methods are used to extract saliency features or map from image such as Achanta et al. [44,45], Cheng et al. [46] and Goferman et al. [47] to name a few. Although the saliency idea originated from local features (regions or edges) but there are many attempt to extract it from global contrast of image or combine between them such as Achanta et al. [45], Cheng et al. [46] and Zhai and Shah [48].

In this work, two of the salient object detection algorithms are used. One for natural images and another simple one for cartoon images because the latter type of images is characterized by the object of cartoons are surrounded by bold dark contours [49,50]. This salient object detection algorithm is integrated with another proposed algorithm, which assumes that the object mostly located in the image's center, to give weight to each dominant color according its spatial location whether it is belong to the salient object or background (border).

In other hand, similar to the dynamic quantization-based histogram, MP7DCD [9] also uses QSM with some modification (as presents in Eq. (7)) to measure the dissimilarity between the query image and database images. However, QSM is not void of serious drawbacks. For instance, it does not match human color perception [11,25]. Therefore, some extensions to QSM have been proposed as shown below:

(1) Ma et al. [23] proposes the similarity measure as follows:

$$D_{Ma}(I_1, I_2) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (I_2(C_i, C_j) * |P_i, P_j|),$$

\[3\]

where $L_2$ is the Euclidian distance between the colors $C_i$ and $C_j$; $P$ represents the percentages of the DC $C_i$ and $C_j$; $M$ and $N$ represent the number of DCs in image $I_1$ and $I_2$, respectively.

(2) Mojsilovic et al. [24] proposes a similarity measure as stated below:

$$D_{Mojsilovic}(I_1, I_2) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (I_2(C_i, C_j) + |P_i, P_j|).$$

\[4\]

(3) Po and Wong [25] propose a merging palette histogram for similarity measure (MPHSM). Firstly, they merged similar DCs of the two images. This step is done if $L_2$ distance between them is smaller than certain threshold. This step helps produce a common palette (that has $N_m$ colors) of the two images (with $N_1$ colors and $N_2$ colors respectively, where $N_m \leq N_1 + N_2$). Secondly, two new DC histograms of the two image palettes (each has $N_m$ colors) are generated based on the new common palette. The colors in the latter palette that have a distance smaller than a certain threshold to the original DC will get the same frequency of original bin (DC); otherwise, a new bin will get zero. Thirdly, a conventional histogram intersection method is applied to the new (equal bins) histogram to find their dissimilarity. To illustrate histogram intersection considers the following equation:

$$D_{MPHSM}(I_1, I_2) = \sum_{i=0}^{N_m-1} \min(P_{I_1}, P_{I_2}).$$

\[5\]

where $P_{I_1}$ and $P_{I_2}$ represent the percentages of DC in the two images $I_1$ and $I_2$, respectively.

(4) Yang et al. [11] propose a new similarity measure that simulates human color perception (as present in Eqs. (10)–(12)). The new similarity measure is proved to be better than the aforementioned similarity measures and is closer to the human perception. All the above similarity measures and their originated QSM suffer from a common problem. The latter is represented by their dependencies on the biggest (largest percentage) DCs in the image during the process of retrieving the images. To reduce these dependencies, mutual color ratio is proposed (as similarity measure level-based solution) and added to their similarity measure to improve their results.

3. Proposed descriptor and modification to similarity measure

Although extracting proper DCs can solve the problems of color histogram; however, DCD (colors values and their percentages in the image) still lacks a good description about the object in the image (same histogram problem). Such a thing happens particularly when the background color has the largest percentage among the dominant colors of the images. That is to say, it lacks the semantic information in its representation. In other words, most of the histogram-based and DC-based image retrieval approaches cannot apply to object recognition problem [51]. This is because the object occupies a portion (small or large) in an image [22]. So, the process
of retrieving the images depends basically on the color that occupies the largest area in the image (i.e., the longest bin in color histogram or DC that has the largest percentage of the image area).

To obtain object DCs, two methods that depend on two assumptions can be applied. Firstly, some studies show that the object that is located at the center of an image and that its size is approximately 25% only [21,52]. They also showed that the background color will be distributed in the corners and borders of the images [21,52]. On the other flip, the photographer tends to locate the object in the middle of the picture when he takes a photo [21]. Therefore, this hypothesis is taken into account to extract object DCs where the DC that appear on the image border mostly it is not belonging to the object. An explanation of this method is detailed in Section 3.1 and its outcome called as Border Weight of DC (BW-DC).

Secondly, the previous assumption is hardly to consider alone to extract semantic feature about the object; instead it can be integrated with the second assumption, which is salient object detection method to be as complementary part. There are two Salient Object Detection (SOD) methods are used in this work; one for natural images, Global Contrast based Salient Region Detection (GC-SRD) that proposed by Cheng [43] because its effectiveness compared with other saliency methods. Second salient object detection is used for cartoon images, due to its different characteristics, which is Laplacian of Gaussian (LoG) filter with flood fill algorithm that proposed by Yu and Seah [53] and Yu et al. [54].

Integration of the both methods (BW-DC and SOD) is necessary to detect the object in the image because the first assumption of being the object is almost located at the center and do not touch the border is not guaranteed in all images. In the other side, although the effectiveness of the SOD methods that suggest to use here but they have are some limitations such as it is effective for single object only (for GC-SRD) or it is not working with very complicated background (for LoG).

Therefore, this paper extracts importance of each DC in the image and adding it to DCD. This will help produce the semantic DCD, which is called the weighted dominant color descriptor (WDCD), for color- and object-based image retrieval.

### 3.1. Proposed weighted dominant color descriptor

The previous section is firstly illustrating the presumption that assumes an object is located at the center of the image while the background color is distributed to corners and boundaries. Such a step embeds extracting the dominant colors with their percentages from the image via using MP7DCD or fast LBA. Then, the weight of each DC (resulted from MP7DCD or LBA) is computed depending on its location in the image, where the weight of DC equal to the frequency of it on the border, hence it is called as border weight (BW). The color that has high frequency at the image borders, it will obtain a high weight. This mean it is considered the background color. In addition, each color that has a low frequency in the border will obtain a low border weight (consider as object color), as shows in Fig. 1.

One can notice that the way the weight is given to each DC depends on the percentage of each color\(^1\) on the image border. Each DC color appears with a high frequency in the border. It represents the background color and so gets a high Border Weight (get BW = 1 if the image has one background DC, its frequency = border length). Moreover, the DC that does not appear in the border (frequency = 0) gets a low weight (get BW = 0). It represents an object in an image (as colors White, Brown and Yellow in Fig. 1).

\(^{1}\) For interpretation of color in Fig. 1, the reader is referred to the web version of this article.

Therefore, the border weight of each DC can be computed by taking into account the frequencies on the image border, as shown in the following formula:

\[
\text{BorderWeight}_{DC} = \frac{\text{Freq}_{DC} (\text{Border})}{\text{BorderLength}}
\] (6a)

From Eq. (6a), one can compute the weight of all DCs in Fig. 1, as stated below:

\[
\begin{align*}
\text{BorderWeight}_{\text{blue}} &= \frac{\text{Freq}_{\text{blue}}}{\text{BorderLength}} = \frac{325}{500} = 0.65, \\
\text{BorderWeight}_{\text{green}} &= \frac{175}{500} = 0.35, \\
\text{BorderWeight}_{\text{white}} &= \text{Weight}_{\text{brown}} = \text{Weight}_{\text{yellow}} = \frac{0}{500} = 0.
\end{align*}
\]

According to the above formula, one can notice that the DCs, which represent the background, get higher BW than the DCs which represent the object. This first step state helps reduce the effect of background in similarity decision and gives some semantic information to the DCD by giving more importance to the object.

Nevertheless, there are some cases that conflict and refute the previous assumption; they can be explained as follow. First case, some images have a large object that may touch the border of an image (as present in Fig. 2a). Second case, the background color has the same object color (as present in Fig. 2b); this will remove the object from consideration as well as the background. Third case, there is a thin line surround the image (as present in Fig. 2c); this will consider false background color. Therefore, Salient object detection method can be used to determine and solve the conflicted cases and complement the proposed border weight method.

Computing the border weight of DCs of an image can be expressed using the following algorithm:

---

**Fig. 1.** The way an image gives weight to each DC depending on whether it is an object or background. Border Weight (blue) = 0.65, BW (green) = 0.35, BW (white) = 0, BW (brown) = 0 and BW (yellow) = 0. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
salient object is illustrated in Fig. 3. The complement and second part of WCD is detecting salient object weights (SOW) and DC’s percentages (weights) in the DCD’s resulted image (DCW). Firstly for each of the three input weights; there are two symbols, either “Large” (L) or “Small” (S) must be used to describe them depending on specific thresholds. The thresholds that extracted experimentally and can be considered in this work are 0.05, 0.10, and 0.10 for SOW, BW, and original DC weights or percentages (DCW) respectively. Additionally, description (either “L” or “S”) of all these weights can be denoted as DSOW, DBW, and DDW respectively as presented in Table 1. Since there are three weights, therefore there are 8 possible cases of their combinations. Before start of describing these cases, it is necessary to mention the indication of each used weight where SOW of certain DC with symbol “L” indicate that DC is belonging to the object with high percentage otherwise it belongs to the object with low percentage. BW with symbol “L” also indicate that DC belongs to border (background) with high percentage otherwise it belongs to the border with low percentage. Finally, DCW with symbol “L” indicate that this DC has large percentage in the image; otherwise it refers that the DC percentage is small in the image. Final weights of DCs can be computed as in Table 1.

Final weight of DCs represents an importance of this DC in the image. That mean, if it is high (≈1) then DC is belonging to the object and must be considered while if it is low (≈0) then it is belonging to background and must be removed from consideration when computing similarity measure. Other values (0 < Final Weight Value < 1) indicate the importance of this DC, as the weight value was high, this mean the DC is important to consider. Weight Value < 1) indicate the importance of this DC, as the weight value was high, this mean the DC is important to consider. Other values (0 < Final Weight Value < 1) indicate the importance of this DC, as the weight value was high, this mean the DC is important to consider. For Table 1, one can notice that there is no zero value is set to the DC weight in all cases. This is because; zero value will remove the DC completely from consideration and this color may have some importance but a mistake in computing its BW or SOW is occurred. Hence, zero weight value is avoided and it may be computed implicitly through the cases of Table 1 (for example: image in Fig. 2c, yellow color will get weight equal to 0 because it matches to the case 6).

Case 1 (“L” for three input weights “LLL”) indicate to the DC is belong to the object and background and it has large percentage in the image. This DC makes the decision to be confused to decide it belongs to the object or background. In this case, the maximum among these weights of DC is selected to represent its importance in the image because it may be a large object and cover large area in the image and sure its weight must be select carefully. BW represents background weight of certain DC, so it cannot compares with object weight (SOW), thus its reverse (1-BW) is considered to represent its object weight regarding to border. Case 2 (“LLS”) is same to the case 1 but DC has small percentage in the image, hence the maximum between two object weights (SOW, 1-BW) will be considered as final weight. Case 3 and 4 (“LSL” and “LSS”)
refers to the DC appears obviously in the extracted salient object and disappear from the border, hence it is confirmed as object color and must obtain full importance by giving it the value “1” as final weight. Case 5 and 6 (“SLL and SLS”) refer to the DC appears with large percentage in the background and does not appear in the object, hence it considered as background color and remove its effect from similarity measure by giving it low weight (1-BW). DC in case 5 represents background color that has large percentage in the background.

### Table 1

<table>
<thead>
<tr>
<th>Case no.</th>
<th>DSOW</th>
<th>DBW</th>
<th>DDCW</th>
<th>Final DC weight</th>
<th>Case description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>L</td>
<td>L</td>
<td>L</td>
<td>Max (SOW, 1-BW, DCW)</td>
<td>Confused DC, it is belonging to Object and Background with high percentage (e.g., Fig. 2a and b)</td>
</tr>
<tr>
<td>2</td>
<td>L</td>
<td>L</td>
<td>S</td>
<td>Max (SOW, 1-BW)</td>
<td>Same above but it has small percentage in the image</td>
</tr>
<tr>
<td>3</td>
<td>L</td>
<td>S</td>
<td>L</td>
<td>1</td>
<td>Confirmed DC, it represents big Object</td>
</tr>
<tr>
<td>4</td>
<td>L</td>
<td>S</td>
<td>S</td>
<td>1</td>
<td>Confirmed DC, it represents small Object</td>
</tr>
<tr>
<td>5</td>
<td>S</td>
<td>L</td>
<td>L</td>
<td>1-BW</td>
<td>It is Background DC of large percentage, it must obtain low weight</td>
</tr>
<tr>
<td>6</td>
<td>S</td>
<td>L</td>
<td>S</td>
<td>1-BW</td>
<td>It may be a thin line around the image, it must be ignored (e.g., Fig. 2c)</td>
</tr>
<tr>
<td>7</td>
<td>S</td>
<td>S</td>
<td>L</td>
<td>DCW</td>
<td>Confused DC, hence we consider its percentage in the image</td>
</tr>
<tr>
<td>8</td>
<td>S</td>
<td>S</td>
<td>S</td>
<td>DCW</td>
<td>Same above but it is a small object</td>
</tr>
</tbody>
</table>

Fig. 3. Two images examples about salient object extraction steps.

Fig. 4. Two images examples to illustrate the inaccuracy of the SOD methods (GC-SRD as in (a) and LoG as in (b)).
image while DC in case 6 represents just thin line surrounding the image only and it should be ignored as shown in the Fig. 2c. Case 7 ("SSS") represents that the DC does not appear in the object area nor in the background area but it has large percentage in the image. Removing it from consideration may be lead to big mistake because it may be missing object; thus its percentage in the image is considered as final weight. Case 8 ("SSS") refers to the same previous case but DC is a small percentage; hence, giving it low weight is suitable.

3.2. Similarity measure of the proposed DCD

In [9], MPEG-7 used quadratic distance to compute the dissimilarity measures between the two images, as shown in following formula:

$$D_0(I_1, I_2) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} p_i p_j$$

where $I_1$ and $I_2$ represent the two images that their similarity is required to be measured. After the process of extraction of the DCs of the images, the latter can represent features, as the following: $F_1 = \{(C_i, P_i) | i = 0, \ldots, N - 1\}$ represents features of $I_1$ that has $N$ dominant colors; $F_2 = \{(C_j, P_j) | j = 0, \ldots, M - 1\}$ represents features of $I_2$ that has $M$ dominant colors, $C$ and $P$ represent color value and the percentage of each DC in the image, respectively. Finally, $a_{ij}$ represents the similarity coefficient between the colors $C_i$ and $C_j$. It can be computed using the following equation:

$$a_{ij} = \begin{cases} 1 \frac{d_{E, max}}{d_{E}} & \text{if } d_{ij} \leq T_h, \\ 0 & \text{if } d_{ij} > T_h, \end{cases}$$

where $d_{ij}$ represents Euclidean distance between $C_i$ and $C_j$, the abbreviation $C$ represents the 3-D color values (in CIE-Luv color space), which can be computed as follows:

$$d_{ij} = \sqrt{(C_i - C_j)^2 + (C_i' - C_j')^2 + (C_i'' - C_j'')^2}.$$ 

The threshold $T_h$ represents the maximum distance whereby the two colors are considered similar, and $d_{E, max} = x T_h$, $x = 1$ or 1.2. The latter state assumes that the maximum distance between the two colors is slightly greater than color threshold. As it is stated previously, this distance has serious drawbacks; accordingly, it does not satisfy human perception [11,25]. Therefore, Yang in [11] proposes a new efficient similarity measure for DC as shown in the following equations:

$$S_{ij} = [1 - |p_i(i) - p_j(j)|] \times \min(p_i(i), p_j(j)),$$

$$SIM^{Yang}(I_1, I_2) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} S_{ij},$$

$$D_{yang}(I_1, I_2) = 1 - SIM^{Yang}(I_1, I_2).$$

In Eq. (10), $p$ represents the percentage of DC in the image, $S_{ij}$ refers to the similarity between color percentages. On the other hand, $a_{ij}$ in Eq. (11), represents color similarity between the two colors $C_i$ and $C_j$ as indicated in Eq. (8). In Eq. (11), $SIM^{Yang}(I_1, I_2)$ represents the similarity ratio of the two images. Finally, to measure the dissimilarity between the two images, one can use Eq. (12). Yang in [3] pinpointed that such a measure resembles the mechanism of human perception of colors. Besides, it helps overcome problems of quadratic distance and proves its efficiency over the two improvements of quadratic distance proposed by Ma et al. [23] and Mojsilovic et al. [24].

Therefore, the present similarity measure conducted in the present paper depends on $D_{yang}$ dis-similarity distance. However, it modified to be able to obtain the proposed semantic feature, which is the weight of DCs that based on their belonging to the object or background. The adapted similarity measure can be formulated using the following equations:

$$W_{ij} = \min(W_i, W_j),$$

$$SIM^{W}(I_1, I_2) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij} S_{ij} W_{ij},$$

where $W_{ij}$ in Eq. (13) represents the intersection of weights of the two DCs, $i$ and $j$, which in return represent the smaller weights. That is, it reduces the importance of the color if it represents the background color in either the first or second image. As for the DCs weights of $W_i$ and $W_j$, they can be computed by the Table 1 mentioned above.

The $SIM^{Yang}(I_1, I_2)$ in Eq. (11) helps compute the similarity between the two images. It is further increased whenever the images are more similar (in term color value $a_{ij}$ and percentages $S_{ij}$).

![Fig. 5](image-url) Carton image example to show the effectiveness of the proposed WCD in object-based image retrieval; where (A) represents image $I_1$; (B) represents image $I_2$; and (C) represents image $I_3$. 
Eq. (14), the DC’s weight is multiplied with the formula to decrease the consideration of the background DC by multiplying them by a low weight or even 0. Or, it keeps them as they are if they are an object DC by multiplying the formula by 1 or high weight (as resulted from Table 1). Finally, the dissimilarity measure that depends on the DCs weight (DW) can be computed as follows:

$$DW(I_1, I_2) = 1 - SIM^{W}(I_1, I_2).$$

To illustrate the effectiveness of the proposed method in object-based image retrieval compared with that of Yang [11] and his dissimilarity distance $D_{yang}$, a cartoon image example is presented, as shown in Fig. 5.

As it is shown previously, $SIM^{yang}$ or $W(I_1, I_2)$ is used to measure the similarity between the two images. On the other flip, $D_{yang}$ or $W$ is dissimilarity measure that is used for the retrieval purpose. Such a step is done by inverting the result of $SIM$, by $D(I_1, I_2) = 1 - SIM(I_1, I_2)$ to make the distance small in all the similar images and large for all the dissimilar images. Therefore, SIM can only be computed, as present in Tables 2 and 3, to show the similarity among the cartoon images that presented in Fig. 5.

From Table 3, one can notice that $SIM^{yang}$ depends basically on the color that has the largest percentages regardless whether it represents the background color or the object color. Therefore, the similarity value of image1 (Barney with White background) and image2 (Goofy with White background) is larger than that of image1 (Barney with White background) with image3 (Barney with Gray background). Using the adapted method, one can notice that its $SIM^{W}$ helps remove the effect of background on similarity value. The reduction is based on the object color and is obtained by multiplying the background color by 0, as shown in the column “white” in Table 3. Additionally, the proposed method depends on the largest colors of object and alleviates effect the small percentage colors such as “Yellow” and “Black”.

3.3. Modification of similarity measure

To improve the retrieved results, one can modify the similarity measure by adding the ratio of the mutual colors between the two compared images. This modification has a number of advantages, such as: (1) it alleviates the major dependencies of the similarity measure on the largest percentage DC. This modification will take other colors into consideration even though these colors have small percentages; (2) it enhances the rank of the most similar relevant images by shifting the images that have similar colors up and shifting the others down. This modification will enhance retrieving the images; especially, the image classes that have more than one related colors, such as the images of beach that are frequently come with the sky and sand regardless the colors of objects that exist on the beach (people or trees). From another perspective, this modification will conflict with the proposed WDCD descriptor because the latter concerns with object only regardless its background. As example, the category of elephant images in Corel dataset come in two different backgrounds first is sky and water whereas the second is grass and tree. Hence, MCR will decrease the similarity between two images that have different backgrounds even they have the same object. Therefore, some changes will be done on this ratio to suit the proposed WDCD. This modification; accordingly, enhances the result of MP7DCD [9], Yang [11] and the proposed descriptor. The Mutual Color Ratio (MCR) of the two images DCs can be computed via the following algorithm:

**Algorithm 2. Mutual Color Ratio (DC1, FW1, N, DC2, FW2, M)**

- Set Mutual Colors counter to zero
  
  $MC \leftarrow 0.$

- Find mutual colors by pass on all DCs of two images
  
  - [for each DC in the Images]

    - $\forall \ dc_i \in DC1 \ i = 0, \ldots, N - 1,$
    - $\forall \ dc_j \in DC2 \ j = 0, \ldots, M - 1.$

  - [count number of similar colors between two images]

    $$\text{if}(\text{distance}(dc_i, dc_j) < \text{Th}_2)$$
    $$MC + ;$$
    $$\text{or} \ MC = MC + \text{min}(FW1, FW2) ;/|\text{for WDCD}|$$

- Compute ratio of mutual colors of two images

  $$\text{MCR} \leftarrow \frac{MC}{\text{Max}(M, N)}.$$  

- Return

FW1 and FW2 represent final weights of the two compared images that will be used for computing MCR for the proposed WDCD. MCR for WDCD considers weights of colors instead of “1” during calculation of its value to overcome the aforementioned problem with WDCD. The similarity measure of the proposed approach (Eq. (14)) will be modified by adding MCR as shown below:
also be modified as illustrated below:

Each of these classes contains quantitative metrics descriptors, one. Besides, to measure the performance of the competing parameter, which are used for comparison with the proposed descriptors together with the proposed similarity measure modification.

4.1. Experimental setup

This section is dedicated to identify some setup parameters that will be used in the experiments of the current work. These parameters are image datasets that are used for the purpose of testifying and comparing the performance of the proposed descriptor together with the candidate descriptors. The latter is the second parameter, which are used for comparison with the proposed one. Besides, to measure the performance of the competing descriptors, quantitative metrics will be used as the third parameter for measuring the performance of candidate and proposed descriptors as well as the proposed similarity measure modification (MCR).

4.1.1. Image datasets

Evaluating the proposed semantic DCD (WDCD) will be conducted on three datasets: (1) the well-known Corel-1 K dataset that contains 1000 images. This dataset consists of 10 classes (elephants, buses, flowers, dinosaurs, and others). Each of these classes has 100 images; (2) the well-known Caltech-101 dataset that contains 101 classes. In this dataset, 26 classes only are selected; each class characterized by its images’ object have the same color to show effectiveness of the proposed WDCD in color-based object image retrieval. The selected classes are Flamingo, Leopards, Starfish and others; minimum number of images in each class is 35 and the total number of images is 2562 images; (3) Cartoon-5 K dataset that contains 5128 images collected from the web. Cartoon dataset has 85 classes (cartoon characters); each one has at least 50 images. Selection of cartoon dataset is intended to show effectiveness of WDCD. This is because; cartoon characters often appear in the same colors in most cartoon images [55] (that fit the objective of this paper) as well as this type of image does not suffer from illumination variation. Some samples from Cartoon dataset are presented in Fig. 6.

4.1.2. Competing descriptors

The descriptors that are selected to be compared with the proposed WDCD are MPEG-7 DCD [9] and LBA DCD [11]. This is because the former represents the original DC descriptor whereas the latter is the best DC descriptor so far. In this context, Yang [11] shows that the adapted descriptor surpasses the other DC descriptors [9,23–25] in both accuracy and time. Hence, there is no need to compare with them.

In another side, the comparison with other Object-Based Image Retrieval (OBIR) methods is unfair because these methods combine other features with color, such as texture and shape, to get their results. Our proposed descriptor is just a step forward in object-based image retrieval. It needs to combine with other features, such as spatial color relations such as Kiranyaz et al. [14] and Wong et al. [15] or shape such as Khan et al. [55], to be suitable for OBIR. Moreover, comparison with some color-based object recognition

<table>
<thead>
<tr>
<th>White</th>
<th>Green</th>
<th>Pink</th>
<th>Black</th>
<th>Yellow</th>
<th>Peach/Puff, Orange, Gray</th>
<th>Overall similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIM\textsuperscript{WDCD}(I_1, I_2)</td>
<td>[1–0.55 + 0.01]</td>
<td>[1–0.09 + 0.01]</td>
<td>0</td>
<td>[1–0.09 + 0.02]</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SIM\textsuperscript{LBA}(I_1, I_2)</td>
<td>[1–0.55 + 0.01]</td>
<td>[1–0.09 + 0.01]</td>
<td>0</td>
<td>[1–0.09 + 0.02]</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

The similarity measure of MP7DCD (Eq. (7)) and Yang (Eq. (11)) can also be modified as illustrated below:

\[
D_{Q}(I_1, I_2) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{i,j} S_{i,j} W_{i,j} (1 + MCR).
\]

\[
SIM\textsuperscript{WDCD}(I_1, I_2) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{i,j} S_{i,j} MCR.
\]

4. Experimental results

Fig. 6. Samples from Cartoon-5 K Dataset.
techniques such as Geusebroek [56] that depends on color invariant feature is also out of the scope of this paper.

4.1.3. Performance measure metrics

Four quantitative performance measure metrics will be utilized to measure the performance of the proposed semantic descriptor and the other DCD that are candidate for the purpose of comparison. Moreover, the new similarity measure modification (MCR) will also be checked.

The first metric is the Average Retrieval Rate (ARR) [4]. It can be computed via the following equation:

\[ ARR = \frac{1}{N_q} \sum_{q=1}^{N_q} RR(q) \leq 1, \]  

where \( N_q \) represents the number of queries that are used for the purpose of verifying the descriptor in certain dataset. On the other hand, the abbreviation, RR represents the retrieval rate of a single query (it is similar to Recall metric). It can be calculated using the following equation:

\[ RR(q) = \frac{N_c(q)}{N_q(q)} \leq 1. \]

\( N_c(q) \) denotes the number of ground truth images (database relevant images) of a query \( q \). \( N_q(q) \) indicates the number of the relevant images found in the first \( k \) of \( N_q(q) \) images. The latter is resulted from the retrieving images of the query, whereas \( k \) should be more than or equal to 1. In the present experiments, \( k = 1 \) and the number of Query \( N_q(q) \) will be set to more than 1% of the database size. This is because many researches such as Manjunath et al. [4], Chen et al. [57] and Zhang [58] states that the number of the queries must be at least 1 percent of the database size. Any high ARR value \((\approx 1)\) will result in a good retrieval rate whereas the low ARR value \((\approx 0)\) indicates a bad retrieval rate.

The second used metric is Average Normalized Modified Retrieval Rate (ANMRR) [4,12]. It considers as one of the most accurate metrics used in CBIR. This is because it combines many conventional metrics, which are hit-miss counters, precision-recall and ranking information. Besides, it represents all of them in one value. To compute ANMRR, one can use the following equations:

\[ ANMRR = \frac{\sum_{q=1}^{N_q} NMRR(q)}{N_q} \leq 1, \]  

\[ NMRR(q) = 2 AVR(q) - N_c(q) - 1 \]  

\[ AVR(q) = \frac{\sum_{k=1}^{N_c(q)} R(k)}{N_c(q)}, \]

where \( R(k) \) is the rank of each ground truth images in the query result window \( W \) of the size \( 2 + N_c(q) \). Any non-relevant images appear within the window \( W \) will get \( R(k) = W + 1 \). The best value of NMRR \( (q) \) is 0; it represents all ground truth images are found in the window \( W \) of the query results. The worst case for NMRR \( (q) \) is 1; when there are no relevant images retrieved. Therefore, the lower value of ANMRR is better than the higher value.

The third metric is mean average precision (MAP), this metric is one of the widely used metrics in CBIR and also it is a compromise between precision and recall in a single metric [17,59]. Therefore, it became one of the leading performance evaluation metrics in the ad hoc retrieval systems [59]. It can be computed as follows:

\[ MAP = \frac{1}{Q} \sum_{q=1}^{Q} AP(q). \]

\( Q \) represent number of queries, \( AP \) denotes to average precision that represent arithmetic mean of the precision values of all relevant images. \( AP \) can be computed as explain below:

\[ AP = \frac{1}{r} \sum_{r=1}^{r} P_r. \]

\( P_r \) represents precision value for all relevant images \( r \), where precision is computed after retrieving of each relevant image. Precision \( (P) \) is a famous term that can be computed as below:

\[ P = \frac{\text{No. of relevant images in the Query}}{\text{No. of total retrieved images in the Query}}. \]

Fourth metric is \( P(10) \); it is a precision value of the first 10 retrieved images by specific query. It can be computed using Eq. (26) but need to change the denominator by 10. It widely used metric in web-based image retrieval [17]. This is because, the user often tends to see the result of his query in the first page, and he prefers to reformulate the query instead of checking the second page of result. The best value for metrics \( P(10) \) and MAP is close to 1, that mean the relevant images is retrieved in good rank. MAP differs from ANMRR that it measures the retrieval accuracy to all relevant images in the database to specific query while ANMRR measure the retrieval accuracy within specific window \( W \) only.

4.2. Retrieval performance

Retrieval performance of the competing descriptors in the above specified datasets can be measured using the four aforementioned metrics (ARR, ANMRR, MAP, and \( P(10) \)). Diversity of queries is very important to ensure fair and honest results [59], thus the evaluation queries are selected from all classes of the database.

4.2.1. Retrieval performance of Corel-1 K dataset

To illustrate more about the visual comparison of the candidate DCDs on Corel-1 K dataset refer to Fig. 7. Additionally, the four evaluation metrics are computed in accordance with 33 queries on this dataset (3.3% from total dataset size) as presented in Table 4.

As shown in Table 4, the proposed WDCD helps improve the performance of the image retrieval process. The percentages of improvement of WDCD (without MCR) over original LBA and original MPEG-7 are presented in Table 5(left part) in terms of ARR, ANMRR, MAP and \( P(10) \). The average improvement percentages of the proposed descriptor are 11.8% and 36.5% over LBA and MPEG-7 descriptors respectively. Moreover, the newly proposed similarity measure modification (MCR) also enhances the retrieval performance of all descriptors (MPEG-7, LBA, and WDCD) by 4.9%, 5.2% and 1.35%, respectively in terms of average percentages of the four used metrics, as presented in Table 5(right part).

From presented results, the proposed descriptor outperforms the other descriptors and all their enhanced versions, which contain the proposed MCR, in all four evaluation metrics. In Corel dataset, there are many classes that have object of different colors within same class such as bus, flower and others, as presented in Fig. 10. This certainly will effect on the accuracy of the WDCD, which mainly depends on colors in its retrieval. Therefore, 26 categories, from Caltech-101, of the same colored-object are selected to show effectiveness of the proposed descriptor, evaluation result of Caltech-26 is presented in the next section.

4.2.2. Retrieval performance of Caltech-26 dataset

To visually compare among the competing descriptors in the Caltech-26 dataset (according to 26 considered classes) refers to Fig. 8. Table 6 showcases the quantitative comparisons that are computed using the four evaluation metrics ARR, ANMRR, MAP, and \( P(10) \).

From Table 6, one can notice that the proposed descriptor WDCD performs better than all the other competing descriptors. It raises the performance average by 55.3% and 28.6% over MPEG-7 and LBA respectively in terms of average of the four evaluation metrics, as present in Table 7(left part). In addition,
MCR modification term enhances the average of performance for LBA, MPEG7 and WDCD by 10.5%, 10.3% and 2.1% respectively in terms of average of the four evaluation metrics, as shown in Table 7 (right part).

4.2.3. Retrieval performance of Cartoon-5 K dataset

Color feature plays an essential role in cartoon images [55]. Therefore, the latter have been used in color-based object image retrieval such as the work in [55]. In this work, the authors introduced new cartoon image dataset of 18 classes and 586 images as total. This motivates us to introduce large cartoon dataset (Cartoon-5 K) to test our proposed descriptor WDCD. Our dataset has 85 classed with 5128 cartoon images collected from Google. To visual compare among all the competing descriptors in Cartoon-5 K dataset refer to Fig. 9. Table 8 presents the quantitative comparisons that are computed using the aforementioned four metrics for 106 queries (2.1% from database size).

The visual and quantitative comparison of Cartoon dataset shows the discrimination power of the proposed semantic descriptor. With this method, the same cartoon character with different backgrounds will be retrieved. On the other hand, the other competing descriptors retrieve different characters with similar background. Moreover, as shown in Table 8, one can observe that combining the proposed descriptor and proposed similarity measure modification can successfully lead to have the best retrieval

Table 4
Four evaluation metrics values that computed for 33 queries in Corel-1 K dataset, the best result values are bolded.

<table>
<thead>
<tr>
<th>Descriptors</th>
<th>ARR</th>
<th>ANMRR</th>
<th>MAP</th>
<th>(P(10))</th>
</tr>
</thead>
<tbody>
<tr>
<td>MP7DCD</td>
<td>0.225</td>
<td>0.722</td>
<td>0.239</td>
<td>0.41</td>
</tr>
<tr>
<td>MP7DCD + MCR</td>
<td>0.241</td>
<td>0.708</td>
<td>0.250</td>
<td>0.44</td>
</tr>
<tr>
<td>LBA DCD</td>
<td>0.330</td>
<td>0.598</td>
<td>0.328</td>
<td>0.56</td>
</tr>
<tr>
<td>LBA DCD + MCR</td>
<td>0.342</td>
<td>0.581</td>
<td>0.345</td>
<td>0.62</td>
</tr>
<tr>
<td>Proposed WDCD</td>
<td>0.374</td>
<td>0.536</td>
<td>0.384</td>
<td>0.62</td>
</tr>
<tr>
<td>WDCD + MCR</td>
<td>0.380</td>
<td>0.535</td>
<td>0.387</td>
<td>0.64</td>
</tr>
</tbody>
</table>

Table 5
Improvement percentages of the proposed descriptor WDCD (without MCR) over original LBA and MPEG-7 descriptors (Left) and improvement percentage of adding MCR to all descriptors (Right) in Corel-1 K database.

<table>
<thead>
<tr>
<th>Improvement ratio</th>
<th>WDCD over LBA</th>
<th>WDCD over MPEG-7</th>
<th>LBA + MCR over LBA</th>
<th>LBA + MCR over MPEG-7</th>
<th>MPEG7 + MCR over LBA</th>
<th>MPEG7 + MCR over MPEG-7</th>
<th>WDCD + MCR over LBA</th>
<th>WDCD + MCR over MPEG-7</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARR</td>
<td>11.7</td>
<td>39.8</td>
<td>3.5</td>
<td>6.6</td>
<td>1.5</td>
<td>1.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANMRR</td>
<td>11.5</td>
<td>34.7</td>
<td>2.8</td>
<td>1.9</td>
<td>0.1</td>
<td>0.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAP</td>
<td>14.3</td>
<td>37.7</td>
<td>4.9</td>
<td>4.4</td>
<td>0.7</td>
<td>0.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(P(10))</td>
<td>9.6</td>
<td>33.9</td>
<td>9.6</td>
<td>6.8</td>
<td>3.1</td>
<td>3.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>11.8%</td>
<td>36.5%</td>
<td>5.2%</td>
<td>4.9%</td>
<td>1.35%</td>
<td>1.35%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
performance. The high performance of the proposed descriptors in the cartoon images back to the cartoon character often has the same colors. Besides, the cartoon images do not have illuminating variations that affect the most color-based descriptors. Despite of the large numbers of images and diversity of Cartoon-5 K dataset, the improving ratio in this dataset exceeds those of the two natural images datasets. The proposed descriptor results in improving the rates by 49.8% and 37.8% over MPEG-7 and LBA, respectively in terms of the average of the four used metrics, as shown in Table 9left part. MCR further improves the rates by 13.2%, 7.8% and 4.9% of LBA, MPEG-7 and WDCDs respectively, as presented in Table 9(right part).

As mentioned before, Corel dataset has some classes in which the object (within same class) has different color; this will degrade the proposed descriptor accuracy. Fig. 10 shows examples of how the proposed descriptor retrieves results worse than the other competing descriptors.

The reason behind the failure of the proposed descriptor in dealing with such cases is that an object of the query image has different colors from its ground truth images. Besides, the query has a similar background color to these ground truth images. Hence, the MPEG-7 and LBA DCDs that depend on the color of the large percentage (background) will outperform the proposed WDCD. However, the latter provides more semantic information (the color

<table>
<thead>
<tr>
<th>Descriptors</th>
<th>ARR</th>
<th>ANMRR</th>
<th>MAP</th>
<th>P (10)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MP7DCD</td>
<td>0.098</td>
<td>0.871</td>
<td>0.094</td>
<td>0.25</td>
</tr>
<tr>
<td>MP7DCD + MCR</td>
<td>0.111</td>
<td>0.855</td>
<td>0.106</td>
<td>0.30</td>
</tr>
<tr>
<td>LBADC DCD</td>
<td>0.186</td>
<td>0.763</td>
<td>0.182</td>
<td>0.40</td>
</tr>
<tr>
<td>LBADC DCD + MCR</td>
<td>0.218</td>
<td>0.725</td>
<td>0.205</td>
<td>0.45</td>
</tr>
<tr>
<td>Proposed WDCD</td>
<td>0.293</td>
<td>0.642</td>
<td>0.274</td>
<td>0.54</td>
</tr>
<tr>
<td>WDCD + MCR</td>
<td>0.298</td>
<td>0.635</td>
<td>0.280</td>
<td>0.56</td>
</tr>
</tbody>
</table>

Table 7
Improvement percentages of the proposed descriptor WDCD (without MCR) over original LBA and MPEG-7 descriptors (Left) and improvement percentage of adding MCR to all descriptors (Right) in Caltech-26 database.

<table>
<thead>
<tr>
<th>Improvement ratio</th>
<th>WDCD over LBA</th>
<th>WDCD over MPEG-7</th>
<th>LBA + MCR over LBA</th>
<th>MPEG7 + MCR over MPEG-7</th>
<th>WDCD + MCR over WDCD</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARR</td>
<td>36.5</td>
<td>66.5</td>
<td>14.6</td>
<td>11.7</td>
<td>1.6</td>
</tr>
<tr>
<td>ANMRR</td>
<td>18.8</td>
<td>35.6</td>
<td>5.2</td>
<td>1.8</td>
<td>1.1</td>
</tr>
<tr>
<td>MAP</td>
<td>33.5</td>
<td>65.6</td>
<td>11.2</td>
<td>11.3</td>
<td>2.1</td>
</tr>
<tr>
<td>P (10)</td>
<td>25.9</td>
<td>53.7</td>
<td>11.1</td>
<td>16.6</td>
<td>3.5</td>
</tr>
<tr>
<td>Average</td>
<td>28.6%</td>
<td>55.3%</td>
<td>10.5%</td>
<td>10.3%</td>
<td>2.1%</td>
</tr>
</tbody>
</table>
of the object) than the previous methods. For more illustration consider the retrieval result of the yellow flower image in Fig. 10. The image shows that the previous methods managed in retrieving images of flowers with different colors. The proposed method of the current work, on the other flip, managed to retrieve images of yellow flowers and yellow objects only. That is because the newly proposed descriptor is mainly an object’s color-based descriptor.

Moreover, in all evaluation datasets, there are many classes have same object color (e.g. yellow flower and yellow bus in Corel dataset, Sponge Bob and yellow Rabbit in Cartoon dataset as shown in Fig. 9 and many others). This will allow retrieving different object but of same color that in turn will degrade performance of the proposed descriptor. Therefore, additional features need to be integrated with color (such as spatial colors relations such as Kiranyaz...
et al. [14] and shape such as Khan et al. [55]) to complement the proposed descriptor to be semantic and suitable for object-based image retrieval.

From another perspective, the time is one of the important issues that can be considered in retrieval systems, especially in web-based image retrieval systems. To extract dominant colors from single image, GLA that used in MPEG-7 DCD requires 2.5 s while LBA requires 0.37 s. To extract salient object of an image, GC-SRD requires 1.5 s while LoG with flood fill algorithm requires 1.2 s. All experiments were conducted using Dual Core 2.0 GHz CPU with 3 GB RAM; the time is averaged from tens of experiments on different image resolutions. Therefore, the time required for the proposed method compared with the MPEG-7 and LBA DCDs is presented in Table 10. The accuracy of the proposed descriptor tends to be equaled in using any of DCs extraction method (GLA et al. [14] and shape such as Khan et al. [55]) to complement the proposed descriptor to be semantic and suitable for object-based image retrieval.

From another perspective, the time is one of the important issues that can be considered in retrieval systems, especially in web-based image retrieval systems. To extract dominant colors from single image, GLA that used in MPEG-7 DCD requires 2.5 s while LBA requires 0.37 s. To extract salient object of an image, GC-SRD requires 1.5 s while LoG with flood fill algorithm requires 1.2 s. All experiments were conducted using Dual Core 2.0 GHz CPU with 3 GB RAM; the time is averaged from tens of experiments on different image resolutions. Therefore, the time required for the proposed method compared with the MPEG-7 and LBA DCDs is presented in Table 10. The accuracy of the proposed descriptor tends to be equaled in using any of DCs extraction method (GLA et al. [14] and shape such as Khan et al. [55]) to complement the proposed descriptor to be semantic and suitable for object-based image retrieval.

From another perspective, the time is one of the important issues that can be considered in retrieval systems, especially in web-based image retrieval systems. To extract dominant colors from single image, GLA that used in MPEG-7 DCD requires 2.5 s while LBA requires 0.37 s. To extract salient object of an image, GC-SRD requires 1.5 s while LoG with flood fill algorithm requires 1.2 s. All experiments were conducted using Dual Core 2.0 GHz CPU with 3 GB RAM; the time is averaged from tens of experiments on different image resolutions. Therefore, the time required for the proposed method compared with the MPEG-7 and LBA DCDs is presented in Table 10. The accuracy of the proposed descriptor tends to be equaled in using any of DCs extraction method (GLA et al. [14] and shape such as Khan et al. [55]) to complement the proposed descriptor to be semantic and suitable for object-based image retrieval.

From another perspective, the time is one of the important issues that can be considered in retrieval systems, especially in web-based image retrieval systems. To extract dominant colors from single image, GLA that used in MPEG-7 DCD requires 2.5 s while LBA requires 0.37 s. To extract salient object of an image, GC-SRD requires 1.5 s while LoG with flood fill algorithm requires 1.2 s. All experiments were conducted using Dual Core 2.0 GHz CPU with 3 GB RAM; the time is averaged from tens of experiments on different image resolutions. Therefore, the time required for the proposed method compared with the MPEG-7 and LBA DCDs is presented in Table 10. The accuracy of the proposed descriptor tends to be equaled in using any of DCs extraction method (GLA et al. [14] and shape such as Khan et al. [55]) to complement the proposed descriptor to be semantic and suitable for object-based image retrieval.

From another perspective, the time is one of the important issues that can be considered in retrieval systems, especially in web-based image retrieval systems. To extract dominant colors from single image, GLA that used in MPEG-7 DCD requires 2.5 s while LBA requires 0.37 s. To extract salient object of an image, GC-SRD requires 1.5 s while LoG with flood fill algorithm requires 1.2 s. All experiments were conducted using Dual Core 2.0 GHz CPU with 3 GB RAM; the time is averaged from tens of experiments on different image resolutions. Therefore, the time required for the proposed method compared with the MPEG-7 and LBA DCDs is presented in Table 10. The accuracy of the proposed descriptor tends to be equaled in using any of DCs extraction method (GLA et al. [14] and shape such as Khan et al. [55]) to complement the proposed descriptor to be semantic and suitable for object-based image retrieval.
5. Conclusion

This paper introduces an adapted semantic dominant color descriptor that can be used in object-based image retrieval. The mechanism of the proposed descriptor is based on assigning weight to each DC in the image in accordance with its belonging to the object or to the background. The background colors, which are in contact with the image borders and out of salient object area, will receive a lower weight whereas the object colors, which are located at the salient object area and don’t touch the border, will receive a higher weight. Such a method helps alleviate the background effect. Additionally, the paper also introduces a new modification for the purpose of measure the similarity. This modified representation represents the mutual color ratio. The experimental results further show that the proposed semantic descriptor with the newly introduced similarity measure modification outperforms the original [9] and the best existing one of DC descriptors [11] in terms of four quantitative metrics (ARR, ANMRR, MAP and P(10)).

Acknowledgments

We thank the anonymous reviewers for their valuable comments and constructive recommendations for improving this manuscript.

References

[18] D. Sykora, J. Biraneck, and J. Zara, Sketching cartoons by example. 2nd Eurographics Workshop on Sketch-Based Interfaces and Modeling, 2005, pp. 27–34.
[31] D. Sykora, J. Biraneck, and J. Zara, Sketching cartoons by example. 2nd Eurographics Workshop on Sketch-Based Interfaces and Modeling, 2005, pp. 27–34.
[52] W. Rodhetbhai, Preprocessing for Content-Based Image Retrieval, Faculty of Engineering, Science and Mathematics School of Electronics and Computer Science, University of Southampton, 2009.


[58] J. Zhang, Robust content-based image retrieval of multi-example queries, University of Wollongong, 2011.