International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:5, No:11, 2011

Image Indexing Using a Color Similarity Metric based on the Human Visual System

Angelo Nodari, Ignazio Gallo

Abstract—The novelty proposed in this study is twofold and consists in the developing of a new color similarity metric based on the human visual system and a new color indexing based on a textual approach. The new color similarity metric proposed is based on the color perception of the human visual system. Consequently the results returned by the indexing system can fulfill as much as possibile the user expectations. We developed a web application to collect the users judgments about the similarities between colors, whose results are used to estimate the metric proposed in this study. In order to index the image's colors, we used a text indexing engine to facilitate the integration of visual features in a database of text documents. The textual signature is build by weighting the image's colors in according to their occurrence in the image. The use of a textual indexing engine, provide us a simple, fast and robust solution to index images. A typical usage of the system proposed in this study, is the development of applications whose data type is both visual and textual. In order to evaluate the proposed method we chose a price comparison engine as a case of study, collecting a series of commercial offers containing the textual description and the image representing a specific commercial offer.

Keywords—Color Extraction, Content-Based Image Retrieval, Indexing.

I. Introduction

MOTIVATED by the lack of the standard color metrics based on the RGB space [1] and other color spaces derived from it, we decided to follow a new strategy to obtain a set of colors correspondences based on the human visual system. This choice is motivated by the fact that we want to obtain a similarity metric, with the purpose of using it in an indexing engine, in which the expectation of the users about the results must be as similar as their color visual perception. We added subjectivity in color similarity perception and variations in perception according to age, gender and cultural environment [2] collecting ground truth from a large sample of people. To construct the proposed metric, we developed an online application allowing a user to select the set of colors considered most similar to a given sample color. This allows us to collect color information which is invariant to the color similarity perception between different individuals. To facilitate the ground truth data collection, we selected a set C of colors to discretize the whole RGB space, while to reduce the amount of information to be indexed we selected a subset M of color that we used for the color description. Using the collected judgments based on the C color set, we developed a similarity metric that we used to extract color information from the image. This metric is used to train a neural network which takes in input the value of a pixel and returns as output the most similar color belonging to the M color set. This process is repeated for all the pixels of interest in order to obtain a histogram that describes the overall distribution of color within the image. A common problem in Content-Based Image Retrieval [3] is the segmentation of the object of interest from the background, in order to extract the information only from a specific object. We have addressed the segmentation problem in a previous work [4] therefore, in this study we focus only on the extraction of color from an already segmented image. Once we have extracted the color histogram, we transformed this feature into two textual documents by repeating the colors belonging to C and to M a number of times equal to their occurrence in the image. Using these documents we have succesfully integrated the color feature in a text indexing engine ables to index millions of images offering high-speed responses to hundreds of queries per second[1].

The method to apply the queries depends on the type of search. In the case of a search using color facets, we look for the value of the selected facet in all the documents and in particular in their Main Colors field. Solr ranks the returned documents in according to the frequency of the search term in the documents. In the case of a Query by Example, the descriptive colors are concatenated in a single query and weighted according to their occurrence in the example image.

Unlike conventional indexing systems that for example use color histogram-based numerical data [5], vector quantization [6] and Color Correlograms [7], the way is open for exploitation of textual indexing systems. The state of the art in CBIR engine indexing is represented by the R*-trees [8], as they are primarily geared for handling numerical data, however we have opted to use Apache Solr[2], a frequently used search and indexing engine in the web environment, offering the guarantees of quality performance and functionality needed for document management and exhaustive dataset searching. A project that can be considered similar to our own, which uses an early version of Apache Solr called Lucene, is the Lucene-based open source CBIR project Lire [9] which extracts data from images that Lucene is unable to index, exploiting only the level of access to the file system. Our approach differs in that because we extract a textual color description from images that can be directly handled by Apache Solr.

Our case of study is a set of images related to a commercial offers set in the fashion domain. To test our system we have

A. Nodari and I. Gallo are with the Dipartimento di Scienze Biomediche, Informatiche e della Comunicazione, Universita degli studi dell'Insubria, Varese (VA), 21100 Italy e-mail: angelo.nodari@uninsubria.it (see http://www.dicom.uninsubria.it/∼angelo.nodari) and ignazio.gallo@uninsubria.it (see http://www.dicom.uninsubria.it/∼ignazio.gallo)

[1]http://www.drezzy.com
[2]http://lucene.apache.org/solr/

manually labeld a set of 600 images and for each of them we have specified the colors in the image and an estimation of their quantity. The dataset collected was uploaded to our homepage in order to be used by other methods for comparison[3].

II. Color Similarity Metric

In this section we describe the new strategy used to obtain the colors correspondences based on the human visual system. This choice is motivated by the fact that we want to obtain a similarity metric, with the purpose of using it in an indexing engine, in which the expectation of the users about the results must be as similar as their visual perception. We collect user judgments developing an online application allowing the user to select the color set $\{c_j|c_j \in C\}$ considered most similar to a given color $c_i \in C$. This allows us to detect the invariance in color similarity perception between different individuals. The web application has been online for 6 months and we have collected 3200 user judgments which are a sufficient number for a first estimation of the similarity measure proposed in this study.

In order to reduce the dimensionality of all the RGB color space and to select a proper subset of colors that are easily represented by a textual description for the indexing phase, as set of color C, we used the color descriptors derived from the ISCC/NBS system. This set has been proposed by the Inter-Society Council[4] and defines a lexicon for the construction of English words subdividing the color space into 267 centroids. The centroids names was created using the words outlined in Table I, obtaining an acceptable discretization of the whole color space (see [10] for more details).

Table I
Color name words used by the ISCC/NBS which we used to create the textual descriptors of the colors sets C used in this study.

Hue-p.	Hue-s.	Satur.	Lightness	Achrom.
Red	Reddish	Grayish	Blackish	Black
Orange	Brownish	Moderate	Very-dark	Gray
Brown	Yellowish	Medium	Dark	White
Yellow	Greenish	Strong	Medium	
Green	Bluish	Vivid	Light	
Blue	Purplish		Very-light	
Purple	Pinkish		Whitish	
Pink				
Beige				
Magenta				
Olive				

As shown in Fig.1, the data collected with this tool forms a similarity table, corresponding to a weighted similarity graph. The weights of similar colors are computed as follows:

$$w_{i,j} = \frac{\sum_p U_p(c_i, c_j)}{\sum_j \sum_p U_p(c_i, c_j)} \quad (1)$$

where $U_p(c_i, c_j)$ identifies the number of times a user U_p has chosen a color c_j as similar to c_i, where $c_j, c_i \in C$.

[3] http://www.dicom.uninsubria.it/arteLab/
[4] http://www.iscc.org/

In order to obtain a color similarity metric $S(c_i, c_j)$ from the users' judgments we applied a user votes normalization and a color votes closure using a minimization paths algorithm.

The first step consists of a normalization of the $w_{i,j}$ users' votes in order to guarantee the propriety $S(c_i, c_j) = S(c_j, c_i)$ for each color pair (c_i, c_j). The Algorithm 1 shows the normalization procedure.

Algorithm 1 Users' votes normalization

Require: The users' judgments for each color
1: **for all** $(c_i, c_j) \in C \times C$ **do**
2: $\quad \hat{w}_{i,j} = (w_{i,j} + w_{j,i})/2$
3: **end for**

The second step consists of a closure of all the paths from a color c_i to each other colors $c_j \in C$. In other words we want guarantee a distance between c_i and c_j also if there are no user judgments between different colors and that all the distances between c_i and c_j represent the minimum path in the graph showed in Fig 1. In order to obtain this constrain we used the Dijkstra algorithm [11] as showed in the Algorithm 2.

Algorithm 2 closure and minimum path

Require: The normalized users' judgments for each color
1: **for all** $c_i \in C$ **do**
2: $\quad K = Dijkstra(c_i), K = \{(c_{i,j}, \hat{w}_{i,j})|c_{i,j} \in C \times C, \hat{w}_{i,j} \in \mathbb{R}\}$
3: $\quad$ **for all** $(c_j, \hat{w}_{i,j}) \in K$ **do**
4: $\quad\quad \bar{w}_{i,j} = \hat{w}_{i,j}$
5: $\quad$ **end for**
6: **end for**

In this way we have obtained a color similarity metric $S : C \times C \Rightarrow \mathbb{R}$. With the previous steps we guarantee for each $c_i, c_j, c_k \in C$ the following conditions:

- $S(c_i, c_j) \geq 0$ (non-negativity)
- $S(c_i, c_j) = 0$ if and only if $c_i = c_j$
- $S(c_i, c_j) = S(c_j, c_i)$ (symmetry)
- $S(c_i, c_j) \leq S(c_i, c_k) + S(c_k, c_j)$ (triangular inequality)

The advantages of using a metric estimated with the method described above are as follow: possibility of continual refinement and updating as new user contributions are collected, freedom from rigid mathematical color-space constrictions and, finally, results tending to more closely approach user expectations, as showed in Section IV.

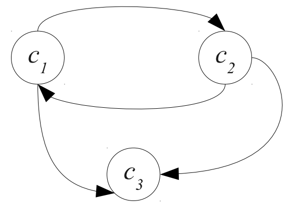
A. Color Extraction

To extract the color information from the image, we used a quantization technique, but the colors which we want to extract must be a sufficient number and typology to describe the entire color space without losing too information during the quantization phase. This step allowing us to switch from the RGB color space, composed by 16 Million of colors (2^{24}) represented by 8 bit for each channel, to a space of 27 *Main*

International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:5, No:11, 2011

	c_1	c_2	c_3	...
c_1	-	$w_{1,2}$	$w_{1,3}$	
c_2	$w_{2,1}$	-	$w_{2,3}$	
c_3	-	-	-	
...				

(a)



(b)

Fig. 1 Simple scheme of the color similarity table (a), and its equivalent graph (b)

Colors M subset of C, inspired by the *Fundamentals Colors* proposed in [2].

The *Main Colors* are used to briefly describe the dominant colors within the images, so it is possible to split the whole set of images according to a small set of colors. This set of colors, being very small, can be used in a typical faceted navigation, where the user can search for an item in a structure where all the facets of an item are a possible entry point. An example of this type of query is reported in Figure 5. The *Descriptive Colors*, instead, are used to describe more specifically the image and are then used to more specific types of navigation like Query by Example, as showed in Figure 6.

Through the process that we have described in [4], it is possible to select only the pixels belonging to the object of interest, called *Pixels of Interest*, avoiding the extraction of the colors from the background.

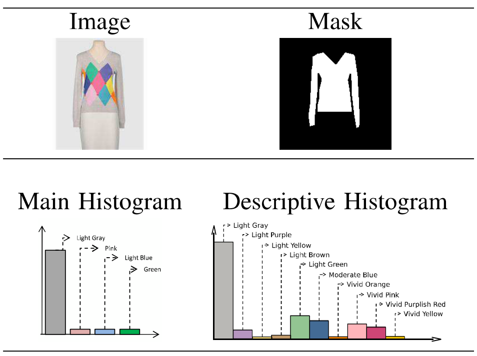


Fig. 2 An example of colors extracted from an offer's image in according to the set of pixel belonging to the object of interest showed in the alpha mask. The histogram represents the amount of colors contained in the object of interest. For each histogram's bin the textual descriptor of the representative colors is displayed.

A pixel of interest can represent any color in the RGB space, therefore the first step consists in the association of this pixel to a Descriptive Color C and after that to a Main Color M.

The time required to associate each pixel of an image to a Descriptive Color is $O(|C| * p)$ where $|C|$ is the number of Descriptive Colors and p is the number of pixels of interest in the image, the time complexity represents the time required to select the closest Descriptive Color using the Euclidean distance in the RGB space which is performed for each pixel of interest.

Each pixel was associated also with a Main Color M and at this phase, we did not use the Euclidean distance but the similarity measure proposed before, and in order to associate a color in the RGB space to a Main Color, we have chosen a Neural Network approach. We used a Multi Layer Perceptron with Resilient Back-propagation [12] which takes in input the RGB and HLS channels of a Descriptive Color C and returns, as output, the membership degree to each main color in M. The number of training patterns is $s * |M|$, where s represents the number of most similar Descriptive Colors for each Main Color. s has been calculated by dividing the number of Descriptive Colors by the number of Main Colors and rounded down to obtain an equal distribution for the Main Color in the ISCC/NBS space. In this way we succeeded in training the neural network using the similarity metric proposed in this study.

In order to verify the quality of the results of our approach we have performed two types of experiments. In the first experiment we have quantized, using a nearest neighbor color algorithm, the color space representation plotting the Hue and the Saturation on a set of images whose Lightness has been changed from 50 to 250, as showed in Figure 3. In the second experiment we have compared different metrics using the Euclidean distance in the RGB, HLS, CieLab spaces and the MLP approach explained in this study and we have measured the results using different evaluation approaches as explained in Section IV.

III. Color Indexing

The indexing approach we propose in this paper takes advantage of the *Apache Solr*-text indexing system. Once extracted the colors from the image, as explained in Section II-A, this information is converted in a text document containing the Main Colors M and the Descriptive Colors C as shown in Listing 1.

```xml
<?xml version="1.0"?>
  <doc>
    <title>Offer title</title>
    <description>Description of the offer</
        description>
    <mainColors>Gray Gray Gray Gray Gray Gray
        Gray Pink Light_Blue Green</mainColors>
    <descriptiveColors>Light_Gray Light_Gray
        Light_Gray Light_Gray Light_Gray
        Light_Gray Light_Gray Light_Gray
        Light_Gray Light_Gray Light_Gray
        Light_Gray Light_Gray Light_Gray
        Light_Gray Light_Gray Light_Gray
        Light_Gray Light_Gray Light_Gray
        Light_Gray Light_Gray Light_Gray
        Light_Gray Light_Gray Light_Gray
        Light_Gray Light_Gray Light_Gray
        Light_Gray Light_Gray Light_Gray
        Light_Gray Light_Gray Light_Gray
```

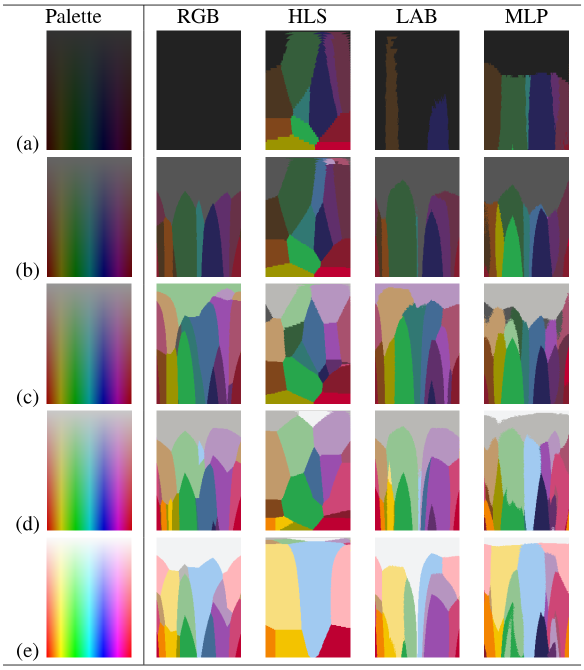


Fig. 3 Quantization using a Nearest Neighbor algorithm performed on the RGB, HLS and CieLAB distances compared with the method proposed which uses a Multi Layer Perceptron MLP. The Palette images are a color space representation of Hue and Saturation with a Luminance of (a) 50, (b) 100, (c) 150, (d) 200 and (e) 250

```
    Light_Gray Light_Gray Light_Gray
    Light_Gray Light_Gray Light_Gray
    Light_Gray Light_Gray Light_Gray
    Light_Gray Light_Gray Light_Gray
    Light_Gray Light_Gray Light_Gray
    Light_Gray Light_Gray Light_Gray
    Light_Purple Light_Purple Light_Purple
    Light_Purple Light_Purple Light_Yellow
    Light_Yellow Light_Brown Light_Brown
    Light_Brown Light_Green Light_Green
    Light_Green Light_Green Light_Green
    Light_Green Light_Green Light_Green
    Light_Green Light_Green Moderate_Blue
    Moderate_Blue Moderate_Blue
    Moderate_Blue Moderate_Blue
    Moderate_Blue Moderate_Blue
    Moderate_Blue Vivid_Orange Vivid_Orange
     Vivid_Pink Vivid_Pink Vivid_Pink
    Vivid_Pink Vivid_Pink Vivid_Pink
    Vivid_Pink Vivid_Purplish
    Vivid_Purplish Vivid_Purplish
    Vivid_Purplish Vivid_Purplish
    Vivid_Purplish Vivid_Purplish
    Vivid_Yellow Vivid_Yellow <!-- starting
     the repeated colors--!> Medium_Gray
    Medium_Gray Medium_Gray Medium_Gray
    Medium_Gray Light_Brownish_Gray
    Light_Brownish_Gray Light_Brownish_Gray
     Light_Brownish_Gray
    Light_Brownish_Gray Light_Purplish_Gray
     Light_Purplish_Gray
    Light_Purplish_Gray Light_Purplish_Gray
     ...</descriptiveColors>
</doc>
...
```

Listing 1. Example of a text document which describes a commercial offer of the dataset reported in Figure 2. It is possible to notice the fields related to the the Main Colors M and to the Descriptive Colors C.

The feature associated with each color image is composed of two fields that are generated in a different way. The first field concerns the Main Colors and set a fixed number of terms t, the text string is built according to the percentage of each Main Color extracted from the image. For example if 20% of $Color_1$ and 80% of $Color_2$ is extracted from an image with $t = 100$, we will repeat the textual labels identifing the $Color_1$ and $Color_2$, 20 times and 80 times respectivelly. The repetition of a color within the image according to its presence is needed in the retrieval phase in which Apache Solr makes a ranking depending on the term frequency of each words in the document. In this way is possible to sort the images according to the amount of the selected color which they contain.

As regards the Descriptive Colors, the textual feature generation is similar to that previously described with the only difference that for every Descriptive Color d we add in the document even the first $(d_s1, ..., d_sn)$ similar colors a number of times equal to $r_i = S(d, d_si) * O_d$ where O_d represents the percentage of occurrence in the document of d and S is the similarity function on the set of all colors C, $S : C \times C -> \mathbb{R}$ such that $S(c_i, c_j) = \{r|r \in \mathbb{R}, 0 \leq r \leq 1\}$. Where the multitude of colors extracted can best describe the image, but can not be used in a search using facets because introduce excessive noise to the query.

In the example showed in Figure 2, 53% of the pixels of interest have been associated with the color Light Gray and given $t = 100$ the number of word in the description field, the color in issue will be repeated 53 times. Fixing the number of similar colors to be considered to $n = 3$, in Table II the n most similar colors are showed as follow: "Medium Gray", "Light Brownish Gray" and "Light Purplish Gray" with a similarity value of 0.97, 0.95 and 0.84 respectively. So the color "Medium Gray" in according to the previous equation is repeated $0.53 * 0.97 * 10 = 5$ times, the "Light Brownish Gray" $0.53*0.95*10 = 5$ times and the "Light Purplish Gray" $0.53 * 0.84 * 10 = 4$ times.

Our indexing system can be queried in two different ways: using a query by facets or performing a query by example. In the first case is possible to search through all the images in the indexing engine which have among their *Main Colors* the selected *Color Facet* as showed in Figure 5. Furthermore, the result is ranked according to the term frequency within the documents that describe the image.

A query by example is performed using the information relating to the *Descriptive Colors* as contain a greater information and an example is reported in Figure 6. A typical feature, provided by the textual indexing engines, consists in the possibility of using a boosting factor used to weight the relevance of the terms which compose the query. Given a query image we use the descriptive colors boosted in according to their presence within the image. As an example given the following descriptive colors: 60% of $c1$, 30% of $c2$ and 10% of $c3$ they can be used to build the following query: "$c1^{0.6}$ OR $c2^{0.3}$ OR $c3^{0.1}$", performed on the descriptive color field

Table II
TABLE OF COLOR SIMILARITY EXTRACTED IN FIGURE 2. THE COLORS ARE REPORTED FROM THE TOP TO THE BOTTOM AT THE SAME ORDER THEY APPEAR ON THE HISTOGRAM IN FIGURE 2 FROM THE LEFT TO THE RIGHT IN ORDER TO FACILITATE THE READING. FOR EACH DESCRIPTIVE COLOR ARE DISPLAYED THE FIRST 3 SIMILAR COLORS.

Descriptive Color	First Similar Color	Second Similar Color	Third Similar Color
53% Light Gray	0.97 Medium Gray	0.95 Light Brownish Gray	0.84 Light Purplish Gray
5% Light Purple	0.98 Light Reddish Purple	0.92 Pale Violet	0.88 Medium Purple
2% Light Yellow	0.96 Pale Yellow	0.93 Brilliant Orange	0.88 Brilliant Yellow
3% Light Brown	0.97 Brownish Orange	0.91 Moderate Yellowish Brown	0.89 Grayish Yellow
10% Light Green	0.98 Moderate Yellow Green	0.95 Strong Yellowish Green	0.90 Moderate Bluish Green
8% Moderate Blue	0.98 Strong Purplish Blue	0.96 Moderate Purplish Blue	0.89 Strong Greenish Blue
2% Vivid Orange	0.97 Strong Orange Yellow	0.95 Deep Orange Yellow	0.92 Vivid Orange Yellow
8% Vivid Pink	0.99 Deep Purplish Pink	0.96 Moderate Purplish Pink	0.91 Deep Pink
7% Vivid Purplish	0.98 Vivid Purplish Red	0.94 Deep Purplish Pink	0.90 Deep Reddish Purple
2% Vivid Yellow	0.99 Light Yellow	0.96 Vivid Orange Yellow	0.87 Strong Yellow

of each documents in the dataset.

IV. EXPERIMENTS

We conducted two types of experiments, the first to evaluate the color extraction and the second to evaluate the retrieval performance in the indexing phase.
In order to evaluate the quality of the colors extracted by our system, we used the *color-texture* dataset proposed in a previous work [4] in which each image is provided by an alpha mask that identifies the object of interest. To each image of this dataset, we associated a set of colors using a graphical user interface that allows a user to choose the colors of the object of interest from a predefined set of colors M. In addition, for each color added by the user, is automatically estimated the percentage of the selected color within the image in order to have a measure of the distribution of colors. In this way we collected a dataset of truth in which every image is associated with the colors of the object of interest with the value of occurrence of each color and the dataset is available at this page[5] to facilitate the comparison with other methods.

We used a metric that takes into account the occurrences of the colors extracted to verify the correctness of the extraction phase regardless of the amount of colors found in the image, that we called "Coarse Metric". This metric simply counts the number of True Positive, False Positives, True Negative and False Negative for each Main color of each image in the extraction phase in order to evaluate the Precision, Recall, Overall Accuracy and F-measure reported in Table III. We also used a metric for the evaluation of the extracted colors which takes into account the correctness of the colors extracted and also the amount of color in relation to the quantity of labeled color, as explained in [13], that we called "Fine Metric". This metric takes in consideration the amount of color correctly extracted (True Positive), the amount of color incorrectly extracted (False Positive), the amount of color correctly not extracted (True Negative) and the amount of color incorrectly not extracted (False Negative) for each Main Colors. In Figure IV we report a graphical explanation of this evaluation system and the results of its application are reported in Table IV. The results show how the proposed method outperforms the standard method of color extraction based on the Euclidean distance on the RGB, HLS and CieLAB space using both the Coarse Evaluation and the Fine Evaluation.

In addition, to fully evaluate the proposed method, we used additional metrics based on the comparison of color histograms showed in Table V. The Bhattacharyya measure [14], which is an index used in the Color domain to compare histograms, shows that the proposed method is closer to the truth. The Bhattacharyya measure indicates a measure of distance between two histograms, the higher the index more different are the histograms and it was computed as average of all the patterns in the dataset. Considering the color histogram as a probability distribution, we can use a correlation coefficient to analyze how the histogram of test is similar to the histogram of truth. At the same time also the Correlation value highlight the relationship between the proposed method and the truth, but the higher the index the greater the degree of correlation between the two distributions. The ChiSquare distance is computed by the sum of the square errors of each bins in the histogram and the Intersect distance measures the amount of area of the intersection between two histograms.

Table III
RESULTS OF THE RGB, HLS AND CIELAB EUCLIDEAN DISTANCE COMPARED WITH THE PROPOSED METHOD EVALUATED WITH A METHOD WHICH TAKES INTO ACCOUNT ONLY THE OCCURRENCES OF THE COLORS EXTRACTED.

Coarse Evaluation	RGB	HLS	CieLAB	MLP
Precision	0,28	0,25	0,30	**0,41**
Recall	0,75	0,64	0,74	**0,78**
Overall Accuracy	0,78	0,77	0,79	**0,85**
F-measure	0,39	0,34	0,45	**0,51**

[5]http://www.dicom.uninsubria.it/arteLab

Table IV
RESULTS OF THE RGB, HLS AND CIELAB EUCLIDEAN DISTANCE COMPARED WITH THE PROPOSED METHOD EVALUATED WITH A METHOD WHICH TAKES INTO ACCOUNT THE CORRECTNESS OF THE COLORS EXTRACTED AND ALSO THE AMOUNT OF COLOR IN RELATION TO THE QUANTITY OF LABELED COLOR.

Fine Evaluation	RGB	HLS	CieLAB	MLP
Precision	0,44	0,31	0,44	**0,49**
Recall	0,48	0,36	0,48	**0,53**
Overall Accuracy	0,60	0,54	0,61	**0,64**
F-measure	0,46	0,33	0,46	**0,50**

Since the computational time of the algorithm depends on the number of the computed pixels depending by the size of the image, we have investigated the way to reduce it, resizing the images but preserving the color information. All the tests were performed using a single thread C# code, on an Intel®Core™Duo processor T8100 @2.10Ghz. We have thus reached a tradeoff between speed and quality of results, without losing color information, by resizing the images before the extraction phase, to 100×100 pixels. The average time required to extract both the Main Colors and the Descriptive Colors is $48ms$ for each image, obtaining the performance showed in Table III and Table IV. In such a way it is possible to use the proposed method in real-time applications.

Table V
RESULTS COMPARING THE RGB, HLS AND CIELAB STANDARD METHODS AND THE PROPOSED METHOD USING DIFFERENT HISTOGRAM DISTANCES MOREOVER THE AVERAGE COMPUTATION TIME FOR A SINGLE IMAGE IS REPORTED.

	RGB	HLS	CieLAB	MLP
Bhattacharyya	0,94	0,70	0,60	**0,54**
Correlation	**0,60**	0,35	0,53	0,56
ChiSquare	**0,53**	1,23	0,95	0,84
Intersect	0,44	0,31	0,44	**0,49**
Average time	50,74	46,03	58,77	48,00

From a qualitative point of view, the results in Figure 7, highlight the problems using a metric based on the Euclidean distance in the RGB, HLS and CieLAB space using a limited number of colors, as the set proposed in this study. In particular the quantization which adopts this method leads to the introduction of artifacts during the color extraction. We report an example of colors related with the mismatch showed in Figure 7. From the table it is possible to calculate all the Euclidean distance between the "Test Pixel" extracted from the gray skirt of the first picture in the Figure 7 and its most similar Main Colors.

Table VI
THE "TEST PIXEL" IS A GRAY PIXEL EXTRACTED FROM THE IMAGE SHOWED IN FIGURE 7, IN PARTICULAR FROM THE SKIRT OF THE FIRST IMAGE AND WE REPORT THE RGB VALUES OF THE MOST SIMILAR MAIN COLORS.

	Test Pixel	Dark Gray	Medium Gray	Light Gray	Light Green
R	154	85	130	185	147
G	155	85	132	184	197
B	147	85	119	181	146

Given E the Euclidean distance on the RGB space we report all the distances between the color of the Test Pixel (TP) showed in Table VI with the colors: Dark Gray (DG), Medium Gray (MG), Light Gray (LG) and Light Green (LGr)

$$E(TP, DG) = \sqrt{(154-85)^2 + (155-85)^2 + (147-85)^2} = 116,21$$
$$E(TP, MG) = \sqrt{(154-130)^2 + (155-132)^2 + (147-119)^2} = 43,46$$
$$E(TP, LG) = \sqrt{(154-185)^2 + (155-184)^2 + (147-181)^2} = 54,38$$
$$E(TP, LGr) = \sqrt{(154-147)^2 + (155-197)^2 + (147-146)^2} = 42,59$$

As we can see from the previous calculations the most similar Main Color to the "Test Pixel" is the "Light Green" color, which is the closest color from a mathematical approach using a RGB Euclidean distance, but not from a human perception point of view. Using the proposed method it is possible to avoid this kind of artifacts as showed in Figure 7.

Once the documents have been indexed, to evaluate the ranked results of our indexing system, we used a well-know method in Information Retrieval called ($nDCG$) Normalized Discounted Cumulative Gain [15]. The $nDCG$ is computed starting from the (DCG) Discounted Cumulative Gain:

$$DCG_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{\log_2 1 + i} \quad (2)$$

Where p is the rank position of a specific retrieved document, rel_i is the relevance value associated to a specific type of document (in our case $rel_i \in \{0, 1\}$ such that $rel_i = 0$ is a not relevant document and $rel_i = 1$ is a relevant document).

The $nDDG$ take also in consideration the ($IDCG$) Ideal Discounted Cumulative Gain computed on the truth corpus of documents:

$$nDCG_p = \frac{DCG_p}{IDCG_p} \quad (3)$$

The value of the $nDCG$ represents the goodness of the ranking and in particular $nDCG \in [0, 1]$ where a value of 1 represents the ideal ranking result.

We have performed two type of experiments reporting the results in Table VII choosing $p = 50$ the number of the first ranked documents to evaluate, which is a reasonable number which summarize the average number of elements usually returned by a search engine in the first page. In the first experiment (Query by Facet) we performed a query for each Main colors and the Ideal Ranking was built ordering the documents in according to the quantity of color information in the truth dataset. In the second experiment (Query by Example) for each document of the dataset we have performed

International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:5, No:11, 2011

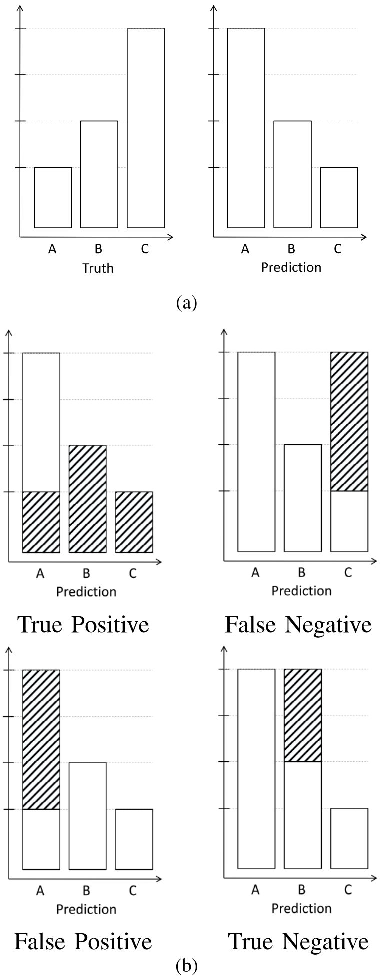


Fig. 4 The histograms in the figure represent three colors such as A, B and C belonging to the hypothetical image of test and the height of the histograms represent the amount of colors. In Figure IV we report the histogram of truth and the histogram of the extracted colors. The Figure IV highlights the areas of prediction relative to the True Positive, False Negative, False Positive and True Negative respectively, in according to the truth.

a query as explained in Section III. In this experiment the Ideal Ranking, for each example, was built using the distance between the truth color histograms of the dataset. The results in Table VII show the performance of our color indexing system.

Table VII
$nDCG$ ON THE RANKED RESULT OF THE INDEXING SYSTEM PROPOSED. IT REPRESENTS THE CORRECTNESS IN THE RETRIEVED RESULTS TAKING IN CONSIDERATION THE POSITION OF THE RETRIEVED DOCUMENTS. ITS VALUE LIES BETWEEN 0 AND 1 WHERE 1 REPRESENTS THE IDEAL RANKING RESULT.

	Normalized Discounted Cumulative Gain
Query by Facet	0,82
Query by Example	0,93

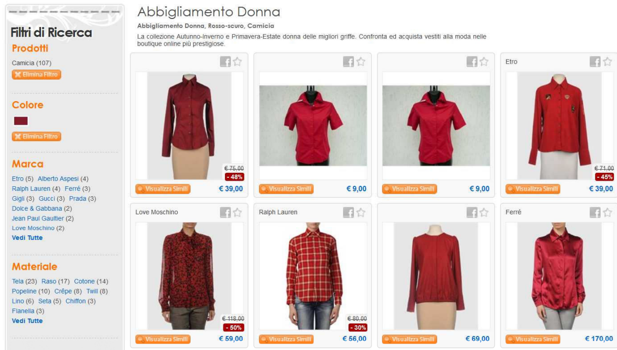


Fig. 5 An example of the Query by Color Facet using the proposed method taken by drezzy.com. On the top left side of the image, under the label "Colore" it is possible to see the selected facet (the red color facet).

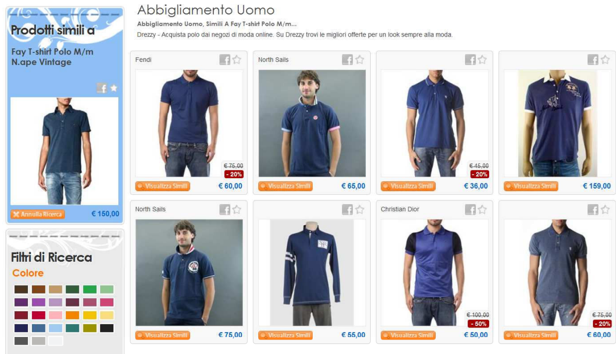


Fig. 6 An example of the Query by Example using the proposed method taken by drezzy.com. On the top left of the image, in the blue box, it is possible to see the example image which is used to search similar images with the same colors.

V. CONCLUSION

In this study, we have addressed the problem of image indexing focusing on the color feature. We have presented an innovative solution based on a textual representation of the color feature, introducing a new similarity metric based on the human visual system.

The advantages in the use of a textual representation appear to be the easy integration of the color feature in a text indexing engine (Apache Solr has been used in this study) without the necessity of an auxiliary database for the integration of the information about the images. Which is one of the limiting factors in the integration of text and images in a single indexing engine. In this way it is possible to make queries based both on text within the document and color from the associated images, delegating the management of the ranking of the results to the indexing system.

The use of the similarity metric proposed and evaluated in this study made it possible to overcome the limitations imposed by the usual methods of similarity based on strict mathematical constraints of the RGB space or its linear transformations, as showed in Section IV. Experiments performed in this study show the advantages compared to other methods and leave the way open to the investigation in the field of image quantization taking advantage of this new metric.

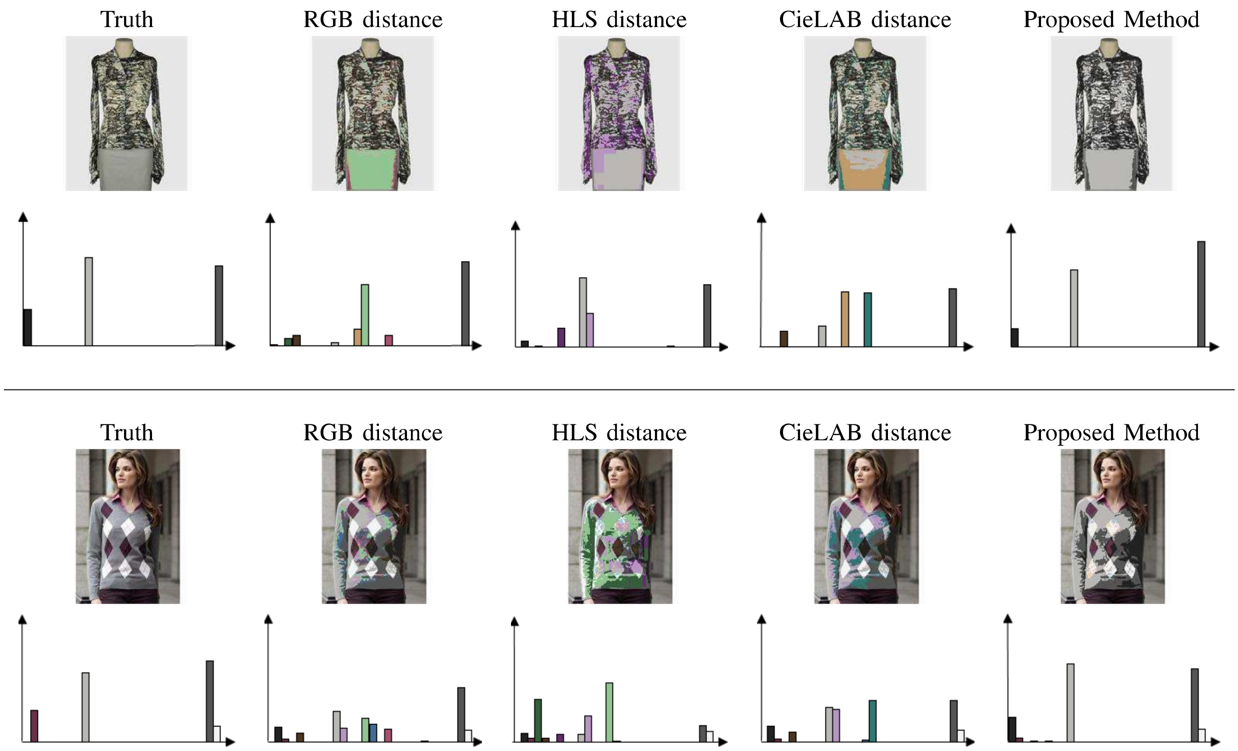


Fig. 7 An example of colors extracted from an offer's image in according to the set of pixel belonging to the object of interest. In particular only the pixels belonging to the object of interest have been quantized in the reported images. The histograms represent the amount of colors contained in the object of interest. For each histogram's bin the textual descriptor of the representative colors is displayed. As we can see each distance method presents problems in the color extraction phase due to the smaller set of main colors chose in this study for the feature description. In contrast, the proposed method is able to manage these constrains and gives a fine and robust color description close to the human visual perception of color.

The performance results of the indexing phase show the full usability and robustness of the proposed method. Although it was developed an online application to collect the largest possible number of user judgments, the results highlight a limitation of the data collected and the noise due to the fact that to get a more accurate metric, it is needed much more user judgments. Moreover, different types of user's screen resolution, have made the collection of the judgments a very hard task.

A. Future Works

The main purpose of image quantization consists in the reduction of the number of distinct colors of an image with the aim that the compressed image should be as visually similar as possible to the original image. For this reason in this study we open the way for investigation using the novel quantization method based on the metric explained in Chapter II.

There are several ways to perform a quantization: the nearest color algorithm which is used mainly for fixed palettes, the Median Cut algorithm [16] and the algorithms based on Octrees [17] which are slower but adaptive and considered the state of the art. The solution proposed in this study can be easily used in the first approach and the main advantage is that using the proposed metric is possible to reduce the number of distinct colors in the image while preserving visual similarity and at the same time keeping high speed performances. At the same time can also be used as a method of color distance for the other quantization approaches.

The types of images for which the quantization is of primary importance concerns the images downloaded from the web, whose short download times are an important (and often the primary) consideration. At the same time the result of quantization must be as similar as possible to the original image, in order to preserve the visual content of the image.

This study opens also the way for the generation of a more descriptive textual description starting from other visual features such as texture or shape in order to develop a more powerful textual indexing system able to manage both text and images.

REFERENCES

[1] B. Funt, K. Barnard, and L. Martin, "Is machine colour constancy good enough?" in *In Proceedings of the 5th European Conference on Computer Vision*. Springer, 1998, pp. 445–459.

[2] B. Berlin and P. Kay, *Basic Color Terms: Their Universality and Evolution*. Center for the Study of Language and Inf, 1969.

[3] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1349–1380, 2000.

[4] I. Gallo and A. Nodari, "Learning object detection using multiple neural netwoks," in *VISAP 2011*. INSTICC Press, 2011.

[5] M. J. Swain and D. H. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, pp. 11–32, 1991.
[6] S. W. Teng and G. Lu, "Image indexing and retrieval based on vector quantization," *Pattern Recogn.*, vol. 40, no. 11, pp. 3299–3316, 2007.
[7] J. Huang, S. R. Kumar, M. Mitra, W. Zhu, and R. Zabih, "Image indexing using color correlograms," in *CVPR '97*. Washington, DC, USA: IEEE, 1997, p. 762.
[8] N. Beckmann, H. Kriegel, R. Schneider, and B. Seeger, "The r*-tree: an efficient and robust access method for points and rectangles," *SIGMOD Rec.*, vol. 19, pp. 322–331, 1990.
[9] M. Lux and S. A. Chatzichristofis, "Lire: lucene image retrieval: an extensible java cbir library," in *MM '08*. New York, NY, USA: ACM, 2008, pp. 1085–1088.
[10] G. Wyszecki and W. S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae (Wiley Series in Pure and Applied Optics)*. Wiley-Interscience, 2000.
[11] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, Dec. 1959.
[12] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: The rprop algorithm," in *IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS*, 1993, pp. 586–591.
[13] V. Pedoia and e. a. Colli, V., "fMRI analysis software tools: an evaluation framework," in *SPIE Medical Imaging 2011*, 2011.
[14] T. Kailath, "The divergence and bhattacharyya distance measures in signal selection," *IEEE Transactions on Communications*, vol. 15, pp. 52–60, 1967.
[15] K. Jarvelin and J. Kekalainen, "Cumulated gain-based evaluation of ir techniques," *ACM Transactions on Information Systems*, vol. 20, pp. 422–446, 2002.
[16] P. Heckbert, "Color image quantization for frame buffer display," *Computer Graphics*, vol. 16, pp. 297–307, 1982.
[17] M. Gervautz and W. Purgathofer, "Graphics gems," A. S. Glassner, Ed. San Diego, CA, USA: Academic Press Professional, Inc., 1990, ch. A simple method for color quantization: octree quantization, pp. 287–293.

Angelo Nodari received his MSc degree in Computer Science from the Universita degli Studi dell'Insubria, Varese, Italy in 2009. In 2010 he joined the Dipartimento di Informatica e Comunicazione at Universita degli Studi dell'Insubria, Varese, Italy as a Phd Student. His main fields of interest are Computer Vision and Machine Learning, in particular Content-Based Image Indexing and Retrieval and Object Segmentation. He has recently started to collaborate with 7Pixel s.r.l. in the Research and Development department in the developing of a visual and price comparison engine in the domain of visual shopping online.

Ignazio Gallo received his degree in Computer Science at the University of Milan, Italy, in 1998. From 1998 to 2002 he worked at the National Research Council (CNR) in Milan for the laboratory of Artificial Intelligence and Soft Computing. He was involved in the definition and development of neural models for classification and recognition of remote sensing images. He was also involved in the definition and development of neural models for decision support activities in engineering and environmental field. From 2002 to 2003, he worked at the Department of "Informatica e Comunicazione" at the University of Insubria, Varese, dealing with simulators in a distributed environment. He is also interested in stereoscopic reconstruction analysis of 3D images produced by a scanning electron microscope and in features selection and classification methods applied to hyperspectral data. Since 2004 he is assistant professors at the University of Insubria. Conducts research in Image Processing, Pattern Recognition, Neural Computing, Computer Vision and Information Extraction. Is is responsible for all aspects of the courses of "Software Engineering Laboratory", "Laboratory of Languages" and part of the course "Intelligent Systems II".