Quantistic Approach for Classification of Images

Federico F. Barresi, Giuseppe Battista, Jacopo Pellegrino
Dipartimento di Fisica
Università di Torino
Torino, Italy
e-mail: federico.barresi@studenti.unito.it,
giuseppe.battista@studenti.unito.it,
jacopo.pellegrino463@studenti.unito.it

Walter Allasia
Research Department
EURIX
Torino, Italy
e-mail: allasia@eurix.it

***Abstract*—This paper describes a novel approach for managing low level descriptors of images in order to allowing automatic classification and similarity searches. Several works have been made in this field, mostly making use of vector spaces and classical mathematical approaches. The focus of this paper is to investigate the more sophisticated formalisms of Quantum Mechanics that allows to manage images as *quantum states*. In order to check our theoretical method, we have considered a simple set of low level descriptors of images, the Hue Saturation Values (HSV). On the one hand, they are obviously not exhaustive and limited; but, on the other hand, they are enough for demonstrating that low level image descriptors can be represented such as Functions on Hilbert spaces. Since the distance between two colors in HSV space coincides with the human eye perception, its evaluation enabled us to collect results on similarity even if we were making use of few descriptors around HSV such as the MPEG-7 Visual Descriptors ColorLayout and DominantColor. The reader will not find a deep proof of a novel similarity technique applied to large image samples. Instead, she can assess the value of adopting Quantum Mechanics formalisms, translating thresholds of classical vector space distances onto probability density functions of low level image descriptors, such as HSV.**

Keywords*—Information retrieval; Image indexing; Image feature management; Novel spaces for indexing low level image features; Similarity comparison; Perception of similarity of image features

I. Introduction

The similarity between images is one of the most important topics in computer vision. Although it is very easy for people to decide if two images are similar, the same can not be said for computers that must learn to see like humans. Each image is perceived by the computer as a set of pixels ordered according to their position but not to their color, each pixel does not take into consideration the color of its neighbors unless any feature is applied. Therefore, we tried to take advantage of the position and color information that the computer is able to provide for each pixel in order to translate them into the kind of information that a human eye would receive observing the image. For this purpose, several MPEG-7 descriptors [12] have been implemented which allow to extract certain features from an image. In our work, we focused the attention to a couple of them, ColorLayout and DominantColor, in order to propose an alternative implementation of these two descriptors based on a new quantistic approach and on the HSV color space. The ColorLayout is rewritten as ColorDistribution to underline that the main difference is that for each area we consider a distribution of color rather than an average color. The similarity between images is then given by the comparison between the distributions, for each corresponding area of the images to be compared, which returns the percentage of similarity. Concerning the HSVDominantColor, the division of the image into areas is introduced and for each area three dominant colors are estimated. To obtain a percentage of similarity, HSVDominantColor calculates the distance in the HSV cone between colors of the areas of the images to be compared.

The paper is organized as in the following: Section III describes the approach we applied to implement the application, which is presented in Section IV. Section V shows the experimental results achieved with a sample made up of about 3000 images belonging to the collection provided for the research purposes by IRMA project [5].

II. State of the art

MPEG-7 [11], [12], [15] formally named *Multimedia Content Description Interface*, is developed by MPEG (*Moving Pictures Experts Group*) [7]. It is one of the most common standard for describing multimedia contents that provides a rich set of multimedia content description tools for applications ranging from content management, organization, navigation and automated processing.

MPEG-7 Visual [8] standardizes the description tools to describe video and image content. The Visual Descriptors are based on visual features that allow to estimate similarity in images or video. Therefore, they can be used to search and filter images and videos based on several visual features like color, texture, object shape, object motion and camera motion. Among Color Descriptors, we have taken into account ColorLayout and DominantColor due to previous research work at EURIX S.r.l. [20].

ColorLayout is a low-level descriptor that extracts information about color and its position within the image. This descriptor divides the image into 64 areas to which associates a representative color and then compares it with the color of the corresponding area of another image by calculating the Euclidean distance in the RGB color space [13]. The representative color can be evaluated with any method. In our work, we followed the [8] standard recommendations using the average of the pixel colors in a block as the corresponding representative color.

 ISBN: 978-1-61208-265-3

DominantColor is also a low-level descriptor that extracts the most present color within the image without caring at its spatial distribution. As ColorLayout, comparison between images is due to the calculation of the distance between the two colors in the RGB space multiplied by constant factors taking into account the spatial coherence [18].

In order to apply for a search on images and videos, we have tried to change the nature of these descriptors by introducing the formalism of Quantum Mechanics[16]. In this kind of approach, each object becomes a normalized vector $|x\rangle$ in a real Hilbert space of finite dimension $\mathcal{H}$. The vector contains the answers to all possible queries [14] [17], each one represented by another vector $|y\rangle$. Usually, in Information Retrieval, similarity matching is accomplished by computing $\langle x|\ y\rangle$. In the following section, we describe how this approach has been implemented.

III. Approach and methods

This work mainly aims at suggesting a novel kind of descriptors whose implementation takes into account the formalism of Quantum Mechanics. Another target is to improve the perceptivness in order to model the way human eye perceives the similarity between images. It is necessary to switch from global information, such as average values, to more detailed information, as distributions of color occurrences. It is also important to describe colors in a space where the perceptive distance is preserved. We selected the HSV space.

Since we want to apply the formalism of Quantum Mechanics in a HSV space, we have implemented the already mentioned new descriptors ColorDistribution and HSVDominantColor.

A. The ColorDistribution

As described later in Section IV-B, we have split each image into 64 rectangular cells of equal area. Since we did not consider necessary the use of interpolation or anyother solutions, partitioning inevitably means little loss of information because not every pixel of the image can be associated with one of the cells. For each cell, the R, G and B components of all the pixels are extracted, with an appropriate algorithm the conversion is carried out and the values of the components H, S and V are stored into arrays. From these lists, for each cell, relative frequency histograms of the three components are built. It is necessary to highlight that each component spans discreetly in its domain by unitary steps. This choice is reasonable since the variation of one unit of any of the three components is unnoticeable to the human eye and makes the histogram more fittable with a polynomial function. The histogram of H has 360 bins while the ones of S and V have 100 bins because of their percentual variation.

At this point, each histogram is interpolated with a polynomial function whose degree is set to 10 because this order allows us an acceptable level of flexibility. The 11 parameters of the polynomial are calculated from the fit with the method of least squares [10]. Each cell, therefore, is described by 3 polynomial functions of degree 10. Reiterating the process for all the cells of the image 3×64 functions are obtained, that describe the color distribution within the image. This is the main difference of ColorDistribution compared to ColorLayout which returns 64 average colors in the RGB space.

Once the color distributions of 2 images to be compared is obtained, it must be expressed as an information about their similarity. The procedure we adopted is to make a comparison between the three distributions of the corresponding cells of the two images, for every cell. At this point, the formalism of Quantum Mechanics is applied: the distributions can be considered as normalized vectors in the Hilbert space and can be compared using the standard scalar product between functions. The result of this comparison is the probability that the two distributions coincide. The comparison through the scalar product is performed cell by cell and component by component in order to obtain, for the whole image, a percentage of compatibility for each component (see: Section IV-C).

B. The HSVDominantColor

As ColorDistribution, HSVDominantColor makes use of colors of the HSV space as well as the partitioning of the image into cells. In order to decide the actual dominant colors, we divided the HSV space into relevant fields obtained through the division of Hue into 6 areas, Saturation into 4 areas and Value into 5 areas. Each area is identified by its average value. This quantization allows us consider only $6 \times 5 \times 4$ colors which is fundamental for at least two reasons. First of all taking into account all the possible colors does not make sense and it is completely useless since we would find out that identical colors occurs rarely in an image, to gather them into a finite number of ranges seems a valid solution. On the other hand, human eye is not so sensitive to distinguish a unitary variation of any of the three component. For each cell, the 3 most frequent representative colors are calculated and can be considered as the dominant colors. We decided to split the HSVDominantColor space into 3 subsets in order to take into account any edges contained in a single cell. If a single dominant color is found this is repeated twice. If two dominant colors are found, the most frequent is repeated once. The procedure is iterated for every cell obtaining 64 triples of dominant colors. The comparison between two images is therefore the calculation of the distance between the corresponding dominant colors of cells that occupy the same position. From the three distances, the average distance for each cell is evaluated and the total percentage of compatibility between the two images is finally obtained as the total normalized average distance.

C. The HSV distance algorithm

In order to calculate the distance [9], [19] between two colors in the HSV space, we have introduced the following Algorithm 1. Firstly, it selects color whose Value is maximum and sets it as $Color_1$. The other color is set to $Color_2$. Then it projects $Color_2$ onto $Color_1$ plane, calculates the distance on that plane and trough Carnot's theorem finds the distance in the HSV cone.

IV. Implementation and tests

A. Software architechture

The software architecture diagram in [1] shows the class diagram structure of the application based on elementary classes such as *Point*, *Pixel*, *Color* and *Cell*. Starting from them, more

 ISBN: 978-1-61208-265-3

Require: $Color_1 := (h_1, s_1, v_1), Color_2 := (h_2, s_2, v_2)$
if $v_1 > v_2$ **then**
{Colors swap}
$c' := c_1$
$c_1 := c_2$
$c_2 := c'$
end if
$\Delta h := |h_2 - h_1|$
$\Delta v := \frac{|v_2 - h_1|}{100}$
{Projection on $v = v_2$}
$d_c := \frac{\Delta v}{\cos(\pi/4)}$
{Distance on the plane $v = v_2$}
$d_p := \sqrt{(\frac{s_1}{100})^2 + (\Delta v)^2 + 2 \cdot \frac{s_1}{100} \cdot \Delta v}$
$+\sqrt{(\frac{s_2}{100})^2 - 2 \cdot (\frac{s_1}{100} + \Delta v) \cdot (\frac{s_2}{100} \cdot (\Delta h))}$
{Distance between $Color_1$ and $Color_2$}
return $d := \frac{\sqrt{d_c^2 + d_p^2 - 2 \cdot d_c \cdot d_p \cdot \cos(\pi/4)}}{2}$

Algorithm 1. HSV distance

TABLE I. Waste percentage of pixels

Cells	Loss percentage %
64	1.20
144	1.97
256	2.61
400	3.42
576	3.94
784	4.31
1024	6.00

complex objects and new methods are implemented in order to perform the following steps:

Step 1 Image management: the application reads two JPEG files as arguments
Step 2 Partitioning (see: IV-B) and HSV conversion: a *Cell-Builder* object instantiates *Pixel* objects, containing information regarding color and position, and associates them with the corresponding cell according to its location within the image.
Step 3 Image analysis: ColorDistribution and HSVDominantColor extract the features by iterating over the cells that the picture is divided into.
Step 4 Comparison: the exctracted features of the two images are compared using the scalar product for ColorDistribution and the HSV distance for HSVDominantColor.

B. Image partitioning

Since our descriptors must preserve spatial information it is necessary to split the images into cells. The image partitioning occurs through the division into 64 $(n \times n)$ rectangular areas. Neither the vertical nor the horizontal size are in any case integer multiples of n. Hence a certain number of pixels can not be associated with any cell. Partitions with various values of n in steps of 4 are tested on a 36 images sample in order to evaluate and minimize the loss of pixels. Results are reported in Table I.

The choice of considering a number of cells greater than 64, currently used by cell-based descriptors, is due to the need to operate with higher precision. Test demonstrates that a 64 cells partitioning minimizes the loss of pixels.

TABLE II. Survey results regarding Hue

ΔH %	Dissimilarity %
10	96.7
15	98.8
20	99.5
Mean:	98.3

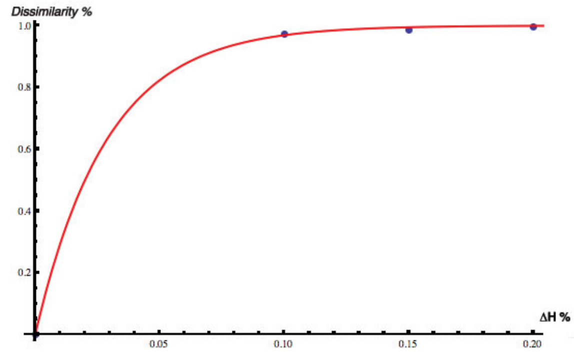

Fig. 1. Graph of Hue perception: It shows dissimilarity between images as function of variation of H

C. Survey

The ColorDistribution descriptor returns a compatibility percentage for each component of the HSV space. In order to estimate the weight of each component of HSV space we set up a survey asking people to recognize images having HSV slightly modified. The proper relevance of each component in determining the similarity between two images is established by the results of the survey that we have submitted to a sample with more than 600 people asked to indicate, among three or four altered images, the most similar to the original one. Survey results are reported in Figure 1 and in Table II and Table III.

Table II is showing the acquired dissimilarity perception whose mean value amounts to H = 98%. The remaining 2% can be split into S and V according to the results shown in Table III.

According to the collected results, it is possible to assert that a small variation in Hue leads to perceive the image as very different from the original. We have chosen H as the most relevant component and its relevance has been set to 98%. The remaining 2% has been shared between Saturation (8.3%) and Value (91.7%).

D. Multi-threading

Comparison between polynomial functions, performed by the ColorDistribution, has been evaluated through the standard scalar product in Hilbert space. This process requires much more computing power than needed by ColorLayout descriptor. We needed to implement a Thread-Manager which distributes the computation on all the available *cores* of the computer running the application.

TABLE III. Survey results regarding Saturation and Value

Δ %	S Dissimilarity %	V Dissimilarity %
30	3.06	96.94
40	8.29	91.71
50	15.22	84.18
60	3.96	96.04
70	11.01	88.99
Mean:	8.31	91.69

E. Database indexing

Given that the aim of the designed application is to compare a query image to a sample set, it was necessary to implement an index in order to improve the query performances. We coded index choosing the first image and setting it as reference point of the features space in which each coordinate represents the similarity percentage with respect to a certain descriptor, in this case HSVDominantColor and ColorDistribution. For storing the indexed sample we made use of Apache Derby [2] database. The position in the feature space of a query image has been evaluated calculating its similarity with respect to the reference point image. The query returns all the images of the sample included in a range represented by a Gaussian function centered in the query image with standard deviation equal to $1 - t$, where t is the similarity threshold chosen by the user trough the Graphical User Interface (GUI) shown in Section IV-G.

F. Total correlation

Taking into account that ColorDistribution has a greater precision with respect to the HSVDominantColor, we define a novel scalar product in the descriptors space according to the following unitary trace matrix:

$$D = \begin{pmatrix} .9 & 0 \\ 0 & .1 \end{pmatrix} \quad (1)$$

which considers the different relevance of each descriptor, defining the total correlation between two images by the formula:

$$Similarity = \sqrt{0.9 \cdot ColorDistribution^2_{Corr}} + \\ + \sqrt{0.1 \cdot HSVDominantColor^2_{Corr}}. \quad (2)$$

We decided to make use of this particular scalar product in which HSVDominantColor is considered as correction of ColorDistribution. This scalar product is defined arbitrarily and it does not constitute a constraint to the discussion.

G. User interface

We implemented a simple GUI (Figure 2) to make the program *user-friendly* and let the user choose the sample set of images and the query. Once indexed the sample set of images, many queries can be performed quickly.

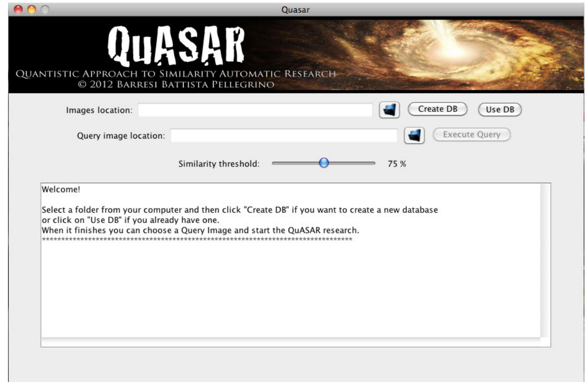

Fig. 2. Graphic User Interface implemented for the application QuASAR [21]

Fig. 3. Graphic interface showing the test results into an interactive HTML5 page

V. Results

Once the software described in Section IV has completed the indexing process, an interactive HTML5 [4] page that we have implemented shows the result's thumbnails around the query image, Figure 4, in a concentric circumference proportional to the total correlation as in Figure 3.

The results are reported in Table IV. Our implementation obtains a recognition rate (*precision*) about 95% and sensitivity rate (*recall*) about 77% with a similarity threshold set to 80% within the 3195 images sampled.

VI. Conclusion and Future Works

This paper has proposed a novel technique for performing a similarity search on an indexed sample set of images making use of new low-level color descriptors. We have adopted a quantistic approach for solving the problem of features extraction and the executed tests have demonstrated a potential improvement on efficacy of queries.

Todarello's goal was to test a linear superimposition of n-dimensional tensors [16]. Nevertheless, our work focuses on the projection of visual information onto a Hilbert space whose elements are n-grade polynomial functions.

 ISBN: 978-1-61208-265-3

TABLE IV. Precision and Recall for the 3195 processed images

Query	ActualIm	Threshold	precision	recall
Fig. 4	53	70%	0.42	1.00
		80%	0.95	0.77
		90%	1.00	0.51
		95%	1.00	0.26

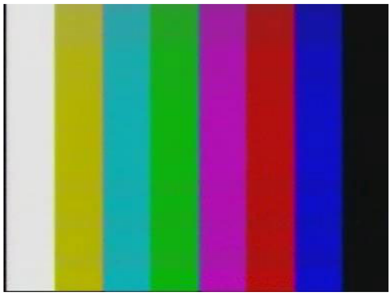

Fig. 4. Query Image used for testing the sample

The intent of this position paper is to put into practice and test the validity of the quantistic approach to similarity search we designed. Further development will consider a larger database of images, tests on the perceptive parameters obtained through survey, other classes of fitting functions and functional spaces, comparison of our results with other similarity techniques in order to define a threshold of performance. It is out of our aim, at this point, to focus onto experimental issues.

The proposed methodology can benefit the CBIR, because compared to the current techniques, making use of vector and the metric spaces where thresholds, usually evaluated experimentally, have to be applied, it enables a probabilistic approach allowing the superimposition of different results. We can foresee an improvement of the pseudo-relevance feedback querying multimedia databases.

Moreover, it is possible to implement other descriptors such as Shape (such as Textures, Edges) or Motion Descriptors in order to add more low level elements to evaluate for better image recognition. Each new descriptor can be represented by an axis in the features space.

Other MPEG-7 descriptors may be reimplemented with the formalism of Quantum Mechanics and the HSV color space described in this paper in order to enable image searches closer to the human being perception of similarity. The authors are analyzing the improvement of the retrieval results adopting more sophisticated visual descriptors as presented in [23].

Furthermore, in order to evaluate the effectiveness of the proposed methodology, it could be useful to make use of a generic publicly-available database [22], where ground-truth is available.

ACKNOWLEDGEMENT

As pointed out in Section I, we acknowledge the IRMA project for providing us with the sample set.

This work was partially supported by the international project FORGET-IT (grant no: 600826), which was funded by the EC FP7 ICT collaborative research programme, call 2011.4.3.

REFERENCES

[1] Software architecture diagram of application implemented, newton.ph.unito.it/∼barresi/UML.png [retrieved: Feb 2013].

[2] Apache derby, http://db.apache.org/derby/ [retrieved: Feb 2013].

[3] Eclipse, http://www.eclipse.org/ [retrieved: Feb 2013].

[4] Html5, http://dev.w3.org/html5/spec/single-page.html [retrieved: Feb 2013].

[5] Irma project, www.progettoirma.it [retrieved: Feb 2013].

[6] Java SE 1.6, http://www.oracle.com/it/technologies/java/ [retrieved: Feb 2013].

[7] Mpeg, http://mpeg.chiariglione.org/ [retrieved: Feb 2013].

[8] ISO/IEC 15938-3 fcd information technology - multimedia content description interface - part 3: Visual, Mar. 2003.

[9] M. K. Agoston, Computer Graphics and Geometric Modeling, Springer, 2005.

[10] G. Cannelli, Metodologie Sperimentali in Fisica, EdiSES, 2000.

[11] B. S. Manjunath, Introduction to MPEG-7, Multimedia Content Description Interface, John Wiley and Sons, Ltd., Jun 2002.

[12] J.-R. Ohm, L. Cieplinski, H. J. Kim, S. Krishnamachari, B. Manjunath, D. S. Messing, and A. Yamada, The MPEG-7 color descriptors In Introduction to MPEG-7: Multimedia Content Description Interface. Wiley, 2001.

[13] L. G. Shapiro, and G. C. Stockman, Computer Vision, Pentice Hall, 2001.

[14] A. Del Bimbo, *Visual information retrieval*, Morgan Kaufmann, 1999.

[15] T. Sikora, The mpeg-7 visual standard for content description-an overview, *IEEE Trans. Circuits Syst. Video Techn.*, vol. 11, no. 6, 2001, pp. 696–702

[16] E. M. Todarello, W. Allasia, and M. Stroppiana, MPEG-7 features in Hilbert spaces: Querying similar images with linear superpositions, Proceedings of the 5th international conference, In Quantum interaction Symposium, Jun. 2011, pp. 223–228, LNCS 7052, Berlin: Springer..

[17] K. van Rijsbergen, The Geometry of Information Retrieval, Cambridge University Press, 2004.

[18] S. Wang, L.-T. Chia, and D. Rajan, Image retrieval using dominant color descriptor, In Proceedings of the International Conference on Imaging Science, Systems and Technology, Las Vegas, Nevada, USA, Jun. 2003, pp. 107-110.

[19] E. W. Weisstein, Cone, MathWorld–A Wolfram Web Resource, May 2012.

[20] EURIX Group S.r.l., www.eurixgroup.com, [retrieved: Feb 2013].

[21] QuASAR project, newton.ph.unito.it/∼barresi/QuASAR/, [retrieved: Feb 2013].

[22] P. Budikova, M. Batko, and P. Zezula, Evaluation Platform for Content-based Image Retrieval Systems, In International Conference on Theory and Practice of Digital Libraries 2011, LNCS 6966. Berlin: Springer.

[23] O. Penatti, E. Valle, and R. Torres, Comparative Study of Global Color and Texture Descriptors for Web Image Retrieval, Journal of Visual Communication and Image Representation, vol. 23, no. 2, Feb. 2012, pp. 359-380.