Enhancing HSV Histograms with Achromatic Points Detection for Video Retrieval

Costantino Grana, Roberto Vezzani, Rita Cucchiara
Department of Information Engineering
University of Modena and Reggio Emilia
Via Vignolese 905/b, 41100 Modena, Italy
{grana.costantino,vezzani.roberto,cucchiara.rita}@unimore.it

ABSTRACT

Color is one of the most meaningful features used in content based retrieval of visual data. In video content based retrieval, color features computed on selected frames are integrated with other low-level features concerning texture, shape and motion in order to find clip similarities. For example, the *Scalable Color* feature defined in the MPEG-7 standard exploits HSV histograms to create color feature vectors. HSV is a widely adopted space in image and video retrieval, but its quantization for histogram generation can create misleading errors in classification of achromatic and low saturated colors. In this paper we propose an *Enhanced HSV Histogram* with achromatic point detection based on a single Hue and Saturation parameter that can correct this limitation. The enhanced histograms have proven to be effective in color analysis and they have been used in a system for automatic clip annotation called PEANO, where pictorial concepts are extracted by a clip clustering and used for similarity based automatic annotation.

Categories and Subject Descriptors

H3.7 [**Information Storage and Retrieval**]: Digital Libraries – *collection, standards, system issues.*

General Terms

Algorithms, Documentation, Performance.

Keywords

HSV, color space, achromatic, video annotation, video retrieval

1. INTRODUCTION

Video archives, and video digital library are becoming available both in stand-alone and web application in many different contexts: cultural heritage, medicine, news and broadcasting, sports, biometrics and surveillance are typical examples. A growing need is the ability to search for videos based on their content instead of relying on manually provided metadata. The diffusion of such systems has been strongly limited by the difficulty to generalize results of visual and audio automated processing techniques obtained on tuned test data sets. On the other hand, internet users are very demanding on search results, so the media search technologies for the mass have to be very intuitive and easy to use such as text retrieval is [1].

From the technical point of view, another question is whether we should go for domain-dependent features or for more general ones, defined at perceptual level only. This last choice could be probably less effective than ad-hoc defined features, but is potentially applicable to wider scenarios. Moreover, general purpose low level features can be used not only to search for perceptual similarity but also to be composed in classifiers to detect visual concepts. The TRECVID workshops shown several interesting prototypes and novel solutions based on the analysis of some primitives such as color, texture, and motion.

Examples of automatic annotation systems have been presented recently, most of them in the news and sports application domain. Most of the proposals deal with a specific context making use of ad-hoc features. In [2] the playfield area, the number and the placement of players on the play field, and motion cues are used to distinguish soccer highlights into subclasses. Differently, a first approach trying to apply general features is described in [3]. Employing color, texture, motion, and shape, visual queries by sketches can be performed, supporting automatic object based indexing and spatiotemporal queries.

In the same direction of [3], we propose a general framework called PEANO (Pictorially Enriched ANnotation with Ontologies) [4] which allows to automatically annotate video clips by comparing their similarity to a domain specific set of prototypes. In particular, we focus on providing a flexible system directly applicable to different contexts and a standardized output by means of the MPEG-7 tools. The clip characterizing features, the final video annotation, and the storage of the reference video objects and classes are realized using this standard.

While the importance of color feature is straightforward, the selection of the best color features and color space is still an open issue. In particular, the MPEG-7 Scalable Color Descriptor feature has been tested and proved to be misleading for achromatic and dark colors, since it is based on the HSV color space. To this aim, we propose in this paper an enhanced HSV color histogram able to explicitly handle achromatic colors.

The paper is organized as follows: in Section 2 similarity of video clips is described, Section 3 focuses on the HSV color space

CIVR'07, July 9-11, 2007, Amsterdam, The Netherlands.

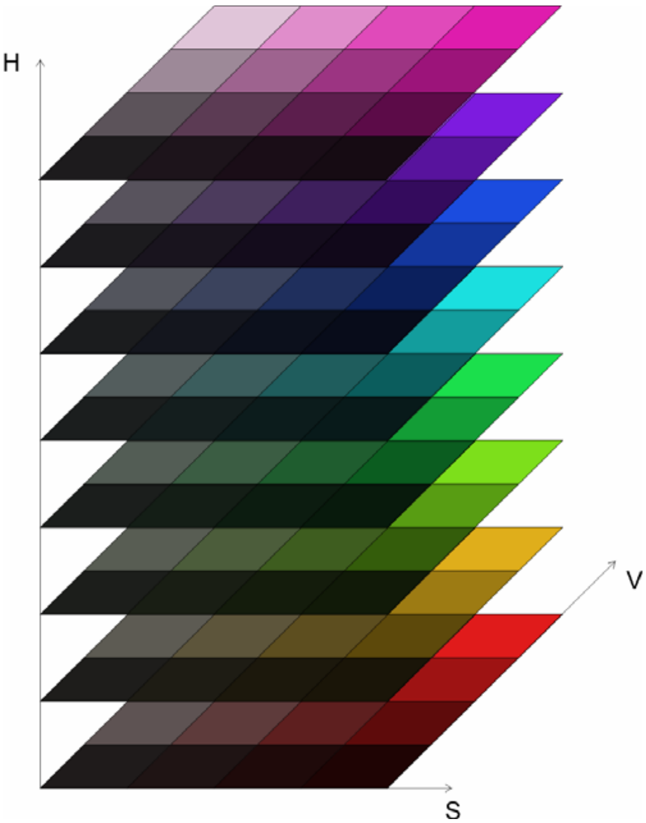


Fig. 1. Quantized view of the HSV color space. S and V are quantized to 4 values, while H to 8 values. The bins with lowest V and S present barely noticeable visual differences.

problems and our solution is proposed in Section 4. A nearest neighbor automatic annotation, a prototype creation algorithm, the proposed index for automatic level selection, and a context based dissimilarity measure are presented in Section 5. Results over sports and news videos are reported in Section 6.

2. SIMILARITY OF VIDEO CLIPS

The problem of clip similarity can be seen as a generalization of the problem of image similarity: as for images, each clip may be described by a set of visual features, such as color, shape and motion. These are grouped in a feature vector:

$$\mathbf{V}_i = \left[F_i^1, F_i^2, \ldots, F_i^N \right] \tag{1}$$

where i is the frame number, N is number of features and F_i^j is the j-th feature computed at frame i. However, extracting a feature vector at each frame can lead to some problems during the similarity computation between clips, since they may have different lengths; at the same time keeping a single feature vector for the whole clip cannot be representative enough, because it does not take into account the features temporal variability. Here, a fixed number M of feature vectors is used for each clip, computed on M frames sampled at uniform intervals within the clip. In our experiments, a good tradeoff between efficacy and computational load suggests the use of $M = 5$ for clips of

Fig. 2. Example of the HSV problem. The image on the right, shows in different colors pixels of the original image assigned to different histogram bins. Dark pixels which are visually very similar, fall in different bins when the color space is quantized.

averaging 100 frames. Thus, the distance between two clips S_u and S_v is defined as

$$d\left(S_u, S_v\right) = \frac{1}{M} \sum_{i=1}^{M} \left\| \mathbf{k}^{\mathrm{T}} \left(\mathbf{V}_{u_i} - \mathbf{V}_{v_i} \right) \right\| = \frac{1}{M} \sum_{i=1}^{M} \sum_{j=1}^{N} k_j \left\| F_{u_i}^{j} - F_{v_i}^{j} \right\|, \tag{2}$$

where, $\mathbf{k} = \left(k_1, \ldots, k_N\right)$ is a weight vector and u_i and v_i are the frame numbers of the i-th sub-sampled frames of S_u and S_v respectively. The weights $k_j \in [0,1]$ provide dimensional coherence among the different features and at the same time they allow to change their relative significance.

To provide a general purpose system, we avoid selecting context dependent features, relaying on broad range properties of the clips. In other works [5], to allow easier interoperability and feature reuse, we tried to select features which comply with the MPEG-7 standard [6]. In particular the Scalable Color Descriptor was selected, which is a color histogram in the HSV color space, with 16 values in H and 4 values in each S and V (256 bins in total).

3. HSV COLOR SPACE

The HSV color space is widely adopted in image and video retrieval, since it allows to separate the chromatic contribution of the image colors. Kotoulas *et al.* [7] for example use HSV color histograms for content-based image retrieval proposing hardware implementations of the algorithms. Despite that, the use of HSV color space has three well known drawbacks: (1) hue is meaningless when the intensity is very low; (2) hue is unstable when the saturation is very low; and (3) saturation is meaningless when the intensity is very low. The same consideration is reported in [8] but applied to the IHS color space, which is conceptually similar to the HSV. In other words, dark colors are insensitive to saturation and hue changes and, similarly, the hue value is negligible for achromatic colors (low saturation). Thus, the MPEG-7 Scalable Color Descriptor suffers of this defect: dark or low saturated colors can be assigned to different bins even if they are visually very similar (see Fig. 1).

To solve this problem, in [8] two regions are defined and separately treated, one for the chromatic and one for the achromatic colors. These areas are obtained with a complicated set of thresholds in the IHS color space. Similarly, in [9] a fuzzy

				
original	λ =0.05	λ =0.10	λ =0.15	**λ =0.20**
	43924 (0.3%)	184024 (1.1%)	433300 (2.6%)	**794248 (4.7%)**
				
λ =0.25	λ =0.30	λ =0.35	λ =0.40	λ =0.45
1286704 (7.7%)	1916380 (11.4%)	2683768 (16.0%)	3561454 (21.2%)	4593604 (27.4%)

Fig. 3. Selection of achromatic colors with different λ values. Our choice is highlighted with the gray background. The numbers below the λ values are the number of RGB colors which get mapped to the achromatic area.

technique has been proposed in order to distinguish among chromatic, achromatic, and black colors. Sural *et al.* [10] propose a histogram modification that takes into account the above mentioned regions. In particular, they identify the achromatic region by thresholding the saturation coordinate with a linear function of the intensity value and based on the outcome chose to represent the color with its hue or its value only. In [11] a detailed comparison of the MPEG-7 color descriptors can be found, proving that the Scalable Color Descriptor is not suitable for monochromatic images.

In the following chapter we propose an enhancement to the traditional HSV histograms that takes into account the achromatic regions overcoming the MPEG-7 Scalable Color Descriptor drawbacks.

4. ENHANCED HSV HISTOGRAMS

The Scalable Color Descriptor requires a quantization of the HSV color space, with 16 values in H and 4 values in each S and V (256 bins in total). Supposing every color channel in the range [0,1), the bin index may be obtained as:

$$\text{bin} = f\left(H,S,V\right) = \lfloor n_H H \rfloor n_S n_V + \lfloor n_S S \rfloor n_V + \lfloor n_V V \rfloor \qquad (3)$$

where n_H, n_S, n_V are the quantization levels devoted to every color channel. Usually these are chosen to be powers of 2 for ease of representation.

Adopting a linear quantization of each coordinate leads to have, for example, 64 different bins for the darkest colors characterized by the lowest values of V. Thus, a visually uniform background can be split on different bins (see Fig. 2). We propose to add n_A bins to the HSV histogram that contains all the achromatic and dark colors. These n_A bins correspond to gray levels, from black to white; for convenience, we choose to set n_A=n_V=4, as the number of levels assigned to the V axis in the MPEG-7 standard. The dark and achromatic colors are selected by imposing a unique threshold λ on the S and V coordinates respectively, as reported in Table 1. In the third column, the index computation for the new bins is reported.

The value of λ has been empirically set to 0.2. In Fig. 3 different thresholds have been selected and the reconstructed images obtained by substituting each color with the center of the corresponding bin are shown. Fig. 3 contains the number and the absolute percentage of RGB colors that are classified as dark or achromatic after the HSV conversion and quantization. λ =0.2 proved to be a good tradeoff between color loss and matching of similar dark or achromatic colors.

In Fig. 4 the obtained results over four sample images are reported. In the first column the input images are shown, while in the second column the image segmentation obtained with the quantization of the MPEG-7 Scalable Color Descriptor is drawn assigning a different random color to each bin. The background color of the first and of the third row images, the sea in the second row, and the hat in the last row are some examples of dark or achromatic uniform areas that are split into different bins. Chromatic areas (obtained with λ =0.2) are marked in red on the rightmost column images, leaving original colors in the dark or achromatic areas. In the third column, the results of the Enhanced HSV Histogram are reported, where the same random colors as before are used for the chromatic area of the histogram, while gray levels are employed for the achromatic area.

Moving some of the colors from the original bins to the n_A achromatic ones makes these original bins less used with respect to the others. In fact, it doesn't make sense anymore to uniformly subdivide the S and V channels, if part of it is then discarded. A better solution is to fully employ the chromatic bins to describe

Table 1. Rules for the HSV regions selection.

Condition	*Region*	*Bin*
$S < \lambda$	achromatic	bin = $n_H n_S n_V + f(0,0,V)$
$V < \lambda$	dark	bin = $n_H n_S n_V + f(0,0,V)$
otherwise	chromatic	bin = $f(H,S,V)$

Fig. 4. Enhanced HSV Histogram results over four sample images are reported. The columns show respectively: input images, segmentation obtained with the HSV quantization, gray levels assignment for the achromatic area, chromatic area masking (λ=0.2).

only the effective chromatic area. To this aim we linearly quantize the remaining HSV space, by simply redefining the *f* function of Eq. 3:

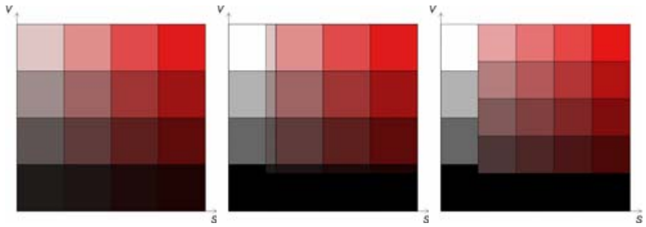
$$f'(H,S,V) = f\left(H, \frac{S-\lambda}{1-\lambda}, \frac{V-\lambda}{1-\lambda}\right) \tag{4}$$

For the achromatic area the original *f* function is still used, since the whole range of V has to be quantized. A graphical visualization of the effect of Eq. 4 is given in Fig. 5. In Fig. 6 the reconstruction of the images after the quantization with the two different functions is reported. Clearly the second approach gives a better match with the original chromatic area.

A different approach could be used to reduce the number of bins in the histogram, without affecting the chromatic area. The threshold λ can be set to $1/n_V$, thus making the achromatic area exactly match the first set of bins for S and V. This forces these bins to 0, thus allowing their removal. This indeed induces a compression with respect to the color representation, but it is selectively applied to the least significant colors. For example with reference to the aforementioned 16,4,4 subdivision, this would lead to $16 \times 3 \times 3 = 144$ bins, plus 4 bins for the achromatic area.

5. AUTOMATIC ANNOTATION

Given a domain-specific video digital library, we assume that it is possible to partition the clips of videos of that context into a set of L classes $\mathbf{C} = (C_1, \ldots, C_L)$, which describe different contents or camera views .Given a large set of training clips, we implemented an interactive user-friendly interface named PEANO [4] to



a) b) c)

Fig. 5. a) Original quantization of the SV plane with H=0 (red); b) Achromatic area detection (λ=0.2); c) linear re-quantization of the chromatic area.

quickly assign each of them to a specific class C_k and then employ it for automatic annotation purposes. The name PEANO (Pictorial Enriched ANnotation with Ontologies) is in the honor of the famous Italian mathematician Peano that is one of the founders of mathematical logic and the first one that used the name "class" to say prototype or set of elements of the same type. The PEANO tool allows a user friendly fast annotation of clips by means of the classes defined in a textual taxonomy or in an ontology. It can be used to divide large training set of clips in manually defined classes. In video clips, the semantics associated with a class cannot be often represented with a single perceptual aspect but normally many clips of the same class have a different pictorial aspects [2]. This is well known, and automatic annotation by similarity is hard to be performed against a single prototype for each class. Many approaches use many examples for each class. The PEANO module uses a hierarchical clustering method to reduce the number of clip of each class, in a variable number of prototypes (or pictorial concepts) keeping only some representatives, which capture the most significant aspects of a set of clips.

An unknown clip can be classified using a nearest neighbor approach and the similarity measure defined above. The weights k_i of Eq. 2 may be tuned to optimize the classification results on a training video, by searching the set which provides the maximum number of correct class assignments. This process is done once for all during the context classifier design, so the optimization phase is not time constrained and an exhaustive search can be employed.

5.1 Intra-class clustering

As above mentioned, since not all the clips are equally important to obtain the final classification due to perceptual redundancy in specific domains, we employ a hierarchical clustering method, based on *Complete Link* [12], to reduce the number of clip of each class, keeping only some representative prototypes, which capture the most significant aspects of a set of clips. This technique guarantees that each clip must be similar to every other in the cluster and any other clip outside the cluster has dissimilarity greater than the maximum distance between cluster elements. For this clustering method a dissimilarity measure between two clusters W_i and W_j is defined as

$$\Delta(W_i,W_j) = \max_{S_x \in W_i, S_y \in W_j} d\left(S_x, S_y\right). \tag{5}$$

where *d* is computed as in Eq. 2.

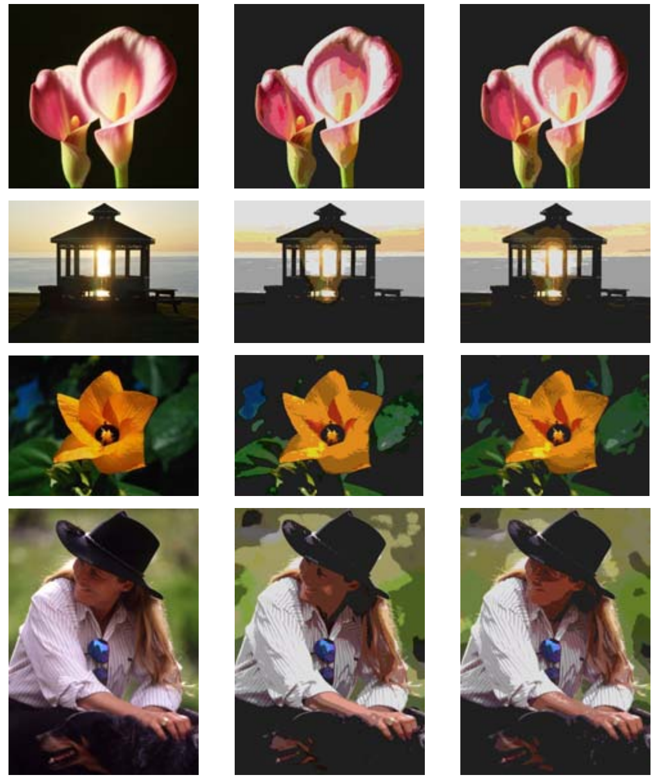


Fig. 6. Reconstruction of the images in the left column, after the quantization with the functions f (center) and f' (right).

The algorithm proceeds as follows:

1. For each class C_k do steps from 2 to 5:
2. Initially each cluster contains a single clip. Let us call E_n the set of clusters at level *n*, and initialize it to $E_{P_k} = \left\{\{S_1\},\{S_2\},\ldots,\{S_{P_k}\}\right\}$, with $P_k = card\left(C_k\right)$.
3. The least dissimilar pair of clusters, $W_i, W_j \in E_n$, is found according to Eq. 5, i.e. $\Delta\left(W_i,W_j\right) \le \Delta\left(A,B\right)\ \forall A,B \in E_n$.
4. W_i and W_j are merged into the same cluster and E_{n+1} is accordingly updated.
5. If everything is merged into a single cluster or a stop condition is met, the algorithm goes to the next class, otherwise it resumes from step 2.

For each class, this algorithm produces a hierarchy of clips partitions with P_k levels, where level 1 is the final step where everything is merged in a single cluster. To implement the algorithm, a proximity matrix was used: initially, it contains the distances between each pair of clips. At each step, the matrix is updated by deleting rows and columns corresponding to the selected clusters and adding a new row and column corresponding to the merged cluster. The values in the new row/column are the maximum of the values in the previous ones.

5.2 Automatic clustering level selection

Instead of a manual selection of the desired clustering level, or a threshold guided one, an automatic selection strategy is proposed. Such a rule has to be based on cluster topology concerns, being a trade-of between data representation and small number of clusters, and it is not possible to choose the *right* one. In literature different proposals are presented, such as the Dunn's Separation Index [13], but the corresponding results on our data sets were not satisfactory. Better results (in terms of a subjective evaluation) have been obtained with the following approach. Let us define the cluster distance as

$$\delta(W_i, W_j) = \min_{S_x \in W_i, S_y \in W_j} d\left(S_x, S_y\right). \tag{6}$$

The *Clustering Score* at level n is defined as

$$CS_n = \min\left(\Delta_1 - \Delta_n, \delta_n\right) \tag{7}$$

where

$$\begin{aligned} \Delta_n &= \max_{W_i \in E_n} \Delta\left(W_i, W_i\right) \\ \delta_n &= \min_{W_i, W_j \in E_n, i \neq j} \delta\left(W_i, W_j\right) \end{aligned} \tag{8}$$

The selected level is the one which maximizes CS_n. It is possible to observe that Δ_n and δ_n are both monotonically increasing with n, thus CS_n has a unique global maximum. Therefore, the clustering algorithm can be stopped when CS_n start to decrease without computing the remaining levels. A single prototype can be generated from each cluster, by computing the M average feature vectors. The clip which minimizes the distance from the prototype features is associated to it, in order to provide a representative of the visual concept.

5.3 Intra-class clustering with context data

In section 5.1 an intra-class clustering has been presented in order to generate a set of significant prototypes for each class. The choice is guided by how similar the original clips are in the feature space, without considering the elements belonging to the other classes (*context data*). This may lead to a prototype selection which is indeed representative of the class but lacks the properties useful for discrimination purposes. To cope with this problem, we define an *isolation coefficient* for each clip as:

$$\gamma\left(S_u\right) = \sum_{i=1, i \neq j}^{L} \sum_{S_v \in C_i} \frac{1}{d\left(S_u, S_v\right)}, S_u \in C_j. \tag{9}$$

Then we can introduce a class based *dissimilarity measure* between two clips as:

$$\bar{d}\left(S_u, S_v\right) = d\left(S_u, S_v\right) \cdot \gamma\left(S_u\right) \cdot \gamma\left(S_v\right). \tag{10}$$

The *intra-class complete link* clustering is thus enhanced with context data by substituting the clips distance in Eq. 5 and Eq. 6 with this dissimilarity measure.

Table 3. Results of Self Annotation on three different videos.

Video	# Frames	# Clips	Correct
Ski	178.181	1212	90%
Bob	50.835	1422	92%
F1	215.638	2339	82%

6. EXPERIMENTAL RESULTS

As a first test a self-annotation task was performed, i.e. we automatically classify the clips of a video using a leave-one-out cross validation (i.e., each clip is classified exploiting all the other clips of the same video). Since this corresponds to counting how many clips belong to the same class of their nearest neighbor, we can check how separable the defined classes are in the selected feature space. For the second test, we checked the generalizing properties of the context classifier by using two different sets for

Table 2. Confusion matrix of the results on the second test, for a soccer video.

Confusion matrix	Test set											
GT	bench	close-ups	entrance in the playing field	far from goal	from stands	from tribune	near to goal	corner view	spectators	subentry	team	Total
bench	7	7					2		3	2		21
close-ups	3	63		7	1	4	3		22	1		104
entrance in the playing field		1		1			1		1			4
far from goal		5		37		6	4		2		2	56
from stands				3	5	4						12
from tribune		2		7	2	44						55
near to goal				3	1	3	10		1		1	19
corner view		1				4		31				36
spectators		1		1					15	1		18
subentry		1		1								2
team		2	1	4			1		4		2	14
Total	10	83	1	64	9	65	21	31	48	4	5	341

the training and test phases.

In Table 3 the results of the first test on three different videos are reported, confirming that the selected feature space is quite effective in the considered contexts. In particular, the ski context presents an higher degree of repetitiveness, so the visual classification performs better than in the F1 one.

Other tests have been performed on soccer videos taken from the 2006 World Championship. A taxonomy of 11 classes has been used to characterize the different clips in terms of content (playing area or bench, spectators, etc…) or point of view (close ups, tribune view, etc…). To create training sets for different contexts, we used a first part of each video and the rest was used as test. All the training clips have been reduced by the prototype creation algorithm and the number of the generated prototypes is about a tenth of the initial samples. On the test set, it is possible to see that the use of this approach reaches a classification rate around 65% on average, depending on the class. In Table 2 the confusion matrix of one experiment is reported. Looking at the table, it is possible to see that the classification provides good results with specific and well constrained classes (e.g. corner view), while most of the errors come from the confusion between content and view point classes. For example, the spectators class contains some near view of people, which may be easily confused with players close ups, by a non specific feature.

7. CONCLUSIONS

We presented an approach for HSV color space analysis and enhanced histogram generation. The proposed enhancement allows to better describe the chromatic area and to avoid meaningless assignment by hue for low saturated and dark colors. In particular an achromatic area detection has been defined. This feature has been tested on a general purpose system for the automatic annotation of video clips. A process for reducing space and computational requirements by the creation of prototypes with context based intra-class clustering was described. The annotation has shown reasonable results without domain specific feature development. This approach allows a system to behave differently by simply providing a different context, thus expanding its applicability to mixed sources digital libraries.

ACKNOWLEDGMENTS

This work is supported by the DELOS NoE on Digital Libraries, as part of the IST Program of the European Commission (Contract G038-507618).

REFERENCES

[1] Jaimes, A., Christel, M., Gilles, S., Sarukkai, R., and Ma, W. 2005. Multimedia information retrieval: what is it, and why isn't anyone using it?. In Proceedings of the 7th ACM SIGMM international Workshop on Multimedia information Retrieval (Hilton, Singapore, November 10 - 11, 2005). MIR '05. ACM Press, New York, NY, 3-8. DOI= http://doi.acm.org/10.1145/1101826.1101829

[2] Bertini, M., Cucchiara, R., Del Bimbo, A., and Torniai, C. 2005. Video Annotation with Pictorially Enriched Ontologies, Proceedings of IEEE International Conference on Multimedia and Expo, Amsterdam, The Netherlands, 1428-1431.

[3] Chang, S., Chen, W., Meng, H.J., Sundaram, H., and Zhong, D. 1998. A Fully Automated Content-Based Video Search Engine Supporting Spatiotemporal Queries, IEEE Transactions on Circuits and System for Video Technology, vol. 8 num.5, 602-615.

[4] Grana, C., Vezzani, R., Bulgarelli, D., Barbieri, F., Cucchiara, R., Bertini, M., Torniai, C., and Del Bimbo, A. 2006. PEANO: Pictorial Enriched ANnotation of VideO. Accepted for publication in Proceedings of the 14th ACM international Conference on Multimedia (Santa Barbara, CA, United States, October 23-27, 2006). http://imagelab. ing.unimo.it/imagelab/peano.asp

[5] R. Vezzani, C. Grana, D. Bulgarelli, R. Cucchiara. 2006. A semi-automatic video annotation tool with MPEG-7 content collections" in Proceedings of IEEE International Symposium on Multimedia (ISM2006), San Diego (CA), USA, pp. 742-745, Dec. 11-13.

[6] Information technology - Multimedia content description interface - Part 3: Visual, ISO/IEC Std. 15 938-3:2003, 2003.

[7] L. Kotoulas, I. Andreadis. 2003. Color Histogram Content Based Image Retrieval and Hardware Implementation", IEE Proc. Circuits Devices & Systems Vol. 150: 387-93, Issue 5, 6 Oct. 2003.

[8] D. C. Tseng, C.H. Chang. 1992. Color segmentation using perceptual attributes. In Proceedings of 11th IAPR International Conference on Pattern Recognition, Vol.III. Conference C: Image, Speech and Signal Analysis, pp. 228-231.

[9] M. Seaborn, L. Hepplewhite, J. Stonham. 1999. Fuzzy colour category map for content based image retrieval. In Proceedings of the British Machine Vision Conference, BMVC99.

[10] S. Sural, Gang Qian, S. Pramanik. 2002. Segmentation and histogram generation using the HSV color space for image retrieval. In Proceedings of the International Conference on Image Processing. 2002, Vol.2, pp II-589- II-592.

[11] H. Eidenberger. 2004. Statistical analysis of MPEG-7 image descriptions, ACM Multimedia Systems Journal, Springer, Vol. 10, No. 2, pp. 84-97.

[12] Jain, A.K., and Dubes, R.C. 1988. Algorithms for clustering data, Prentice-Hall, Englewood Cliffs, NJ.

[13] Dunn, J.C. 1973. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, Journal of Cybernetics, vol. 3 num. 3, 32-57.