Textual Descriptors for Browsing People by Visual Appearance

Francesc Tous, Agnés Borràs, Robert Benavente,
Ramon Baldrich, Maria Vanrell, and Josep Lladós

Computer Vision Center, Dept. Informàtica.
Universitat Autònoma de Barcelona,
08193 Bellaterra (Barcelona), Spain
{ftous,agnesba,robert,ramon,maria,josep}@cvc.uab.es

Abstract. This paper presents a first approach to build colour and structural descriptors for information retrieval on a people database. Queries are formulated in terms of their appearance that allows to seek people wearing specific clothes of a given colour name or texture. Descriptors are automatically computed by following three essential steps. A colour naming labelling from pixel properties. A region segmentation step based on colour properties of pixels combined with edge information. And a high level step that models the region arrangements in order to build clothes structure. Results are tested on large set of images from real scenes taken at the entrance desk of a building.

Keywords: Image retrieval, textual descriptors, colour naming, colour normalization, graph matching.

1 Introduction

This work presents a people description module that is a part of a general surveillance system. Images of people entering a building are processed while they are checking-in. Textual descriptors based on people appearance are extracted from these images. This information is saved in a global database where the security personnel of the building can make queries. This might be useful if they can see in a camera inside the building someone who is causing problems, and they want the information that identifies this person. Here is where our module acquires importance, because in our database there is information about the appearance of the people who have entered the building and that has been automatically extracted. With this purpose, the system allows the user to make queries formulated in terms of textual descriptors, to retrieve those images from the database agreeing with the descriptors. Queries are formulated in terms of colour, texture and structural properties of clothes that people is wearing. The system will automatically build an appearance feature vector from an image acquired while people is checking-in in front of the desk.

Retrieving images from large databases using image content as a key is a largely studied problem in Computer Vision. Two major approaches can be

M.T. Escrig Monferrer and F. Toledo Lobo (Eds.): CCIA 2002, LNAI 2504, pp. 419–429, 2002.

stated. First, similarity retrieval consists in looking for images in a database using a reference image as query. The second approach concerns browsing applications and consists in retrieving images by pictorial content, i.e. using symbolic descriptors as indices. Concerning to features used as the basis to formulate queries, usually early visual primitives such as colour and texture are used. Sometimes, structure of objects in the image is also important. A number of works combine low level visual cues as color and texture with higher level information such as structure (e.g. [4,11,15]). Our work follows this approach. Queries are formulated in terms of textual descriptors like 'we are looking for a man in a red shirt' that are compared with descriptions stored in the database that were previously extracted from the input images.

The approach we present in this paper focus on a computational descriptor of clothes features that is based on a three-step process. Firstly, a colour feature vector of pixels is computed, this colour naming step will be the basis of the further analysis. After, a first region initialisation and considering colour properties of pixels plus edges information, a merging process is proposed, it will allow to model any image region. Finally, a high level interpretation of the image regions allows to model an structural description on the clothes that people are wearing. Some examples of content-based queries are shown, they help to illustrate how the proposed descriptor can behave on the system presented.

The paper is organised as follows: section 2 presents how colour has been modelled to provide with a discrete labelling of image pixels, afterwards, in section 3 a region modelling step is done towards to build in section 4 an structural interpretation of people clothes; and sections 5 and 6 briefly present some examples on how the descriptor behaves and a short discussion on them.

2 Colour Modelling

As we have already introduced in the previous section, the final aim of this work is to build with a browsing application based on textual descriptors. One of the most usual ways to describe people appearance is by using colour names for clothes. Therefore, it is quite common to add colour adjectives to clothing articles in order to better specify a visual description.

The association of a colour term or category to a colour perception is a common activity that humans do. The complexity of this process has confronted researchers from visual science to anthropological linguistics for the last twenty years since the book of B. Berlin and P. Kay was firstly published [2]. An excellent compendium of all these research studies can be found in [6], where colour naming can be seen a multi-disciplinary and huge issue.

In this work, we use a statistical colour naming model to build high-level descriptions of scenes. From an engineering point of view we need a colour-naming module that automatically assign colour terms to image regions fulfilling two important requirements: colour categorisation of the model has to correlate with human perceptual judgements and it has to demonstrate invariance to colour and intensity illuminant changes.

To this end, we have based our model on perceptual data we have collected from a psychophysical experiment explained in [1]. In this experiment we have assumed that colour categories can be represented as fuzzy sets and therefore a colour is defined by the membership degrees to the basic colour terms as has been proposed in [8].

To consider the colour constancy problem, that is, we need to assure that colour naming process will be invariant lighting changing conditions of a real-world scene. We will remove this dependency by using the comprehensive colour normalization proposed in [3], and adapted to this problem in [16] since certain constancy on background content can be assumed. Comprehensive normalisation provides a chromatic image representation that present good colour constancy properties. Because, the intensity of image regions is needed on for some specific colours, it will be also normalised and used separately.

Our model will allow to distinguish eight different chromaticities: grey, blue, green, yellow, orange, red, pink, purple. Some of them will be divided in different colour names depending on a normalised intensity value. Thus, grey will provide three different colours: black, grey and white; and within orange chromaticities we will distinguish: dark brown, orange and light brown, which are usual colours when describing clothes. A thresholding process on the normalised intensity space can provide up to twelve distinct colour names, however only chromaticities will be modeled and intensity will be used in the region growing step we present in the following section.

After colour normalisation has been performed the *RGB* representation will be projected on the 2-D space, we will call **u-v** space, that contains the plane of the chromaticity coordinates. Its origin is located at $(0, 0, 1)$ and the axis directions are given by the vectors $(1, 0, -1)$ and $(-\frac{1}{2}, 1, -\frac{1}{2})$. It is the space where we fit our colour gaussian model.

Our model of colour naming is inspired by the gaussian model of Lammens [9], that is applied to a 3D chromatic space. We use a multivariant gaussian model that is better adapted to our 2D normalised space. The main idea of the model is to work out the parameters of a multivariant gaussian function for each chromaticity, $\mathbf{x}$, and is given by

$$G_{\mu,\mathbf{\Sigma}}(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^d}\sqrt{\|\Sigma\|}} e^{-\frac{1}{2}(\mathbf{x}-\mu)^T \mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)} \tag{1}$$

where μ is the vector (μ_1, μ_2) and $\mathbf{\Sigma}$ is a symmetrical matrix given by

$$\mathbf{\Sigma} = \begin{pmatrix} \sigma_0 & \sigma_1 \\ \sigma_1 & \sigma_2 \end{pmatrix} \tag{2}$$

Estimation of μ and $\mathbf{\Sigma}$ is achieved by minimizing a similar expresion as the one proposed by Lammens in [9], that considers the set of points in the convex hull of each colour, the center of the current colour and the centers of the rest of the colours. The fitting process is done on a sample of 248 labelled colour points. The obtained parameters are given in table 1.

Table 1. Estimated parameters, μ and Σ.

Colour	CId	μ_1	μ_2	σ_0	σ_1	σ_2
Grey	1	0.447963	0.657606	0.004601	-0.000544	0.000862
Blue	2	0.366035	0.689680	0.001789	-0.000701	0.001396
Green	3	0.383917	0.577454	0.008190	0.003398	0.006287
Yellow	4	0.568662	0.595679	0.008864	-0.001249	0.001595
Orange	5	0.571394	0.675418	0.007354	0.001578	0.003556
Red	6	0.572352	0.758278	0.008407	0.005216	0.004064
Pink	7	0.474046	0.720353	0.002485	0.001841	0.002460
Purple	8	0.419032	0.715220	0.000389	-0.000038	0.001498
Skin	9	0.8641	0.3961	0.004703	-0.000601	0.000792

Due to the specific character of the application in which this work is framed, we will add the skin colour model. It is used in the structure interpretation step, and it will be indispensable in further extensions of the clothes interpretation. Thus, skin color model has been estimated from a set of 11250 samples of skin image regions. The skin colour sample is mapped in the **u-v** space in figure 1.(a), and in figure 1.(b) we can see the guassian distribution of this skin sample, that validates the model used.

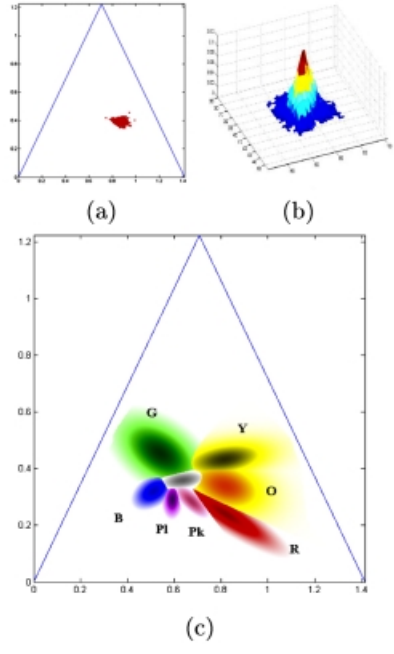

Fig. 1. Colour modelling. (a) Skin samples on **u-v** space, (b) 2D histogram of the skin sample. (c) Adjusted colour gaussian model.

Once, the colour models have been adjusted we can build a complete colour descriptor, C, for any pixel, $I_{x,y}$, of an image I, by

$$C(I_{x,y}) = (p^1_{x,y}, ..., p^9_{x,y}) \tag{3}$$

where $p^i_{x,y} = G_{\mu,\mathbf{\Sigma}}(x, y)$, being the parameters μ and $\mathbf{\Sigma}$ of the colour identified by i in table 1.

3 Region Modelling and Segmentation

Colour is an early visual primitive that is used as a coarse indexing cue to retrieve images from the database. This would allow to look for people wearing clothes of a certain colour. In a finer indexing mechanism we look for region structure in the image. With this purpose, once pixels have been colour labeled, they are hierarchically clustered in order to describe the image in terms of a region relational structure. The segmentation of these regions is formulated in terms of colour and texture homogeneity. This relational description of images in terms of regions labeled by basic colour terms would allow to convert textual description queries to a bi-level indexing formulation, namely, colour and structure for browsing images agreeing with the description.

Thus, we define a *region relational graph* $G = (R, E, L_R, L_E)$ where R is a set of image regions obtained after a segmentation process, E is a set of graph edges representing region relationships, and L_R and L_E are two labeling funtions for graph nodes (image regions) and edges respectively defined as follows. Given an image region, $r \in R$, the region labeling function is defined as $L_R(r) = [T(r), BB(r), P(r), A(r), C(r), Z(r)]$. Where the former attributes represent, respectively, type, bounding box, position, area, colour and zone of the region, r. Among the attributes that characterize a region, let us further describe *Type*, *Colour* and *Zone*. $T(r)$ identifies wether a region is textured or filled by an homogeneous color (referred as plain region in this paper). $C(r)$ is a vector attribute that represents a color of a region as it has been defined in Eq. 3 for an image pixel. Finally, $Z(r) = \{face|hair|hands|clothing\}$ classifies a region as belonging to one of the four classes in which the image is presegmented using information of skin colour and region position. The edge labeling function L_E stores information about region adjacency,

Once we have described the attributed graph model that represents an image in terms of its regions and drives the retrieval, let us describe the segmentation process that computes such graph from an input image. Informally speaking, the process can be described as a region growing in which at each iteration those neighbouring regions having a similar color are clustered in a new region. Actually, the region growing is organized in two stages: first, a presegmentation step performs an initial construction of a region adjacency graph in which colour and texture homogeneity is used as a cue to extract the regions. The second step is a graph clustering procedure that iteratively merges graph regions in terms of the similarity of their attributes defined by L_R.

Image Presegmentation

The presegmentation process is mainly focused on the discrimination between textured and plain regions. Following the idea given by Karu et al. [7], we define a pixel to be candidate to belong to a textured region if there exists a spatially uniform distribution of local gray-value variations around it. We state this variations in terms of the density of contour pixels in a certain region of interest. Thus, we use the boundary information to come apart textured regions from the plain ones. Specifically we apply the Canny edge detector to distinct between contour pixels and non contour pixels. A pixel will be candidate to belong to a textured region if we can not find any area of K non contour pixels adjacent to it, being the threshold K preset in terms of the desired allowable density of the texture. Once textured regions have been discriminated, a colour quantization is applied to the remaining pixels to initialize plain regions. We use the Prosise's method [13] that is based on the Gervautz algorithm [5] for colour quantization. The presegmentation process is illustrated in Fig. 2.

Fig. 2. Original image, contour image, quantization, initial regions.

Construction of the Region Relational Graph

The presegmented image is structured in a region relational graph G according to the definition given above. Two issues must be further described at this point, namely, given a graph region r, how the region attributes $C(r)$ and $Z(r)$ are computed. Given a region r, $C(r)$ is computed by the average of the probabilities of the pixels of the region of being i-labeled. Equivalently to image pixels (see eq. 3), we can see the colour of a region as a vector of probabilities of length N where in each position we have the probability of the region of being i-labelled, i.e. $C(r) = \boldsymbol{c} = [P_1, ...P_N]$. The attribute $Z(r)$ classifies a region as belonging to one of the four parts of the person. We use the information of the colour and the position of the region. Concerning to colour, we discriminate between skin-labeled or not skin-labeled regions. We say that a region is skin labeled when the maximum probability in the vector $C(r)$ is associated to the skin label. We decide if a region is located in the lower part of the image or in the upper depending on the position of the center of $BB(r)$ regarding to the center of the bounding box that comprises all the image regions. Thus, given a region r, we decide $Z(r)$ depending on its skin label and the position with regard to the upper-lower part of the image.

Graph-Based Region Growing

The analysis of the colour distribution is one of the main methods to segment an image. However in some cases the colour information is not complete enough and is combined with edge information. Depending on when contours are combined with colour we can distinguish between embedded segmentation or post-processing segmentation [12]. Our region segmentation process belongs to the first class.

At this point we have a set of segmented regions organized in a graph. A graph clustering algorithm is then applied by iteratively merging neighbouring regions. Informally speaking the region growing process can be described as a graph contraction iterative procedure such that, at each iteration, two neighbouring regions (graph nodes) are merged according to a similarity criterion formulated in terms of colour similarity and the significance of image contours between regions.

Let G^0 be a region graph obtained after the presegmentation step described above. Formally, the region growing can be described as a graph hierarchy $G^0 \subset \ldots \subset G^n$ such that at iteration i two regions $r_a, r_b \in R^i$ are merged in a new region $r_c \in R^{i+1}$ if their distance $D(r_a, r_b)$ is under a given threshold T. The distance D is formulated combining information about colour and contours as follows:

$$D(r_a, r_b) = w_1 D_1(r_a, r_b) + w_2 D_2(r_a, r_b)$$

where D_1 is the colour distance, D_2 is the boundary distance, and w_1 and w_2 are two weighting factors empirically set. The colour distance is defined in terms of the distance between the colour probability vectors as follows:

$$D_1(r_a, r_b) = \sum_{k=1}^{N} |(C(r_a) - C(r_b)|$$

Concerning to the second merging criterion, in order to decide to join two candidate regions we also analyse the presence of contours in their common boundary. Then we establish a measure that relates the amount of contours between the two regions with regard to the length of this boundary. We define the common boundary between regions r_a and r_b as the set of pixels $\in r_a$ that have at least one adjacent pixel $\in r_b$. Let us denote as $LCB(r_a, r_b)$ as the cardinality of the set $CB(r_a, r_b)$. On the other hand, let B be the contour pixels of the image provided by the Canny edge detector (but now using more relaxed values of the parameters than the values we have used to detect textured regions). Let $LCBB(r_a, r_b)$ be the number of contour pixels at the boundary between r_a and r_b, i.e. the cardinality of the set $CB(r_a, r_b) \cap B$. Then, D_2 is defined as follows:

$$D_2(r_a, r_b) = \frac{LCBB(r_a, r_b)}{LCB}$$

We must notice that either textured and plain regions are merged according to the former criteria.

When the two criteria allow to merge two neighbouring nodes, a new instance of the graph is generated at level $i + 1$ such that regions r_a and r_b have been

joined in a new region r_c. The new graph node representing r_c is connected by new graph edges in E^{i+1} to those nodes of R^i such that r_a and r_b were connected. The region attributes $L_R^{i+1}(r_c)$ are computed as the combination of the attributes $L_R^i(r_a)$ and $L_R^i(r_b)$. Particularly, the computation of the colour attribute of the new region is computed as follows:

$$C(r_c) = C(r_a)\frac{A(r_a)}{A(r_a) + A(r_b)} + C(r_b)\frac{A(r_b)}{A(r_a) + A(r_b)}$$

The region growing steps are illustrated in Fig. 3.

Fig. 3. Final regions, zones (hair, face, clothing), labelled image, average RGB.

4 Structure Interpretation

The region-based information encoded in the graph obtained after segmentation is used to match known clothing configuration models. These clothing configuration is the basis for the formulation of queries. The interpretation of the region structure as the description of the clothing must adjust to several predefined models or classes. These classes are formulated in terms of the number or garments, their position and their size. We understand the garments like ordered layers from the most extern to the most internal. For example we describe a person wearing a black jacket and a blue shirt like a structure of two layers, the first black and the second blue. In terms of regions, this can be seen as two black outer regions and one blue inner region.

When images are encoded using a spatial information on regions, indexing is well performed using approaches based on 2D-strings for pictorial description [10]. Shearer et al. [14] recently proposed a variation formulated in terms of inexact isomorphism detection between region graphs representing two images. Our method is inspired by that one. Thus, the clothing configuration models are formulated in terms of ideal position of regions and the corresponding region graph. Graph matching algorithms are time consuming, this is a drawback when retrieval represents browsing of a large database and, thus, compare a given graph with a number of candidate ones. For the sake of increasing the speed in the indexing process, we simplify the matching by defining each model in terms of a grid that divides the body of a person in four zones of interest in relation to the face that we have located before. So we distinct two zones vertically aligned

under the face, another on the left and another on the right. This zoning model can be seen in Fig. 4. Each model is described in terms of the presence of regions with certain features in each zone. For example, the model corresponding to three clothing layers like a jacked, a buttoned up shirt with a tee-shirt underneath, is stated in terms of regions like two regions of similar colour label covering outer zones, one region in the bottom central zone and, finally, a small region in the top central zone. Thus, given a segmented image, it is assigned to a class with a distance D_w that is computed in terms of the overlapping area of input regions to each zone. The strategy to assign images to a clothing class is driven by a decision tree. The process of retrieving images from the database similar to a given description consists in looking for the class model corresponding to the query and compute its similarity regarding to the region graphs representing the database images. In the example of Fig. 4, since there is a region that covers zones A, B and D, and another small region in zone C, the interpretation result in terms of clothing configuration is "outer garnet coloured clothe and inner black coloured clothe".

Fig. 4. Image interpretation in terms of the structure of regions.

5 Results

To test the algorithms a ground truth has been constructed with 1000 images of people acquired in a reception desk of a real environment. For security purposes, sometimes a person must be identified by making a query into the database in terms of a description. In the ground truth, we also store the reference description of each image made by a human operator. This allows to evaluate the performance of our algorithms. In order to asses how the system works, Fig. 5 illustrates the ten most similar images to the query "people who wears a clothing structure consisting in two layers". For each image we show the original one and the colour labeled. If we refine this query adding colour information "people who wears a clothing structure consisting in two layers and colour of the second layer = White" the system provide us only the images (a), (b) and (c).

6 Conclusions

In this paper we have proposed an algorithm for people retrieval from a database in terms of descriptors based on colour, texture and structural properties of

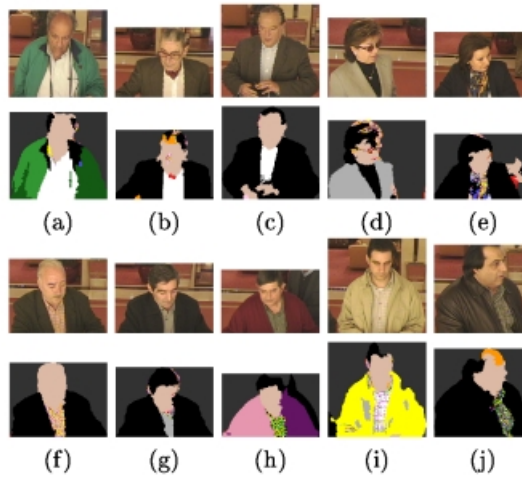

Fig. 5. Results of a simple query.

clothes that people is wearing. This algorithm is used in a real environment by surveillance staff to identify people that has been registered in a check-in desk. The key issues of the system are first, the use of a colour naming process that, considering the colour constancy problem, labels the image pixels with usual colour names of natural language. Secondly, the image segmentation in terms of colour and texture information obtaining an attributed graph structure. Region structure and colour labels are then mapped to people clothing descriptions that form database queries. The use of probability characterizing colour labels and a distance measure between region graphs allow the system to retrieve ranked images in terms of a confidence factor.

The system has been tested using a comprehensive set of images taken in a real environment. Although the results are just preliminary, the overall success of retrieving the desired image within the top ten given a query can be rated near a 70%. A further study on such ratio gives a 81% of success in the colour labeling step and a 75% in the description of structure in terms of clothing configuration. Errors in colour labeling often arise due to subjectiveness in the perception of colours. For example, regions labeled as dark blue can be perceived as black by the human operator that formulates the query. On the other hand, the right identification of clothing configuration is sensitive to occlusions or non frontal position of the people in the image.

From the point of view of the application, since it is not required that the system retrieves just one image after a query but a set of similar ones, the ratio of error is near acceptable levels. Issues to further study and that would contribute to significantly improve the retrieval are: first the inclusion of skin labeled regions in the structural matching. This would allow to identify in the same structure face hands and arms, allowing additional descriptors of clothes

such as short/long-sleeved. Secondly, hair is currently segmented using regions located on top of the face region, however the hair colour model should be better learned from examples. Finally, although region interpretation tolerates distortions, important variations in the ideal frontal position of people can make the system misclassify the image. The inclusion of symmetry cue with the skin colour identification would strengthen the recognition of clothing configurations.

Acknowledgments. This work has been partially supported by projects CICYT TIC2000-0382 and Inverama S.A.

References

1. Robert Benavente and Maria Vanrell. A color naming experiment.Technical Report 56,Computer Vision Center,2001.
2. B. Berlin and P. Kay. *Basic Color Terms: Their Universality and Evolution.* University of California Press, Berkeley 1969.
3. G.D. Finlayson, B. Schiele, and J.L. Crowley. Comprehensive colour image normalization. In *Proceedings of 5th ECCV'98* pages 475–490, 1998.
4. J. Forsyth, D.A. and Malik, M.M. Fleck, H. Greenspan, T.K. Leung, S. Belongie, C. Carson, and C. Bregler. Finding pictures of objects in large collections of images. *Object Representation in Computer Vision*, pages 335–360, 1996.
5. M. Gervautz and W. Purgathofer. A simple method for color quantization: Octree quantization. *Graphics Gems I*, pages 287–293, 1990.
6. C.L. Hardin and L. Maffi *Color categories in thought and language.* Cambridge University Press,Cambridge,1997.
7. K.Karu, A.K. Jain, and R.M. Bolle. Is there any texture in the image? In *Proceedings of 13th.* ICPR pages 770–774, August 1996. Viena, Austria.
8. P. Kay and C.K. McDaniel. The linguistic significance of the meaning of basic color terms. *Language* 3(54):610–646, 1978.
9. J.M. Lammens. A somewhat fuzzy color categorization model. In *Proceedings of ICCV-95*, 1995.
10. S.Y. Lee and F.J. Hsu. 2D C-string: A new spatial knowledge representation for image database systems. *Pattern Recognition*, 23(10):1077–1087, 1990.
11. P. Lipson, E. Grimson, and P. Sinha. Configuration based scene classification and image indexing. *Computer Vision and Pattern Recognition*, 1997.
12. X. Muñoz. *Image Segmentation Integrating Color*, Texture and Boundary Information. PhD thesis, Universitat de Girona, 2001.
13. J.Prosise. Wicked code. *MSJ* October 1997.
14. K. Shearer, H. Bunke, and S. Venkatesh. Video indexing and similarity retrieval by largest common subgraph detection using decision trees. *Pattern Recognition*, 34(5): 1077–1091, 2001.
15. M. Stricker and A. Dimai. Color indexing with weak spatial constraints. *Storage and Retrieval for Image and Video Databases*, 2670:29–40, 1996.
16. M. Vanrell, F. Lumbreras, A. Pujol, R. Baldrich, J. Lladós, and J.J. Villanueva. Colour normalisation based on background information.In *Proceedings of the 8th ICIP*, volume 1, pages 874–877, October 2001. Thessaloniki, Greece.