REGION-BASED SEGMENTATION AND TRACKING OF HUMAN FACES*

Verónica Vilaplana, Ferran Marqués, Philippe Salembier and Luis Garrido
Universitat Politècnica de Catalunya, Campus Nord – Mòdul D5
C/ Gran Capitá, Barcelona 08034, Spain
Tel: (343) 401 64 50, Fax: (343) 401 64 47
e-mail: {veronica, ferran, philippe, oster}@gps.tsc.upc.es

ABSTRACT

A new algorithm for face segmentation and tracking is presented. The face segmentation step relies on a fine segmentation of the image based on color homogeneity. In order to reduce the number of possible mergings, a merging order using chrominance homogeneity is applied. The sequence of mergings is represented as a binary tree. A set of regions (a node in the tree) is selected maximizing an estimation of the likelihood of being a face. The final face partition is obtained by a region merging process. The face tracking step uses motion information. The face partition in the previous partition is motion compensated (projected). The final face partition is obtained using the previous region merging process, driven by the projected face region.

1 INTRODUCTION

Automatic face detection is a very active area of image analysis, given its large number of applications [1]. The new functionalities that the coding standard MPEG4 [4] will allow have even increased this interest. Several techniques have been proposed to detect and track faces or features in faces [1, 9]. Nevertheless, MPEG4 applications rise a new necessity since the analysis algorithm should, not only detect the position of the face, but really segment it, obtaining its actual shape.

One of the basic approaches in face detection is that of view based eigenspaces [7, 2]. It assumes that the set of all possible face patterns is a low dimensional linear subspace within the high dimensional space of all possible image patterns. An image pattern is classified as a face if its distance from the face subspace is below a certain threshold. With this technique, the position of a face in an image can be detected. In Figure 1, an example of face detection is presented. The first image shows an original frame and the face detected using the technique proposed in [2]. The second image presents the type of face segmentation required for MPEG4 applications.

*This work has been partially supported by the ACTS-057 project (VIDAS) of the European Union and the grant 1997FI-00762 of the Direcció General de Recerca (Generalitat de Catalunya)

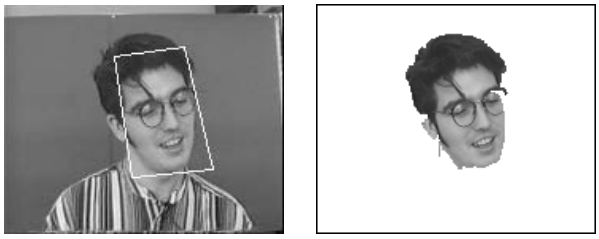

Figure 1: Example of face detection and segmentation

In [10], the face detection method presented in [7, 2] is extended to directly deal with regions. This way, the face is not only detected but correctly segmented. In this paper, the technique proposed in [10] is further improved by using autodual operators and Binary Partition Trees [8] instead of increasing (decreasing) operators and Min-trees (Max-trees). In addition, the new method is extended to the case of sequence analysis.

The organization of this paper is as follows. Section 2 describes the general merging strategy. The implementation of the face detection technique based on this strategy is presented in Section 3. The face tracking step is detailed in Section 4, and some results are shown in Section 5. Finally, Section 6 presents some conclusions.

2 GENERAL MERGING ALGORITHM

The general merging strategy proposed in [3] allows the implementation of segmentation algorithms and filters. The algorithm works on a region adjacency graph (RAG), a graph where each node represents a connected component of the image (regions or flat zones) and the links connect two neighboring nodes. A RAG represents a partition of the image. A merging algorithm on this graph is a technique that removes some of the links and merges the corresponding nodes. The merging is done in an iterative way. In order to completely specify a merging algorithm three notions have to be defined:

1. The *merging order*: it defines the order in which the links are processed. In the case of a segmentation

algorithm the merging order is defined by a similarity measure between two regions, and is closely related to the notion of objects.

2. The *merging criterion*: each time a link is processed, the merging criterion decides if the merging has actually to be done or not. In the case of a segmentation algorithm, the merging criterion always states that two regions have to be merged until a termination criterion is reached.

3. The *region model*: when two regions are merged, the model defines how to represent the union.

All the steps in the face detection process rely on this general merging algorithm. The main differences are the merging order and criteria used in each step.

3 FACE DETECTION

3.1 Initial partition

The proposed technique tries to avoid working at pixel level and, as first step, segments the image into homogeneous regions applying the merging algorithm described in Section 2. Color information is introduced in order to obtain more accurate contours. The region model used for each region is the median of each (y,u,v) component, computed recursively from the median of the two merged regions. The merging order is the relative squared error between region models, and the merging criterion (a termination criterion) is the final number of regions. In Figure 2 the initial partition, made of 70 regions, for the image Foreman#0, is presented.

Figure 2: Original image and initial partition

3.2 Face estimation

Since a face contains a set of regions with chrominance homogeneity, the goal of this step is to merge some regions from the initial partition following this criterion.

The merging strategy discussed in Section 2 is applied to the initial partition, using the same region model and merging order as before, but taking into account the (u,v) color components of the image. The merging is done until only one region remains, and the sequence of mergings is then analyzed.

3.2.1 Merging sequence analysis

The analysis is performed by associating the merging sequence to a binary partition tree [8], where the nodes represent the regions and the links connect two merging nodes. In Figure 3, the Binary Partition Tree describing the merging order based on chrominance information of the partition presented in Figure 2 is shown.

Then a similarity measure between each node and a face class is computed. This measure is an estimation of a distance between the regions associated to the node and the face class. To calculate the distance, an auxiliary image (i_x) containing a scaled version of the region information from the original image is created. This image has the size of the tightest rectangle bounding the original region. In the areas outside the region, the data base background is introduced.

For each node in the tree, the distance between the auxiliary image i_x and the face class is estimated and the node with minimum distance is selected. In the Binary Tree of Figure 3, the node in black is the node with minimum distance to the face class. The region associated to this selected node is presented in Figure 4(a). The region shown in Figure 4(b) is the following region proposed by the merging sequence; that is, the parent node of the selected node (node in gray in Figure 3). Note that the node with minimum distance corresponds to the best node in the tree, since it contains a large number of face components and no wrong component.

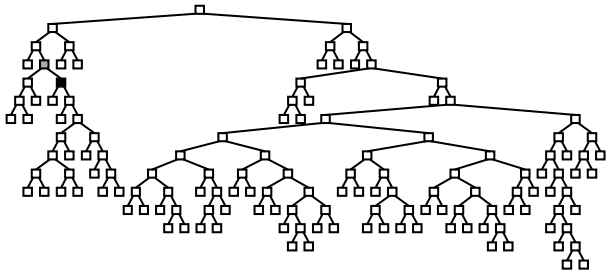

Figure 3: Binary Partition Tree

3.2.2 Distance definition

The distance, proposed in [7], is related to the likelihood of an image of being a face. The class membership of an image i_x is modeled as a unimodal gaussian density:

$$P(x/\Omega) = \frac{\exp[-\frac{1}{2}(x-\bar{x})^T \Sigma^{-1}(x-\bar{x})]}{(2\pi)^{\frac{N}{2}}|\Sigma|^{\frac{1}{2}}} \qquad (1)$$

where x is the N-dimensional vector made up from the lexicographic reading of image i_x, and the mean ($\bar{x}$) and the covariance matrix (Σ) are estimated using a training data set.

(a) (b)

Figure 4: Selected node and the following (rejected) candidate

The Mahalanobis distance is used as a sufficient statistic for characterizing this likelihood

$$d(x,\Omega) = (x-\bar{x})^T \Sigma^{-1} (x-\bar{x}) \tag{2}$$

A computationally tractable estimate of this distance, based on the M first principal components of the covariance matrix, is

$$\hat{d}(x,\Omega) = \sum_{i=1}^{M} \frac{y_i^2}{\lambda_i} + \frac{1}{\rho}\epsilon^2(x) \tag{3}$$

where y_i are the projections of x over the M principal components and λ_i are the M principal eigenvalues, with $M << N$. Moreover, ρ is the average of the $N-M$ remaining eigenvalues, and $\epsilon^2(x)$ is the residual reconstruction error:

$$\epsilon^2(x) = \sum_{i=M+1}^{N} y_i^2 = \|x-\bar{x}\|^2 - \sum_{i=1}^{M} y_i^2. \tag{4}$$

3.3 Face refinement

The initial estimate of the face may lack of some regions that form the face. The use of a merging process based on a chrominance criterion allows the simplification of the face segmentation process. However, it does not ensure that the optimum region (in the sense of the likelihood) is present as a node in the Binary Tree. Nevertheless, once the core components have been detected, a refinement step can be applied to completely extract the face information, without largely increase the computational load.

This refinement is based on the same merging algorithm as before, but the process is constrained to merge regions to the core components of the face. The merging order is given now by the estimated distance between the union of two neighboring regions and the face class.

4 FACE TRACKING

The face partition is not directly used for tracking purposes since its regions do not fulfill any fixed motion or spatial homogeneity. Instead, a second partition level is defined by re-segmenting the face partition [6]. The re-segmentation yields a second partition whose objective is to guarantee the color homogeneity of each region (*texture partition*) while preserving the contours present in the face partition.

The texture partition of the previous image is projected into the current frame to obtain the texture partition at the current image. The projection of the texture partition accommodates the previous partition to the information in the current image. The motion between the previous and current images is estimated and the previous texture partition is motion compensated. Compensated regions are used as markers giving an estimate of the region positions in the current image.

Given that motion compensated markers may be erroneous, they are accommodated to the boundaries of the current image [5]. Such boundaries are obtained from the so-called *fine partition*. This fine partition contains a large number of regions and is obtained using the technique reported in Section 2.1.

Compensated markers are fit into the fine partition to validate them. In a first step, compensated markers are reduced to the set of fine regions that are totally covered by them. Fine regions that are partially covered by more than one compensated marker are assigned to the uncertainty area. Note that this first step is purely geometrical and it yields the main connected components of each projected marker. Once the main components of every compensated region have been computed, neighboring regions from the fine partition can be added to them. This second step takes into account geometrical as well as color information and yields the core components of the face region. The final face partition is created by applying the refinement step on these core components.

5 RESULTS

In Figure 5, three different examples of automatic segmentation and tracking of faces are shown. Results have been obtained using the *Olivetti Research Laboratory* data base of face images (400 images with 40 different people and 10 views for person). The first three columns illustrate the segmentation step, while the two last columns present examples of the tracking capability. In the first column, the original image is presented, whereas the second column illustrates the result of the *Face estimation* step for each image. On turn, the third column contains the final result for each initial image. The fourth column contains a second frame of each sequence which has been segmented by means of the tracking approach. Tracking results are presented in the fith column.

Note that, although just some components are present in the first step of the analysis process, where the merging order is based on chrominance homogeneity (second column in Figure 5), the technique is able to correctly

Figure 5: Examples of face segmentation and tracking using the sequences *Foreman*, *Claire* and *Carphone*

segment the face present in the image (third column). In addition, the tracking step performs correctly even when the person turns his head, as in the case of the sequence *Foreman*.

6 CONCLUSIONS

The face segmentation and tracking technique presented in this paper successfully performs in a large set of sequences. Therefore, it can be used as a generic technique for applications that require the extraction of faces from sequences with human presence. Nevertheless, it has to be noticed that it relies on the quality of the initial segmentation. That is, since the final face partition is built up by means of a merging process starting from a fine partition, regions in this initial partition have to correctly represent the face boundaries.

The current work focus in two main aspects of the previous technique. First, the distance function is being improved so that a decision on the presence or absence of human faces can be taken. In addition, the technique is being speed up. The main computational bottle-neck, which is the calculation of the distance to the face class, is being analyzed.

References

[1] R. Chellappa, C. Wilson, and S. Sirohey. Human and machine recognition of faces: a survey. *Proceedings of the IEEE*, 83(5):705–740, May 1995.

[2] F. Davoine, C. Kervran, H. Li, P. Pérez, R. Forchheimer, and C. Labit. On automatic face and facial features detection in video sequences. In *International Workshop on Synthetic-Natural Hybrid coding and three dimensional imaging*, pages 196–199, Rhodes, Greece, 1997.

[3] L. Garrido, P. Salembier, and D. García. Extensive operators partition lattices for image sequence analysis. *EURASIP, Signal Processing*, 1998.

[4] ISO/IEC JTC1/SC29/WG11. MPEG-4 Proposal Package Description (PPD). July 1995.

[5] F. Marqués and J. Llach. Tracking of generic objects for video object generation. In *International Conference on Image Processing*. Accepted, 1998.

[6] F. Marqués and C. Molina. Object tracking for content-based functionalities. In *Proc. SPIE Visual Communication and Signal Processing-97 Conference*, pages 190–198, Feb 1997.

[7] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):696–710, July 1997.

[8] P. Salembier and L. Garrido. Binary partition tree as an efficient representation for filtering, segmentation and information retrieval. In *International Conference on Image Processing*. Accepted, 1998.

[9] K.K. Sung and T. Poggio. Example-based learning for view-based human face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):39–51, January 1998.

[10] V. Vilaplana and F. Marqués. Face segmentation using connected operators. In *Int. Symposium on Mathematical Morphology*. Accepted, 1998.