

# A Morphological Approach for Segmentation and Tracking of Human Faces

Ferran Marqués and Veronica Vilaplana  
Image Processing Group (TSC Department)  
Universitat Politècnica de Catalunya  
{ferran, veronica}@gps.tsc.upc.es

## Abstract

*A new technique for segmenting and tracking human faces in video sequences is presented. The technique relies on morphological tools such as connected operators to extract the connected component that more likely belongs to a face and partition projection to track this component through the sequence. A Binary Partition Tree (BPT) is used to implement the connected operator. The BPT is constructed based on chrominance criteria and its nodes are analyzed so that the selected node maximizes an estimation of the likelihood of being part of a face. The tracking is performed using a partition projection approach. Images are divided into face and non-face parts, which are tracked through the sequence. The technique has been successfully assessed using several test sequences from the MPEG-4 (raw format) and the MPEG-7 databases (MPEG-1 format).*

## 1. Introduction

Human face location is a key issue for many applications [1]. Currently, activities carried out in the MPEG-4 and MPEG-7 standardization processes require not only the face location, but its segmentation and tracking; that is, obtaining its actual shape in consecutive frames. A common approach to face detection is that of view based eigenspaces [2, 4]. In previous works [6, 7, 9], this approach has been extended to directly deal with regions. This has been done by defining connected operators that, using different implementation [3, 8], allow the merging of flat zones up to creating and detecting region(s) containing the face(s) in the scene.

Here, we present a new technique that improves the face segmentation step presented in previous related works. The connected operator is implemented using a Binary Partition Tree and improvements are associated to a new definition of the decision criterion and a better analysis of the tree, enabling the correct detection of several faces in the scene. In addition, faces are tracked using a method based on the concept of partition projection. The improvements in the segmentation step have been incorporated to the tracking algorithm,

yielding a more accurate and less complex tracking technique.

The structure of the paper is as follows. After this introduction, Section 2 presents the concept of connected operator and describes the way in which they can be applied in face segmentation. Section 3 is devoted to the description of the face-tracking algorithm. Finally, conclusions are given in Section 4.

## 2. Face segmentation using connected operators

**Definition:** An operator  $F(\cdot)$  acting on gray level functions is connected if, for any function  $f$ , the partition of flat zones of  $f$  is finer than the partition of flat zones of  $F(f)$ .

Thus, a connected operator acts by merging flat zones of the image (zones with constant gray value). This way, a connected operator preserves the original contours of the image, simplifies the image and does not create new contours in it.

In the proposed technique, connected operators are first used to avoid working at pixel level. Thus, as first step, the image is simplified and color homogeneous regions are created. Color information is used to achieve more accurate contours. The set of flat zones that define the simplified image creates a partition of the image space. The face component will be obtained as a union of regions belonging to this initial partition. Figure 1 shows an example of this type of initial partition.

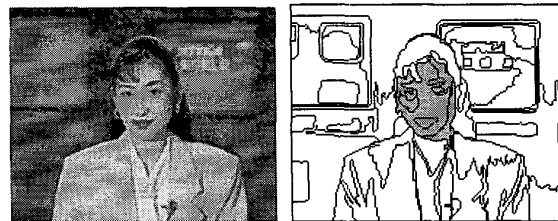


Figure 1. Original image (frame #0 of Akiyo) and example of initial partition where regions forming the face are highlighted

The connected operator used to simplify the original image uses color and contour information. A scalar space is created by computing an Euclidean weighted distance in the (y, u, v) color space, which is combined with contour information to actually decide which regions (flat zones) are to be merged. In it, smooth contours are prioritized since they usually conform better to real objects.

The selection of the union of initial regions that forms the face could be carried out by computing, for each region and possible union of regions, an estimation of their likelihood to belong to a face. This procedure is cumbersome and, as it will be described in Section 2.1, a simplified technique has been implemented.

### 2.1. Creation of the Binary Partition Tree

The basic concept relies on the general merging strategy for connected operators proposed in [3]. In it, a merging algorithm is defined by a merging order, a merging criterion and a region model, and the merging order and criterion are assumed to be independent. An efficient way to implement some of these connected operators uses the concept of Binary Partition Tree (BPT) [8]. Given a partition, where the size of the regions can be as small as a single pixel, a region merging order is defined. This merging order is based on a similarity measure between regions and regions are merged by pairs. The merging sequence that is obtained is represented by a BPT. In it, leaves are related to the regions in the original partition, nodes are associated to the various regions created during the merging process and, therefore, links connect two merging nodes.

Once the merging order has been established, the BPT is analyzed and the merging criterion is assessed at each node. The merging order proposes a set of region unions to the merging criterion. Then, the merging criterion takes the final decision whether two regions have to be merged or not. This strategy is useful in a large set of applications [8] and, specially, if a complex merging criterion is applied which may imply a large computational load.

In the current application, the measure used in the merging order is chrominance similarity. Since faces contain regions that are homogeneous in chrominance, the merging order is computed using the region similarity in the (u, v) color components. The merging order is established up to only one region remains and the merging sequence is then analyzed. Figure 2 presents the BPT of the partition in Figure 1.

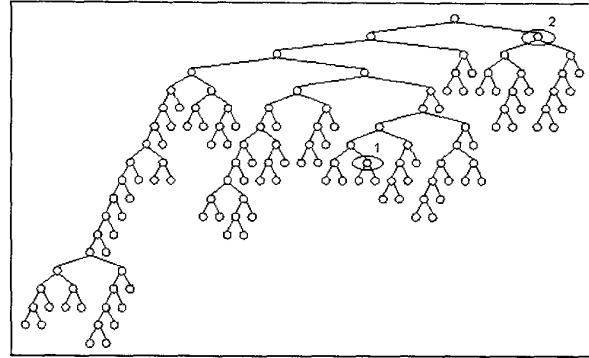


Figure 2. Binary Partition Tree obtained from the partition of Figure 1.

### 2.2. Analysis of the Binary Partition Tree

The distance proposed in [4] is used as merging criterion and, thus, computed at the nodes of the BPT. A face class ( $\Omega$ ) is created relying on a database of normalized face images (N-dimensional vectors). For each region, an N-dimensional vector representing the scaled version of the region  $x_w$  is constructed, in order to be able to detect faces of any size. The class membership of an image  $x_w$  is modeled as a unimodal gaussian density function:

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

where  $y_i$  are the projections of  $x_i$  over the M principal components and  $\lambda_i$  are the M principal eigenvalues, with  $M \ll N$ . Moreover,  $\rho$  is the average of the N-M remaining eigenvalues and  $\varepsilon^2(\cdot)$  is the residual reconstruction error:

$$\varepsilon^2(x_w / \Omega) = \sum_{i=M+1}^N y_i^2 = \|x_w - \bar{x}\|^2 - \sum_{i=1}^M y_i^2$$

To compute and evaluate all possible scaled versions of a given region is not feasible. Therefore, an auxiliary image is created and two possible scaled versions are computed: horizontal and vertical normalization with respect to the images in the database.

Since different scaling is necessary for each node in the BPT, the distance at a given node cannot be computed relying on the distances of its children nodes. To reduce the computational load, some nodes and even sub-trees are pruned based on size and color criteria. In the BPT of Figure 2, the sub-tree associated to node 2 has been pruned. These regions represent the blue screen on the top-right corner of the image that, very unlikely, may be part of a face. The analysis of the BPT shown in

Figure 2 leads to the selection of node 1. That is, node 1 is the region present in the BPT that maximizes the likelihood of being a part of a face. The region associated to this node is shown in Figure 3.a.

### 2.3 Face Refinement

The selected node 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 procedure. However, it does not ensure that the optimum region (in the sense of the likelihood) is present as a node in the BPT. Nevertheless, once the core components have been detected, a refinement step can be applied to completely extract the face information, without largely increasing the computational load.

This refinement is performed in two steps. First, geometrical information about the type of region that is to be obtained is introduced. The second step is based on the same merging algorithm used for obtaining the core face component. Nevertheless, here the BPT is not used and the analysis is constrained to merging the detected face component with its neighbor regions. The face component usually contains a large area of the face and the necessary scaling for analyzing the different possible mergings can be therefore fixed. This allows for speeding up the algorithm since, in this case, the distances to be computed for the different possible unions can be computed recursively. Figure 3.b shows the final face region yield by the refinement algorithm. A second example is presented in Figure 3.c and 3.d, which represent an original image of the Foreman sequence, and the face component finally detected, respectively.

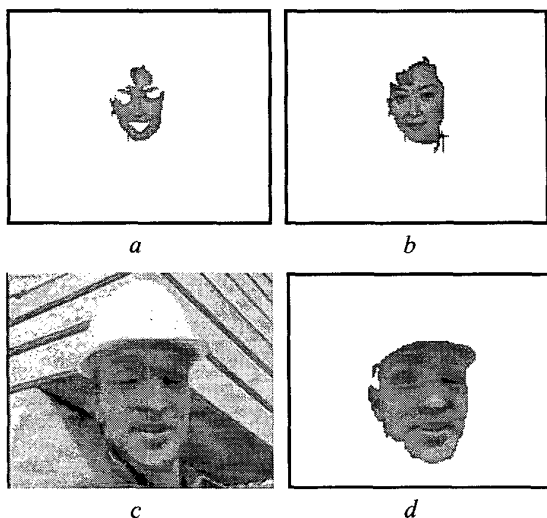


Figure 3. Examples of face detection.

## 3. Face Tracking Using Partition Projection

Once the face partition is obtained in the first image, it is used for tracking the face segment through the sequence. The face partition is not directly used for tracking purposes since its regions do not fulfill any fixed motion or spatial homogeneity. Instead, the partition defined by the leaves of the BPT is used since it guarantees the color homogeneity of each region (the so-called texture partition) [5]. The face in the current frame is obtained by tracking the texture partition using a partition projection approach [6]. An example of this approach is presented in Figure 4.

### 3.1 Partition Projection

The texture partition of the previous image is projected into the current frame to obtain the texture partition at the current image. The projection is done in two steps. First, the motion between the previous and current images is estimated and the previous texture partition is motion compensated. A backward block-matching algorithm is used in this step. Other motion estimation techniques have been tested, leading to similar results while increasing the computational time. Compensated regions are used as markers giving an estimate of the region positions in the current image. Second, motion compensated markers are fitted to a fine segmentation of the current image which contains its real boundaries. This partition is said to be fine because it contains a large number of regions, typically, about 1000 for QCIF images. The fine partition is obtained using very strict color homogeneity criteria that ensure the presence of all contours in the scene.

The fitting process involves geometrical, color and structural information. First, all regions from the fine partition that are totally covered by a motion compensated marker receive its label. This step provides with a first estimate of the position of previous regions in the current image. A second step that merges neighbor regions to those already labeled is then applied. In this step both geometrical as well as color information is used. The color information of the regions that are not totally covered is then analyzed. These regions are labeled if their distance to the compensate markers in the color space is small. The color distance that is used is that described in Section 2:

$$d(x, c) = \sqrt{\gamma(y_x - \bar{y}_c)^2 + \frac{(1-\gamma)}{2}((u_x - \bar{u}_c)^2 + (v_x - \bar{v}_c)^2)}$$

where  $x$  represents the region being analyzed and  $c$  the compensate marker.



Figure 4. Example of face tracking.

After the two previous steps, some parts of the current image may not have been assigned yet to any region from the previous partition. In this application, since the final goal is to obtain the face segment special mechanisms for assigning these areas are used. Regions from the fine partition can be assigned to the face segment if they are largely covered by compensated markers belonging to the face region in the previous image or if they totally surrounded by regions labeled as belonging to the face.

After this step, some uncertainty area may remain; that is, regions from the fine partition may have not been yet assigned. Relying on the classification carried out in the previous image, the current image is divided into three components after the fitting process: area that sure belongs to the face, area that sure does not belong to the face and uncertainty area.

### 3.2 Face refinement

A refinement process similar to that of Section 2.3 is used to obtain the final face. The union of each region forming the uncertainty area with the already detected face area is analyzed. A region is added when the union increases the likelihood of belonging to a face. This approach improves the generic tracking technique proposed in [5] since it includes semantic information in the tracking process.

In order to speed up the algorithm, the scaling of the different region unions for computing the likelihood to belonging to a face is carried out only once. This is done by taking as size for the scaling the union of the face and uncertainty area. Figure 5 presents the results of tracking the face present in the *Foreman* sequence using this technique.

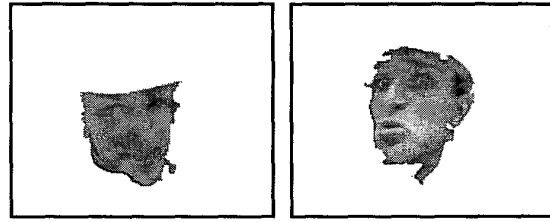


Figure 5. Face tracking in *Foreman* (#40 and #148).

## 4. Conclusions

The proposed face segmentation and tracking method has been successfully assessed on a large number of test sequences from the MPEG-4 database (raw format) as well as from the MPEG-7 database (MPEG-1 format).

In these experiments, the face class ( $\Omega$ ) has been formed using the Olivetti database, which only contains frontal images. Hence the relevance of being able to track the segmented faces in the case of non-frontal images. This is due to the fact of using a projection that provides, for each frame, a good first estimate of the face position.

## References

- [1] R. Chellappa, C. L. Wilson, S. Sirohey, "Human and machine recognition of faces: a survey", *Proceedings of the IEEE*, Volume 83, No. 5, pp. 705-740, May 1995.
- [2] F. Davoine, et al., "On automatic face and facial features detection in video sequences, *International Workshop on Synthetic-Natural Hybrid coding and 3-D imaging*, pp. 196-199, Rhodes, Greece, 1997.
- [3] L. Garrido, P. Salembier, D. Garcia, "Extensive operators in partition lattices for image sequence analysis", *EURASIP Signal Processing*, Vol. 66, No. 2, pp. 157-180, April 1998.
- [4] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation", *IEEE Trans. PAMI*, Vol. 19, No. 7, pp. 696-710, July 1997.
- [5] F. Marqués, J. Llach, "Tracking of generic objects for Video Object generation", In *ICIP 98*, pp. 628-632, Chicago, 1998.
- [6] F. Marqués, V. Vilaplana, A. Buxes, "Human face segmentation and tracking using connected components and partition projection", *Proceedings of the ICIP 99*, paper 27PO303, Kobe, Japan, 1999.
- [7] B. Menser, M. Brünig, "Segmentation of human faces in color images using connected operators", *Proceedings of the ICIP 98*, paper 26PO36, Kobe, 1999.
- [8] P. Salembier and L. Garrido, "Binary partition tree as an efficient representation for filtering, segmentation and information retrieval", In *ICIP-98*, Chicago, 1998.
- [9] V. Vilaplana, F. Marqués, "Face segmentation using connected operators", In *Mathematical Morphology and its Applications to Image and Signal Processing*, pp. 207-214, Amsterdam, 1998.