Object Segmentation and Ontologies for MPEG-2 Video Indexing and Retrieval*

Vasileios Mezaris[1,2] and Michael G. Strintzis[1,2]

[1] Information Processing Laboratory, Electrical and Computer Engineering
Department, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece
[2] Informatics and Telematics Institute (ITI)/ Centre for Research and Technology
Hellas (CERTH), Thessaloniki 57001, Greece

Abstract. A novel approach to object-based video indexing and retrieval is presented, employing an object segmentation algorithm for the real-time, unsupervised segmentation of compressed image sequences and simple ontologies for retrieval. The segmentation algorithm uses motion information directly extracted from the MPEG-2 compressed stream to create meaningful foreground spatiotemporal objects, while background segmentation is additionally performed using color information. For the resulting objects, MPEG-7 compliant low-level indexing descriptors are extracted and are automatically mapped to appropriate intermediate-level descriptors forming a simple vocabulary termed *object ontology*. This, combined with a relevance feedback mechanism, allows the qualitative definition of the high-level concepts the user queries for (*semantic objects*, each represented by a *keyword*) and the retrieval of relevant video segments. Experimental results demonstrate the effectiveness of the proposed approach.

1 Introduction

Sophisticated query and retrieval from video databases is an important part of many emerging multimedia applications. Retrieval schemes of such applications employ descriptors ranging from low-level features to higher-level semantic concepts. In all cases, preprocessing of video data is necessary as the basis on which indices are extracted. The preprocessing is of *coarse granularity* if it involves processing of video frames as a whole, whereas it is of *fine granularity* if it involves detection of objects within a video frame [1]. In this work, a fine granularity approach is adopted.

To this end, several approaches have been proposed in the literature for video segmentation. Most of these operate in the uncompressed pixel domain [2], which enables object boundary estimation with pixel accuracy but requires that the sequence be fully decoded before segmentation. As a result, the usefulness of such

* This work was supported by the EU project SCHEMA "Network of Excellence in Content-Based Semantic Scene Analysis and Information Retrieval" (IST-2001-32795).

P. Enser et al. (Eds.): CIVR 2004, LNCS 3115, pp. 573–581, 2004.

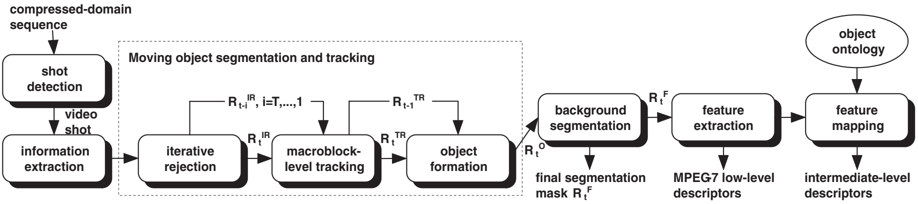

compressed-domain sequence
shot detection
video shot
information extraction
Moving object segmentation and tracking
iterative rejection
R_{t-i}^{IR}, i=T,...,1
R_t^{IR}
macroblock-level tracking
R_{t-1}^{TR}
R_t^{TR}
object formation
R_t^O
background segmentation
final segmentation mask R_t^F
R_t^F
feature extraction
MPEG-7 low-level descriptors
object ontology
feature mapping
intermediate-level descriptors

Fig. 1. Overview of the compressed-domain spatiotemporal segmentation algorithm and the feature extraction procedure.

approaches is usually restricted to non-real-time applications. To counter these drawbacks, compressed domain methods have been proposed for spatiotemporal segmentation and indexing [3,4]. Although significantly faster than most pixel-domain algorithms, some of them cannot operate in real-time [5].

To allow efficient indexing of large video databases, an algorithm for the real-time, unsupervised spatiotemporal segmentation of MPEG-2 video sequences is proposed. Only I- and P-frames are examined, since they contain all information that is necessary for the proposed algorithm; this is also the case for most other compressed-domain algorithms. Both foreground and background spatiotemporal objects are identified for each shot.

In the proposed indexing and retrieval scheme, instead of adopting the query-by-example strategy, the aforementioned segmentation algorithm is combined with simple *ontologies* [6] and a *relevance feedback* mechanism. This scheme (Fig. 1) allows for MPEG-7 compliant low-level indexing features to be extracted for the spatiotemporal objects and subsequently be associated with higher-level descriptors that humans are more familiar with; these are used to restrict the search to a set of potentially relevant spatiotemporal objects. Final query results are produced after one or more rounds of relevance feedback, similarly to [6] where this method was applied to still images.

The paper is organized as follows: in section 2 video object segmentation and tracking methods are developed. The indexing and retrieval scheme is discussed in section 3. Section 4 contains experimental results and finally, conclusions are drawn in section 5.

2 Video Object Segmentation and Tracking

2.1 Compressed-Domain Information Extraction

The information used by the proposed segmentation algorithm is extracted from MPEG-2 sequences during the decoding process. Specifically, motion vectors are extracted from the P-frames and are used for foreground/background segmentation and for the subsequent identification of different foreground objects. In order to derive motion information for the I-frames, averaging of the motion vectors of the P-frames that are temporally adjacent to the given I-frame is performed. Additionally, color information is used in order to further segment the background to its constituent objects. The employed color information is restricted to the DC coefficients of the I-frame macroblocks, corresponding to the Y, Cb and Cr components of the MPEG color space.

2.2 Iterative Macroblock Rejection

Iterative rejection for the estimation of the eight parameters of the bilinear motion model was proposed in [7], where it was used for the retrieval of video clips based on their global motion characteristics. This method is based on iteratively estimating the parameters of the global-motion model using least-square estimation and rejecting those blocks whose motion vectors result in larger than average estimation errors, the underlying assumption being that the background is significantly larger than the area covered by the moving objects.

In this work, iterative rejection based on the bilinear motion model is used to generate the mask R_t^{IR}, indicating which macroblocks have been rejected at time t (or activated, from the segmentation objective's point of view). This is the first step of foreground / background segmentation. Rejected (activated) macroblocks are treated as potentially belonging to foreground objects.

2.3 Macroblock-Level Tracking

In order to examine the temporal consistency of the output of iterative rejection, activated macroblocks are temporally tracked using the compressed-domain motion vectors. The temporal tracking is based upon the work presented in [8], where objects are manually marked by selecting their constituent macroblocks and are subsequently tracked. However, in contrast to the method in [8], the proposed method requires no human intervention for the selection of the macroblocks to be tracked. A shortcoming of the former method, the need for block matching in order to extract motion features for the I-frames, is avoided in the present work by averaging the motion vectors of the P-frames that are temporally adjacent to the given I-frame.

More specifically, let $\tau(.)$ be the tracking operator realizing the tracking process of [8], whose input is a macroblock at time t and its output is the corresponding macroblock or macroblocks at time $t+1$. This correspondence is established by estimating the overlapping of the examined macroblock with its spatially adjacent ones, determined using the displacement indicated by its motion vector. Then, the operator $\mathcal{T}(.)$ is defined as having a mask (such as R_t^{IR}) as input, applying the $\tau(.)$ operator to the set of all foreground macroblocks of that mask, and outputting the corresponding mask at time $t+1$.

Let R_t^{TR} denote the output foreground/background mask derived via macroblock level tracking, using masks R_{t-i}^{IR}, $i = T, \ldots, 0$. The derivation of mask R_t^{TR}, using the operator $\mathcal{T}(.)$ to evaluate and enforce the temporal consistency of the output of iterative rejection over T frames, can be expressed as:

$$R_{t-T}^{temp} = R_{t-T}^{IR}$$

$$\text{for } i = T, \ldots, 1, \;\; R^{temp}_{t-i+1} = \mathcal{T}(R^{temp}_{t-i}) \cap R^{IR}_{t-i+1}$$
$$R^{TR}_t = R^{temp}_t$$

where $\cap$ denotes the intersection of foreground macroblocks and R^{temp}_{t-i}, $i = T, \ldots, 0$ is a set of temporary foreground/background segmentation masks.

2.4 Spatiotemporal Object Formation

After the rejection of falsely activated macroblocks, as described in the previous subsection, the remaining macroblocks are clustered to connected foreground regions and are subsequently assigned to foreground spatiotemporal objects. Clustering to connected regions s^t_k, $k = 1, \ldots, \kappa^t$ using a four-connectivity component labelling algorithm results in the creation of mask R^I_t.

To determine whether a given spatial region of R^I_t belongs to one or more preexisting spatiotemporal objects or to a newly appearing one, and to eventually create the object mask R^O_t, motion projection is performed by applying the tracking operator $\tau(.)$ to the macroblocks of each spatiotemporal object of mask R^O_{t-1}. Thus, every connected region s^t_k of mask R^I_t can be assigned to one of the following three categories:

- Cat. 1. A number of macroblocks $M_{k,q}$, $M_{k,q} \geq S_{k,q}$, of s^t_k have been assigned to spatiotemporal object o_q in mask $\mathcal{T}(R^O_{t-1})$, and no macroblock of s^t_k has been assigned to a spatiotemporal object o_m, $m \neq q$.
- Cat. 2. A number of macroblocks $M_{k,q}$, $M_{k,q} \geq S_{k,q}$, of s^t_k have been assigned to spatiotemporal object o_q in mask $\mathcal{T}(R^O_{t-1})$, and one or more macroblocks of s^t_k have been assigned to different spatiotemporal objects, namely o_m, $m = 1, \ldots, M$.
- Cat. 3. There is no spatiotemporal object o_q in mask $\mathcal{T}(R^O_{t-1})$ having $M_{k,q}$ macroblocks of s^t_k, $M_{k,q} \geq S_{k,q}$, assigned to it.

The parameter $S_{k,q}$ in the definition of the above categories is estimated for every pair of a spatial region s^t_k of mask R^I_t and the projection of a spatiotemporal object o_q in mask $\mathcal{T}(R^O_{t-1})$. Let M^s_k, M^o_q denote their sizes in macroblocks, then parameter $S_{k,q}$ is calculated as $S_{k,q} = a \cdot \frac{M^s_k + M^o_q}{2}$.

The spatial regions s^t_k classified in the third category can not be associated with an existing spatiotemporal object; therefore, each region of this category forms a new spatiotemporal object. Similarly, the regions classified in the first category can only be associated with a single spatiotemporal object o_q; however, more than one spatial regions may be associated with the same spatiotemporal object. In this case, the larger spatial region becomes part of o_q, while the rest are discarded. As for the regions s^t_k classified in the second category, the initial correspondence of specific macroblocks belonging to s^t_k with spatiotemporal objects is employed so as to estimate the parameters of the bilinear motion model for each of the competing objects. Subsequently, each macroblock is assigned to the object for which the motion estimation error in minimized and possible merging of adjacent objects is examined using the parameters of their common motion model.

2.5 Background Segmentation

After foreground spatiotemporal objects have been extracted, background segmentation is performed based on classifying the remaining macroblocks to one of a number of background spatiotemporal objects. Background segmentation begins by applying the *maximin* algorithm to the color DC coefficients of the first I-frame. Then, in I-frames, background macroblocks are clustered to background objects using the K-means algorithm, where K is the number of objects estimated by the maximin algorithm. In P-frames, the absence of color information is dealt with by using the macroblock motion vectors and a previous final mask R^F_{t-1}. Temporal tracking is then performed as discussed in previous sections.

3 Object-Based Indexing and Retrieval

3.1 Overview

The proposed segmentation algorithm is suitable for introducing object-based functionalities to video indexing and retrieval applications, due to the formation of both foreground and background spatiotemporal objects, for which object-based descriptors in the context of the MPEG-7 Visual standard [9] can be extracted.

With the exception of a few MPEG-7 descriptors, most standardized descriptors are low-level arithmetic ones. When examining the specific application of object-based video indexing, however, it is possible to translate certain low-level arithmetic values to intermediate-level descriptors qualitatively describing the object attributes; the latter are more suitable for manipulation by humans. Extending the approach in [6], these intermediate-level descriptors form a simple vocabulary, the *object ontology*. Ontologies are tools for structuring knowledge, defined as the specification of a representational vocabulary for a shared domain of discourse which may include definitions of classes, relations, functions and other objects. In the proposed scheme, ontologies are used to facilitate the mapping of low-level descriptor values to higher-level semantics.

Under the proposed scheme, a query is initiated by the user qualitatively describing the semantic objects and their relations in the desired shot. By comparing the user-supplied qualitative description with the one automatically estimated for each spatiotemporal object, clearly irrelevant ones can be discarded; the remaining, potentially relevant ones are presented to the user at random order. The user then evaluates a subset of them, marking relevant ones simply by checking the appropriate “relevant” box. By submitting this relevance feedback, a support vector machine is trained and subsequently ranks according to relevance all potentially relevant spatiotemporal objects, using their low-level descriptor values; the shots containing these objects are then presented to the user, ordered by rank. This relevance feedback process can then be repeated, to further enhance the output of the query.

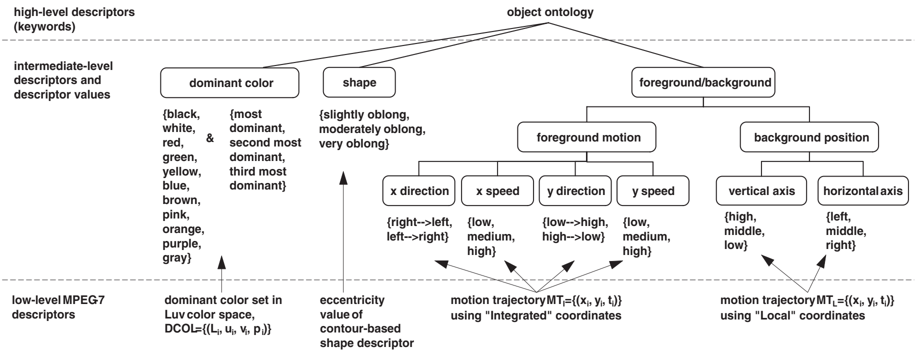

high-level descriptors (keywords)
object ontology
intermediate-level descriptors and descriptor values
dominant color
shape
foreground/background
{black, white, red, green, yellow, blue, brown, pink, orange, purple, gray}
&
{most dominant, second most dominant, third most dominant}
{slightly oblong, moderately oblong, very oblong}
foreground motion
background position
x direction
x speed
y direction
y speed
vertical axis
horizontal axis
{right-->left, left-->right}
{low, medium, high}
{low-->high, high-->low}
{low, medium, high}
{high, middle, low}
{left, middle, right}
low-level MPEG-7 descriptors
dominant color set in Luv color space, DCOL={(L_i, u_i, v_i, p_i)}
eccentricity value of contour-based shape descriptor
motion trajectory MT_I={(x_i, y_i, t_i)} using "Integrated" coordinates
motion trajectory MT_L={(x_i, y_i, t_i)} using "Local" coordinates

Fig. 2. Object ontology. The correspondence between low-level MPEG-7 descriptors and intermediate-level descriptors is shown.

3.2 MPEG-7 Descriptors

As soon as a sequence of segmentation masks is produced for each video shot, a set of descriptor values useful in querying the database are calculated for each spatiotemporal object. Standardized MPEG-7 descriptors are used, to allow for flexibility in exchanging indexing information with other MPEG-7 compliant applications. The different MPEG-7 descriptors used in this work are: Motion Activity, Dominant Color, GoF/GoP Color, Contour Shape, Motion Trajectory using “Local” coordinates, and Motion Trajectory using “Integrated” coordinates.

3.3 Object Ontology

In this work, ontologies [6] are employed to allow the user to query a video collection using semantically meaningful concepts (semantic objects), without the need for performing manual annotation of visual information. A simple *object ontology* is used to enable the user to describe semantic objects, like “tiger”, using a vocabulary of intermediate-level descriptor values. These are automatically mapped to the low-level descriptor values calculated for each spatiotemporal object in the database, thus allowing the association of keywords representing semantic objects (e.g. the “tiger” keyword) and potentially relevant spatiotemporal objects. The simplicity of the employed object ontology permits its applicability to generic video collections without requiring the correspondence between spatiotemporal objects and relevant descriptors to be defined manually. This object ontology can be expanded so as to include additional descriptors corresponding either to low-level properties (e.g. texture) or to higher-level semantics which, in domain-specific applications, could be inferred either from the visual information itself or from associated information (e.g. subtitles).

The object ontology is presented in Fig. 2, where the possible intermediate-level descriptors and descriptor values are shown. Each intermediate level descriptor value is mapped to an appropriate range of values of the corresponding low-level, arithmetic descriptor. With the exception of color (e.g. "black") and direction (e.g. "low→high") descriptor values, the value ranges for every low-level descriptor are chosen so that the resulting intervals are equally populated. This is pursued so as to prevent an intermediate-level descriptor value from being associated with a plurality of spatiotemporal objects in the database, since this would render it useless in restricting a query to the potentially most relevant ones. Overlapping, up to a point, of adjacent value ranges, is used to introduce a degree of fuzziness to the descriptor values; for example, both "slightly oblong" and "moderately oblong" values may be used to describe a single object.

Regarding color, a correspondence between the 11 basic colors [10] used as color descriptor values and the values of the HSV color space is heuristically defined. More accurate correspondences based on psychovisual findings are possible; this is however beyond the scope of this work. Regarding the direction of motion, the mapping between values for the descriptors "x direction", "y direction" and the MPEG-7 *Motion Trajectory* descriptor is based on the sign of the cumulative displacement of the foreground spatiotemporal objects.

3.4 Relevance Feedback

After using the intermediate-level descriptors to narrow down the search to a set of potentially relevant spatiotemporal objects, relevance feedback is employed to produce a qualitative evaluation of the degree of relevance of each spatiotemporal object. The employed mechanism is based on a method proposed in [11], where it is used for image retrieval using global image properties under the query-by-example scheme. This method combines support vector machines (SVM) with a constrained similarity measure (CSM). Under the proposed scheme, the SVMs classify spatiotemporal objects to relevant or non-relevant using their low-level descriptor vectors, while the CSM proposed in [11] is modified to subsequently assign to each spatiotemporal object classified as relevant the minimum of the Euclidean distances between it and all positive training samples.

4 Experimental Results

The proposed algorithms were tested on known test sequences, as well as a collection of video shots. Results of the real-time compressed-domain segmentation algorithm are presented for the "Table-tennis" sequence (Fig. 3). The proposed segmentation approach imposes little additional computational burden to the MPEG decoder: excluding any processes of it, the proposed algorithm requires on the average 5.02 msec per processed CIF-format I/P-frame on an 800Mhz Pentium III.

For each video object created by applying the segmentation algorithm to a collection of video shots, MPEG-7 low-level descriptors were calculated and the mapping between them and the intermediate-level descriptors defined by the object ontology was performed. Subsequently, the object ontology was used

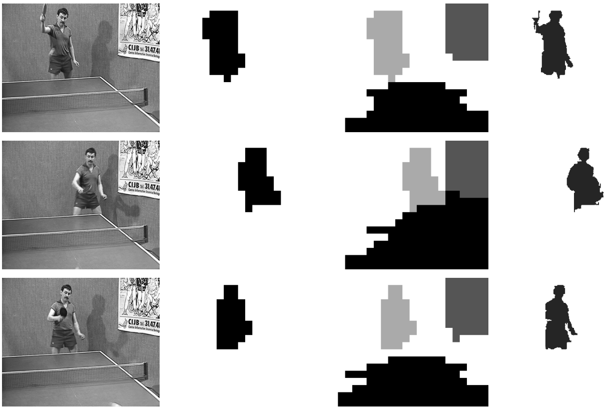
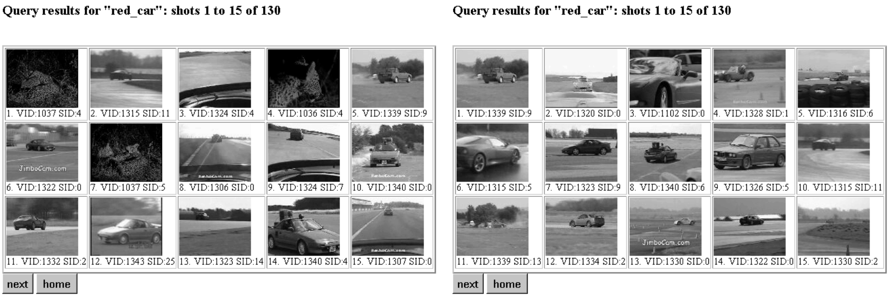


Fig. 3. Results of moving-object detection, final mask after background segmentation, and moving objects after pixel-domain boundary refinement using a Bayes classifier to reclassify specific pixels in a fashion similar to that of [12], for “Table-tennis”.



Query results for "red_car": shots 1 to 15 of 130

1. VID:1037 SID:4	2. VID:1315 SID:11	3. VID:1324 SID:4	4. VID:1036 SID:4	5. VID:1339 SID:9
6. VID:1322 SID:0	7. VID:1037 SID:5	8. VID:1306 SID:0	9. VID:1324 SID:7	10. VID:1340 SID:0
11. VID:1332 SID:2	12. VID:1343 SID:25	13. VID:1323 SID:14	14. VID:1340 SID:4	15. VID:1307 SID:0

(a)

Query results for "red_car": shots 1 to 15 of 130

1. VID:1339 SID:9	2. VID:1320 SID:0	3. VID:1102 SID:0	4. VID:1328 SID:1	5. VID:1316 SID:6
6. VID:1315 SID:5	7. VID:1323 SID:9	8. VID:1340 SID:6	9. VID:1326 SID:5	10. VID:1315 SID:11
11. VID:1339 SID:13	12. VID:1334 SID:2	13. VID:1330 SID:0	14. VID:1322 SID:0	15. VID:1330 SID:2

(b)

Fig. 4. Results for a “red car” query: (a) shots containing potentially relevant objects, identified using the intermediate-level descriptors, (b) results after one round of relevance feedback.

to define, using the available intermediate-level descriptors, semantic objects. Querying using these definitions resulted in initial results produced by excluding the majority of spatiotemporal objects in the database. Finally, one or more pages of potentially relevant spatiotemporal objects were presented to the user for manual evaluation and training of the SVM-based feedback mechanism. Results of this procedure for a “red car” query are presented in Fig. 4.

5 Conclusions

An algorithm for compressed video segmentation was presented in this paper, along with an indexing and retrieval scheme. Due to its real-time, unsupervised operation, the proposed algorithm is very suitable for content-based multimedia applications requiring the manipulation of large volumes of visual data. The proposed video indexing and retrieval scheme, based on the combination of the proposed segmentation algorithm with ontologies and relevance feedback, enabled the formulation of descriptive queries and allowed efficient retrieval of video segments.

References

1. Al-Khatib, W., Day, Y., Ghafoor, A., Berra, P.: Semantic modeling and knowledge representation in multimedia databases. IEEE Trans. on Knowledge and Data Engineering **11** (1999) 64–80
2. O'Connor, N., Sav, S., Adamek, T., Mezaris, V., Kompatsiaris, I., Lui, T., Izquierdo, E., Bennstrom, C., Casas, J.: Region and Object Segmentation Algorithms in the Qimera Segmentation Platform. In: Proc. Third Int. Workshop on Content-Based Multimedia Indexing (CBMI03). (2003)
3. Meng, J., Chang, S.F.: Tools for Compressed-Domain Video Indexing and Editing. In: Proc. SPIE Conf. on Storage and Retrieval for Still Image and Video Databases IV, Ishwar K. Sethi; Ramesh C. Jain; Eds. Volume 2670. (1996) 180–191
4. Sahouria, E., Zakhor, A.: Motion Indexing of Video. In: Proc. IEEE Int. Conf. on Image Processing (ICIP97), Santa Barbara, CA (1997)
5. Babu, R., Ramakrishnan, K.: Compressed domain motion segmentation for video object extraction. In: Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing. Volume 4. (2002) 3788–3791
6. Mezaris, V., Kompatsiaris, I., Strintzis, M.G.: An Ontology Approach to Object-based Image Retrieval. In: Proc. IEEE Int. Conf. on Image Processing (ICIP03), Barcelona, Spain (2003)
7. Yu, T., Zhang, Y.: Retrieval of video clips using global motion information. Electronics Letters **37** (2001) 893–895
8. Favalli, L., Mecocci, A., Moschetti, F.: Object tracking for retrieval applications in MPEG-2. IEEE Trans. on Circuits and Systems for Video Technology **10** (2000) 427–432
9. Sikora, T.: The MPEG-7 Visual standard for content description - an overview. IEEE Trans. on Circuits and Systems for Video Technology, special issue on MPEG-7 **11** (2001) 696–702
10. Berlin, B., Kay, P.: Basic color terms: their universality and evolution. Berkeley, University of California (1969)
11. Guo, G.D., Jain, A., Ma, W.Y., Zhang, H.J.: Learning similarity measure for natural image retrieval with relevance feedback. IEEE Trans. on Neural Networks **13** (2002) 811–820
12. Mezaris, V., Kompatsiaris, I., Strintzis, M.: A framework for the efficient segmentation of large-format color images. In: Proc. IEEE Int. Conf. on Image Processing (ICIP02). Volume 1. (2002) 761–764