A NOISE ROBUST METHOD FOR SEGMENTATION OF MOVING OBJECTS IN VIDEO SEQUENCES

Roland Mech and Michael Wollborn

Institut für Theoretische Nachrichtentechnik und Informationsverarbeitung
Universität Hannover, Appelstraße 9A, D–30167 Hannover, FRG
Phone: +49–511–762–5308, +49–511–762–5312
Email: mech@tnt.uni–hannover.de, wollborn@tnt.uni–hannover.de

ABSTRACT

An algorithm for automatic, noise robust segmentation of moving objects in image sequences is presented. Such algorithms are required for object–based coding techniques like the upcoming ISO/MPEG–4 standard. In a first step, a mask of changed image areas is estimated by a local thresholding relaxation technique. Then, areas of uncovered background are removed from this mask, taking into account an estimated displacement vector field. The resulting object mask is finally improved by applying a greylevel edge adaptation and an object mask memory. The algorithm is compared to a global thresholding technique which is known from the literature. Experimental results show the improvement of the estimated object masks.

1. INTRODUCTION

For the coding of image sequences at very low bit rates, standardized blockbased hybrid coding techniques [4][8] are transmitting texture and motion information in order to minimize the required transmission bit rate at a specific image quality. However, within the area of object–based analysis–synthesis coding [5][7][11] [12] it has been shown that also shape information with respect to arbitrarily shaped moving objects in the captured scene can be used to improve the coding efficiency. Moreover, the upcoming ISO/MPEG–4 standard [9][13] considers such shape information not only for the coding efficiency but also to allow for the so–called content based functionalities.

Current methods for segmentation and shape estimation of moving objects in video sequences are based on thresholding the luminance difference image of two successive frames by using a global threshold. Such a segmentation technique is described in [6] and shall be used as a reference. However, these algorithms are firstly not accurate enough, i.e. the estimated boundaries of moving objects deviate from the real object boundaries. Secondly, due to the global thresholding, these algorithms are very sensitive against noise, leading also to deviations in the estimated boundaries or even to false detection of objects.

In this paper, a more accurate, noise robust segmentation and shape estimation method is presented. Therefore, the segmentation method described in [6] is extended by a local adaptive thresholding technique for the detection of changed image areas which was proposed in [1][2]. Furthermore, in order to get more time coherent segmentation results, a memory for the change detection mask is applied. Finally, the boundaries of the resulting object regions are adapted to grey level edges of the current luminance image in order to improve the accuracy of the estimated object shape. So, assuming non–moving background, a segmentation of each frame of an image sequence into non–overlapping moving objects (denoted as foreground) and static and uncovered background regions (denoted as background) is performed.

2. THE SEGMENTATION ALGORITHM

The proposed algorithm can be subdivided into the following two steps: in the first step, a change detection mask (CDM) is calculated. In this mask, every pel where the luminance of the image has changed due to a moving object is marked. In the second step, the uncovered background areas are eliminated from the CDM, resulting in an object mask (OM) where every pel is marked which belongs to a moving object. These two steps are described more detailed in the following. Fig. 1 shows the segmentation algorithm as a block diagram.

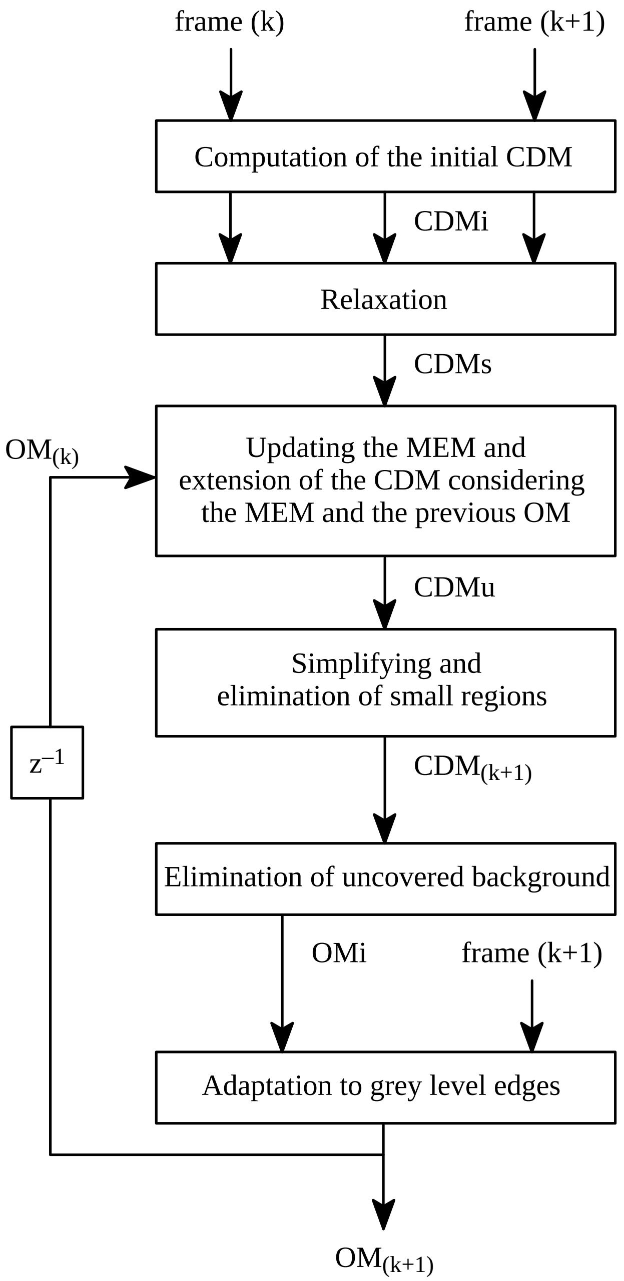


Figure 1: Block diagram of the segmentation algorithm.

2.1 Estimation of a change detection mask

First, an initial change detection mask (CDMi) between two successive frames is calculated, using a global threshold. Then, boundaries of changed image areas are smoothed by a local adaptive relaxation technique described in [1][2], resulting in a mask CDMs. Therefore, the CDMi is processed iteratively. In each iteration step, a decision is made for every border pel k in the CDMi if it belongs to the changed or to the unchanged area. The following local decision rule from [1] is applied, using the squared luminance difference d_i^2 at the location of pel i:

$$d_i^2 \underset{u}{\overset{c}{\gtrless}} 2\frac{\sigma_c^2\,\sigma^2}{\sigma_c^2-\sigma^2} \cdot \left(\ln\frac{\sigma_c}{\sigma} + (v_B(c) - v_B(u))\;B + (v_C(c) - v_C(u))\;C\right) \qquad (1)$$

The rule should read as follows: if d_i^2 exceeds the threshold term on the right hand side of (1), the pel is set to changed ($CDMs_{(k)}(i) := 1$), otherwise it is set to unchanged ($CDMs_{(k)}(i) := 0$). In the threshold term, σ^2 is equal to twice the variance of the assumed Gaussian camera noise distribution. σ_c^2 is the variance of luminance differences within object regions. The terms $v_B(q_k)$ and $v_C(q_k)$, $q_k \in \{u, c\}$, are a measure for the inhomogeneity of the neighbourhood of pel k, which is separated into horizontal or vertical neighbours (potential B) and diagonal neighbours (potential C). So, the term $v_B(q_k)$ denotes the number of horizontal and vertical neighbours of pel k with the opposite label to q_k. In the same way the term $v_C(q_k)$ denotes the number of diagonal neighbours of pel k with the opposite label to q_k.

The algorithm adapts frame–wise automatically to the variances σ_c^2 and σ^2. While in [1] these variances are both measured considering the CDMi, here the object mask of the previous frame is used for calculating the variance σ_c^2 and the change detection mask of the previous frame is used for calculating the variance σ^2. In order to get more stable values for these variances, the currently measured values are averaged with the last three measured values. Thus, the algorithm gets more robust in case of noise.

In order to finally get temporally stable object regions, the CDMs is updated with the previous OM, i.e. in the CDMs additionally all pels are set to changed which belong to the object mask of the previous frame. This is based on the assumption that all pels which belonged to the previous OM should belong to the current change detection mask. However, in order to avoid infinite error propagation, a pel from the previous OM is only labelled as changed if it was also labelled as changed in the CDMs of one of the last L frames, too. For this purpose, a storage for every pel is applied, building a memory (MEM). The value of this storage indicates if the respective pel was set to changed in one of the L previous CDMs; the value L denotes the depth of the memory. Considering that CDMs, OM and MEM are two–dimensional fields and that MEM is the zero–matrix for the first frame, the update of MEM can be formulated as:

$$MEM_{(k)}(x,y) = \begin{cases} L & \text{, if } CDMs_{(k)}(x,y) = 1 \\ max(0, MEM_{(k-1)}(x,y) - 1) & \text{, if } CDMs_{(k)}(x,y) = 0 \end{cases} \quad (2)$$

The current CDMs is then updated by logical OR operation between CDMs and the previous OM, taking into account the memory MEM:

$$CDMu_{(k)}(x,y) = CDMs_{(k)}(x,y) \vee \begin{cases} OM_{(k-1)}(x,y) & \text{, if } MEM_{(k)}(x,y) > 0 \\ 0 & \text{, if } MEM_{(k)}(x,y) = 0 \end{cases} \quad (3)$$

Finally, the resulting change detection mask CDMu is simplified by morphological closing and elimination of small regions, resulting in the change detection mask CDM.

2.2 Calculation of an object mask based on the change detection mask

In the second step, the object mask is calculated from the CDM found in the first step by detecting regions of uncovered background [6]. Therefore, displacement information for pels within the changed areas is used. All pels for which the corresponding negative displacement vector points outside the CDM are assumed to belong to the uncovered background. Now, the CDM is subdivided into one part belonging to the uncovered background and another part which describes the OM. For the displacement estimation hierarchical blockmatching [3] is used. Finally, the boundaries of the resulting object mask are adapted to grey level edges in the corresponding image in order to improve the accuracy of the object mask.

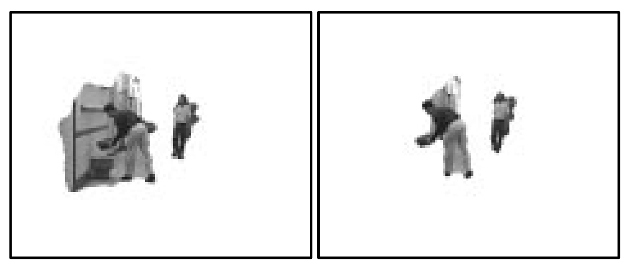
3. EXPERIMENTAL RESULTS

The proposed algorithm was applied to test sequences belonging to two categories: typical videoconference sequences (*Mother Daughter, Akiyo, Claire*) and sequences containing objects with straight forward motion (*Hall Monitor, Container Ship*). All sequences have been tested at a frame rate of 10 Hz in CIF and QCIF format. The resulting object masks are subjectively valued, since the real masks are unknown.

In Fig. 2 two exemplary results of the proposed algorithm are given for the test sequences *Mother Daughter* and *Hall Monitor* and are compared with results from the reference algorithm. As can be seen, the estimated object masks are more obviously belonging to the real objects than those from the reference algorithm. This is firstly due to the relaxation step applying a local thresholding technique, which makes the algorithm more robust against noise. Secondly, the adaptation of the object mask to greylevel edges of the luminance image increases the accuracy of the estimated object boundaries. The temporal coherency, which can not be demonstrated in a printed paper, is also improved due to the application of a memory for the object mask.

The proposed algorithm is currently being investigated within a core–experiment of the ISO/MPEG–4 standardization activities [10]. Future work will deal with an extension of the algorithm for application to sequences from moving cameras, using global motion estimation and compensation.

Reference method Proposed algorithm



Reference method Proposed algorithm

Figure 2: Results of the proposed algorithm compared to the reference for *Mother Daughter* (upper row) and *Hall Monitor* (lower row).

4. REFERENCES

[1] T. Aach, A. Kaup, R. Mester, "Statistical model–based change detection in moving

video”, *Signal Processing*, Vol. 31, No. 2, pp. 165–180, March 1993.

[2] T. Aach, A. Kaup, R. Mester, ”Change detection in image sequences using Gibbs random fields: a Bayesian approach”, *Proc. Int. Workshop on Intelligent Signal Processing and Communication Systems*, Sendai, Japan, pp. 56–61, October 1993.

[3] M. Bierling, ”Displacement estimation by hierarchical blockmatching”, *3rd SPIE Symposium on Visual Communications and Image Processing*, Cambridge, USA, pp. 942–951, November 1988.

[4] CCITT, ”Draft revision of recommendation H.261: video codec for audio visual services at p x 64 kbit/s”, Study Group XV, WP /1/ Q4, Specialist group on coding for visual telephony, Doc. No. 584, November 1989.

[5] P. Gerken, ”Object–Based Analysis–Synthesis Coding of Image Sequences at Very Low Bit Rates”, *IEEE Transactions on Circuits and Systems for Video Technology: Special Issue on Very Low Bit Rate Video Coding*, Vol. 4, No. 3, pp. 228–235, June 1994.

[6] M. Hötter, R. Thoma, ”Image segmentation based on object oriented mapping parameter estimation”, *Signal Processing*, Vol. 15, No. 3, pp. 315–334, October 1988.

[7] M. Hötter, ”Object–oriented analysis–synthesis coding based on moving two–dimensional objects,” *Signal Processing: Image Communication*, Vol. 2, No. 4, pp. 409–428, December 1990.

[8] ITU–T, ”Video coding for narrow telecommunication channels at < 64 kbit/s”, *Draft Recommendation H.263*, January 1995.

[9] MPEG–4 Video Group, *MPEG–4 Video Verification Model Version 3.0*, Doc. ISO/IEC JTC1/SC29/WG11 N1277, July 1996.

[10] MPEG–4 Video Group, *MPEG–4* Automatic Segmentation of moving objects (core experiment N2), Doc. ISO/IEC JTC1/SC29/WG11 MPEG96/841, March 1996.

[11] H.G. Musmann, M. Hötter, J. Ostermann, ”Object–oriented analysis–synthesis coding of moving images”, *Signal Processing: Image Communication*, Vol. 1, No. 2, pp. 117–138, October 1989.

[12] J. Ostermann, ”Object–oriented analysis–synthesis Coding (OOASC) based on the source model of moving flexible 3D objects”, IEEE Trans. on Image Processing, Vol. 3, No. 5, September 1994.

[13] F. Pereira, *MPEG–4: A New Challenge for the Representation of Audio–Visual Information*, keynote speach at Picture Coding Symposium PCS ’96, Melbourne, March 1996