

Segmentation-based Motion Estimation for Video Processing using Object-based Detection of Motion Types

Aishy Amer and Eric Dubois

INRS-Telecommunications

16 Place du Commerce, Verdun, Quebec
H3E 1H6, Canada

Email: amer@inrs-telecom.quebec.ca

ABSTRACT

In this paper, novel techniques for image segmentation and explicit object-matching-based motion estimation are presented. The principal aims of this work are to reconstruct motion-compensated images without introducing significant artifacts and to introduce an explicit object-matching and noise-robust segmentation technique which shows low computational costs and regular operations.

A main feature of the new motion estimation technique is its tolerance against image segmentation errors such as the fusion or separation of objects. In addition, motion types inside recognized objects are detected. Depending on the detected object motion types either ‘object/ unique motion-vector’ relations or ‘object/several motion-vectors’ relations are established. For example, in the case of translation and rotation, objects are divided into different regions and a ‘region/one motion vector’ relation is achieved using interpolation techniques. Further, suitability (computational cost) of the proposed methods for online applications (e.g. image interpolation) is shown. Experimental results are used to evaluate the performance of the proposed methods and to compare with block-based motion estimation techniques.

In this stage of our work, the segmentation part is based on intensity and contour information (*scalar segmentation*). For further stabilization of the segmentation and hence the estimation process, the integration other statistical properties of objects (e.g. texture) (*vector segmentation*) is our current research.

Keywords: Motion Compensation, Motion Estimation, Object-matching, Object Correspondence Finding, Motion Type Detection, Image Segmentation, Binarization, Morphological Operators

1. INTRODUCTION

In time-varying image processing applications, motion estimation is a key technique that determines the quality of the processed image sequences.^{7,6} The most frequently used (and hardware implemented) motion estimation algorithms are block-based (block-matching) algorithms.^{6,2} Three advantages of block-matching algorithms are: 1) easy implementation, 2) better quality of the resulting motion-vector fields compared to previously used methods such as gradient methods or phase plane correlation, and 3) suitability for online applications (e.g. receiver-oriented image interpolation) mostly because of their regular architectures.

Nevertheless, because real objects in real scenes do not coincide with the block boundaries, block-matching algorithms suffer from certain drawbacks. This is particularly true surrounding the object contours where these methods result in erroneous motion vectors leading to discontinuity in the motion-vector fields (causing *ripped contours* artifacts). Another drawback is that the resulting motion vectors inside objects or object regions with a single motion are not homogeneous (producing *ripped region* artifacts). Additionally, using block-based algorithms results in block patterns in the motion-vector field (causing *block patterns* or *blocking* artifacts). These artifacts occur in motion-compensated and motion vector-based applications such as motion-compensated video coding and vector-based image interpolation (upconversion). The human visual system is very sensitive to such artifacts (especially abrupt changes) which are located in the high frequencies. Consequently, image and video segmentation methods that extract information about homogeneous objects should be introduced into motion estimation techniques.^{4,9}

In this paper a novel multilevel, contour-oriented and artifact-tolerant segmentation method and object-matching-based motion estimation are described. Within these methods the whole segmentation and estimation process is divided in simple tasks so that complex arithmetic operations are avoided. In this stage of our work, the segmentation

part is based on intensity and contour information (*scalar segmentation*). For further stabilization of the segmentation and hence of the motion estimation process, the integration of color and other statistical properties (e.g. texture) of objects (*vector segmentation*) is a subject of our ongoing investigations.

The principal aims of this work are 1) to reconstruct motion-compensated images without introducing significant artifacts, 2) to introduce an object-matching technique which shows low computational costs and regular operations, and 3) to extract stable and relevant object features such as intensity, color, contour, and texture. In order not to significantly raise the computational costs of the object-matching, the integration of pre-determined motion information using e.g. a block-based technique is avoided. In addition, because of the i) strong interdependencies between motion-based segmentation and segmentation-based motion estimation and ii) block-based motion estimation artifacts, the integration of pre-determined motion information in segmentation processes using e.g. fast block-based technique could reduce significantly the segmentation quality.

2. ROBUST INTENSITY-BASED IMAGE ANALYSIS

Segmentation- or object-based approaches offer important features to support high-quality video signal processing (e.g. motion estimation, video coding). Video segmentation denotes within this work the technique for extraction of (intensity) homogeneous structures (regions or objects) in time-varying images so that the outlines of these structures coincide as accurately as possible with the physical object outlines in the recorded real scene. Due to psychophysical nature of video segmentation and due to the various nature of its applications (e.g. TV, military, medical images) and requirements (e.g. real-time), the use of heuristics is an unavoidable part of solution approaches.^{11,5}

Image segmentation methods can be distinguished into region and contour-oriented methods. The advantage of region-oriented methods, which can be implemented based on region growth techniques, is robustness in complex video scenes. Due to the region growth techniques, which require high implementation costs, these methods are not suitable for real-time applications. In general contour-oriented methods have low calculation costs. The main disadvantage of contour-oriented methods is their sensitivity to degradation of image quality. For robust, object-based techniques, robust image segmentation methods are needed.

The image segmentation method that is described in this paper (cf. Fig. 1a) is carried out by a multi-region object isolation, a morphological edge detection, a contour analysis, and an object reconstruction. After a threshold-based decomposition of the original gray-level image, the decomposed (bi-level) image is segmented by morphological operators into contour points. Then, the segmented image is further processed by a contour analysis which generates contour chains of the contour points. In the next step — object reconstruction — each of these contours is filled by a unique grey-level which serves as a label for each object in the final object-based processing step.

In this paper, the main enhancement of the segmentation method^{1,2} consists mainly in the new, clustering and multi-threshold-based object isolation. The objectives of the decomposition technique are 1) avoidance of object regions overlapping, 2) better recognition of both light and dark object regions, and 3) more effective object/background separation.

As a result of the segmentation process, a *list of objects* with their features such as area, frame (a bounding box), and the positions of each image in the sequence is provided for further object-based processing. Because of the large amount of object data, object and contour points are compressed using a differential run-length-code. Doing this, the effectiveness (fast access and less memory cost) of the whole algorithm is raised. Other properties of the segmentation method are robustness with respect to noise, low computational cost, and regularity of the main components (e.g. morphological operators).

2.1. Structure-adaptive object isolation

For online application, the transformation of gray-level (or color) images into binary images ('binarization', 'object isolation'), is an essential first step in segmentation techniques. The task of the object isolation (cf. Fig. 1b) is to find out significant potential object regions using an image-structure-adapted multi-threshold function.

Extraction of image-global features: The first binarization step is the determination of a threshold which gives information about the distribution of dark and bright regions in the image. For a robust determination, the combination of global (block-based) and local (histogram-based) decision criteria is needed. Doing this, the threshold is adapted to the contents and changes (e.g. noisy images, cf. Fig. 8b and 8c) in each image of the sequence. Fig. 2 gives an overview of this first binarization step. As can be seen, the image is first divided into N equal blocks. In each

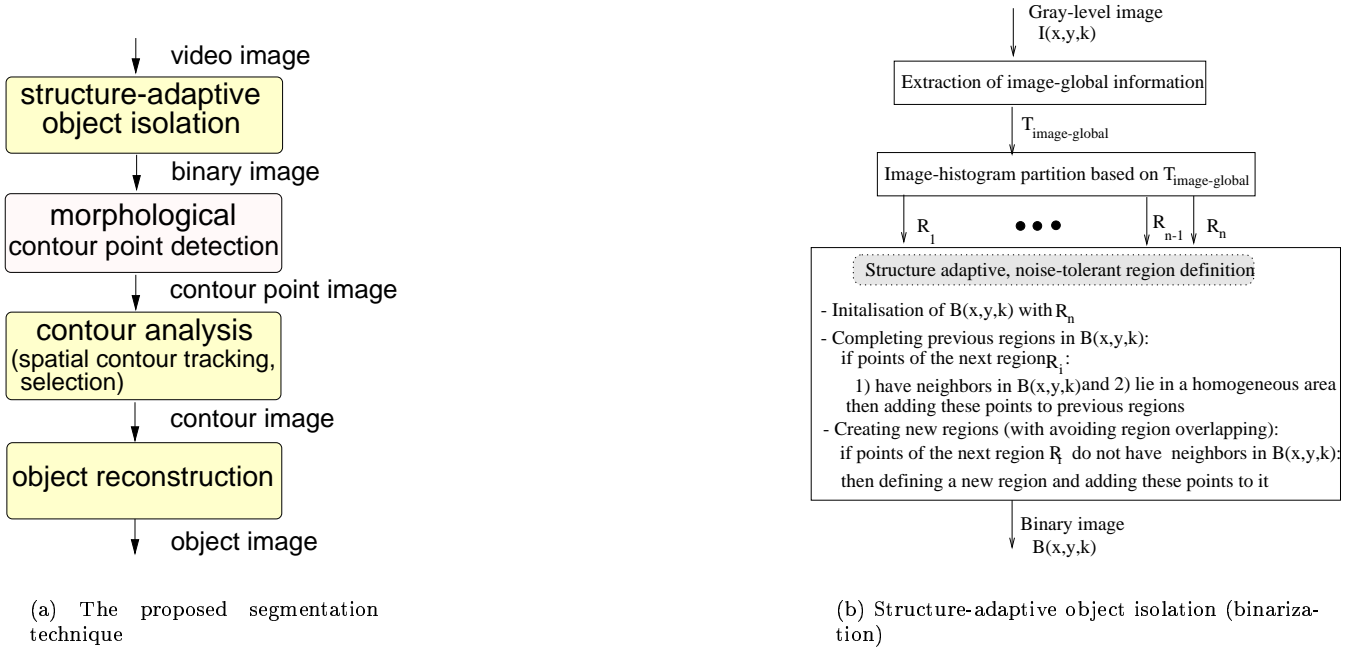


Figure 1.

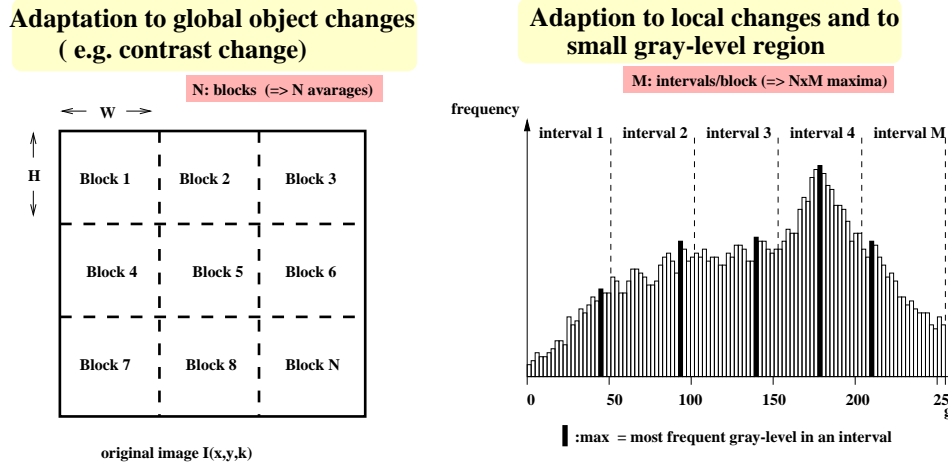


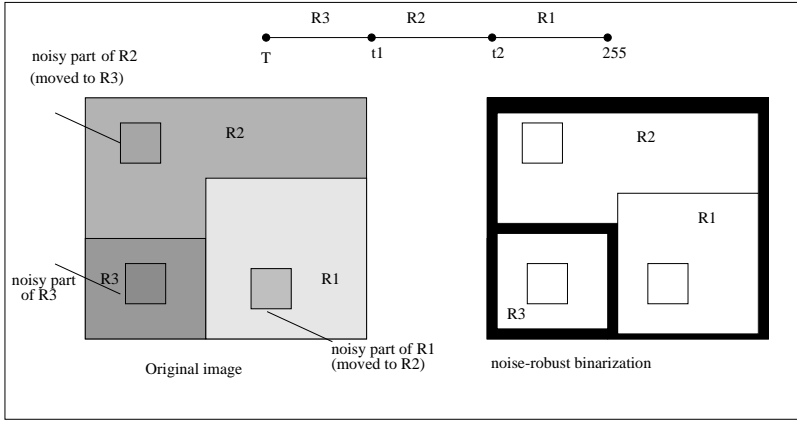
Figure 2. Extraction of image-global gray-level distribution

block, the block-average gray-level is determined. Then, the histogram of each block is divided into M intervals and the most frequent gray-level in each interval is calculated. Finally, the image-wise threshold is determine as follows:

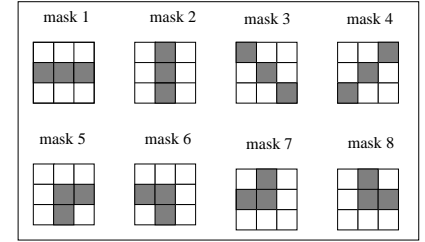
$$T_{block_global} = \frac{\sum(\sum(I(x, y, k)))}{W \cdot H} \quad T_{image_global} = \frac{\sum(\sum(max_{interval}) + T_{block_global})}{N \cdot M + N} \quad (1)$$

Thus, a dynamic adaption of the threshold value is achieved taking into account illumination- and object changes during a video sequence. The calculating function out stands for its interferences invariance.

Structure-adaptive multi-thresholding: The second binarization step is multi-thresholding of the whole gray-level scale (0 – 255). Knowing the information about the concentration and distribution of bright and dark regions in the image, the gray-level scale is divided first into two intervals: $0 - T_{image_global}$ and $T_{image_global} - 255$. Each interval is then divided into sub-intervals as following: In the smallest interval of these two contains dense distribution of the gray-levels. Thus, this interval is refined into sub-intervals. Because the distribution of the gray-levels in larger



(a) An example of the binarization technique



(b) Directions of the homogeneity analyzer

Figure 3.

interval is not dense, it can be divided into coarse sub-intervals. Fig. 3a shows an example of the gray-level-scale partition. Assuming the global threshold T_{image_global} is above 127. This means 1) the bright regions are larger than the dark regions and 2) the bright regions are concentrated on the interval T_{image_global} to 255. In order to take into account this concentration of bright points and to achieve adaptive binarization of this concentration, we need to finely divide this interval into small intervals. Thus, a number of regions are created and each region contains all points in its sub-interval.

The final binarization step is done as following: the output binary image is first initialized with the first region (cf. Fig. 3a). Because of noisy data and because of the thresholding, points of one region could be moved to other regions. Therefore, a point of a new region is added to previous regions if 1) it lays inside a previous region and 2) it lays in an intensity homogeneous area (3x3). Doing this, previous region are completed (cf. Fig. 3a). Border points of a new region do not lay in an intensity-homogeneous area. Therefore, they are not added to the binary image. Hence, region separation is achieved. If a point of a region does not lay in previous added regions and if it lays in an intensity-homogeneous area, it is classified as a point of a new region. Doing this, a new region is created.

Homogeneity detection: Within this work, the demand was for components that are easy to implement. Therefore, the detection of homogeneity is done by using low-pass filters that are able to detect 8 different direction of homogeneities. In Fig. 3b the directions of analysis are depicted. As can be seen special masks for corners are considered. Thus, non homogeneities in object-corners are detected causing stabilization of the binarization process. In this local image analyzer, low pass filters with coefficients $\{-1 \ 2 \ -1\}$ are applied along all given directions for each pixel to be added. If all the directions are below a given threshold, this 3x3 area is defined as intensity-homogeneous. The result is an effective homogeneity detection, which allow robust binarization.

Because of noise, homogeneities could be lost, therefore, if non homogeneity is detected, an additional detection is performed by first noise-reducing of the 3x3 surrounding area of the point to be added. This conditional (mode-oriented) noise reduction is performed using an easy to implement and fast spatial noise reducer.¹⁰ This spatial noise reducer is used here not to reduce the noise in the whole image. It is only used if the first homogeneity detection test failed. Doing that, the total computational costs of the binarization process is not considerably raised.

The main distinguishing aspect of the binarization process is the separation of regions and the noise-robustness which simplifies further segmentation steps such as contour point detection.

2.2. Morphological detection of contour points

Morphological contour point detectors such as

$$C_{image} = I(x, y, k) - (I(x, y, k) \ominus K_{(2 \times 2)})$$

where \ominus denotes erosion operator and $K_{2 \times 2}$ erosion kernel, are very effective applied on binary images.⁸

Standard morphological erosion is defined for kernels around an origin (cf. Fig. 4). To achieve one-pixel-wide and position-precise contours when using the standard erosion a 3x3 kernel is used. Doing this, an incomplete corner detection is given (cf. Fig 4). Thus, a new morphological erosion is used in this work which uses a 2x2 kernel without defining an origin. Therefore, a modified erosion rule has to be defined:

Modified erosion: *If all the four input image points inside the 2x2 kernel are set (white), then all four points in the output image will be set (white), if they were not eroded in a previous step. If at least one of the four input image points inside the kernel is not set, then all the points in the output image are not set (black set or eroded).*

Using this newly defined rule for morphological erosion, the binary image is first being eroded depending on the pixel constellation in the structured element (cf. Fig. 4). Then, the result is subtracted from the binary image, thus producing a contour point image. To achieve high performance of the following image segmentation steps, the accuracy of edge position, the robustness against noise and the detection of one-pixel-wide edges are essential criteria. These goals were taken into account when the new morphological erosion and the structured element was designed. One main feature of the new morphological erosion is its accuracy of projection of the shape of objects (cf. Fig. 4). Due to the small kernel used for the new developed erosion, it requires less line memory and less calculation costs than classical morphological erosion using 3x3 kernels. It is also faster and it is precise on detection of object corners. Its performance is higher, especially in structured objects (e.g. zone-plate). The small expanse of the kernel used allows the realization of the modified erosion with small number of line memories. Furthermore, the use of this erosion in contour detection shows a precise detection in object corners and in diagonal structures.

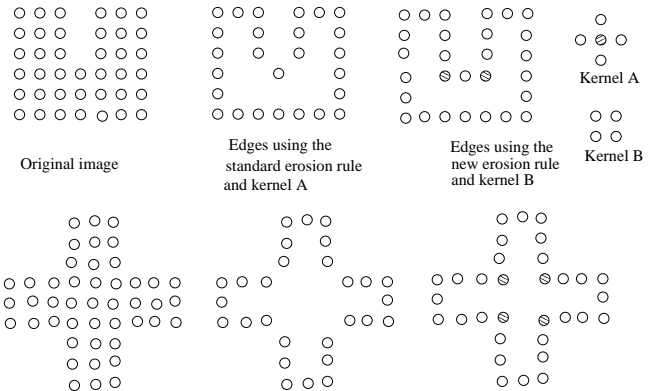


Figure 4. Comparison of morphological edge detection. It is precise on detection of object corners. Its performance is higher, especially in structured objects (e.g. zone-plate). The small expanse of the kernel used allows the realization of the modified erosion with small number of line memories. Furthermore, the use of this erosion in contour detection shows a precise detection in object corners and in diagonal structures.

2.3. Contour and Object reconstruction

The previous image transformation delivers independent contour elements which are not spatial correlated. To become global object information (object contour), discrete chains of the contour points should be generated. Within this processing step, contours which are small or not significant are eliminated. This is due to the fact that small (moving) objects are not important for the perception of an image. In an edge detection algorithm, different artifacts can arise (noise corrupted edges, faulty detection (gaps)). The contour tracking is error tolerant so that small or wrong detected contours, which are not significant for further postprocessing, are eliminated.

In general, the contour image, only characterized by contour points and their spatial relationship, is not sufficient for advanced object-based video processing (e.g. object manipulation and description), which is based on the data of the position of the object points. Therefore an object image, whose discrete objects correspond to the structures included in the input image is reconstructed by simple rules from the contour image.

3. OBJECT-BASED MOTION ESTIMATION

Segmentation-based motion estimation techniques could be distinguished into object-matching, parametric model-based, and motion-vector-field-postprocessing techniques.(cf. Fig. 5a) One motivation for the development of object-based motion estimation algorithms for video postprocessing is the difficulties of block-matching algorithms when applied in advanced video applications. Another motivation is that discontinuities in motion of object regions (e.g. blocks) is, in general, more visible for the human visual system than when the whole object is shifted by a small erroneous motion vector.

In this work, extracted object information (e.g. area, frame, positions, motion direction) is used in a rule-based process with four steps: object correspondence finding (cf. Fig. 6a), motion estimation (cf. Fig. 6b), and motion analysis (cf. Fig. 7):

First, correspondence between objects of image $I(x, y, k - 1)$ (reference objects) and object of the subsequent image $I(x, y, k)$ is found. The object correspondence is based on the following features: object bounding-box height, object bounding-box width, object area, distance between the centers of two objects, and the motion vector difference

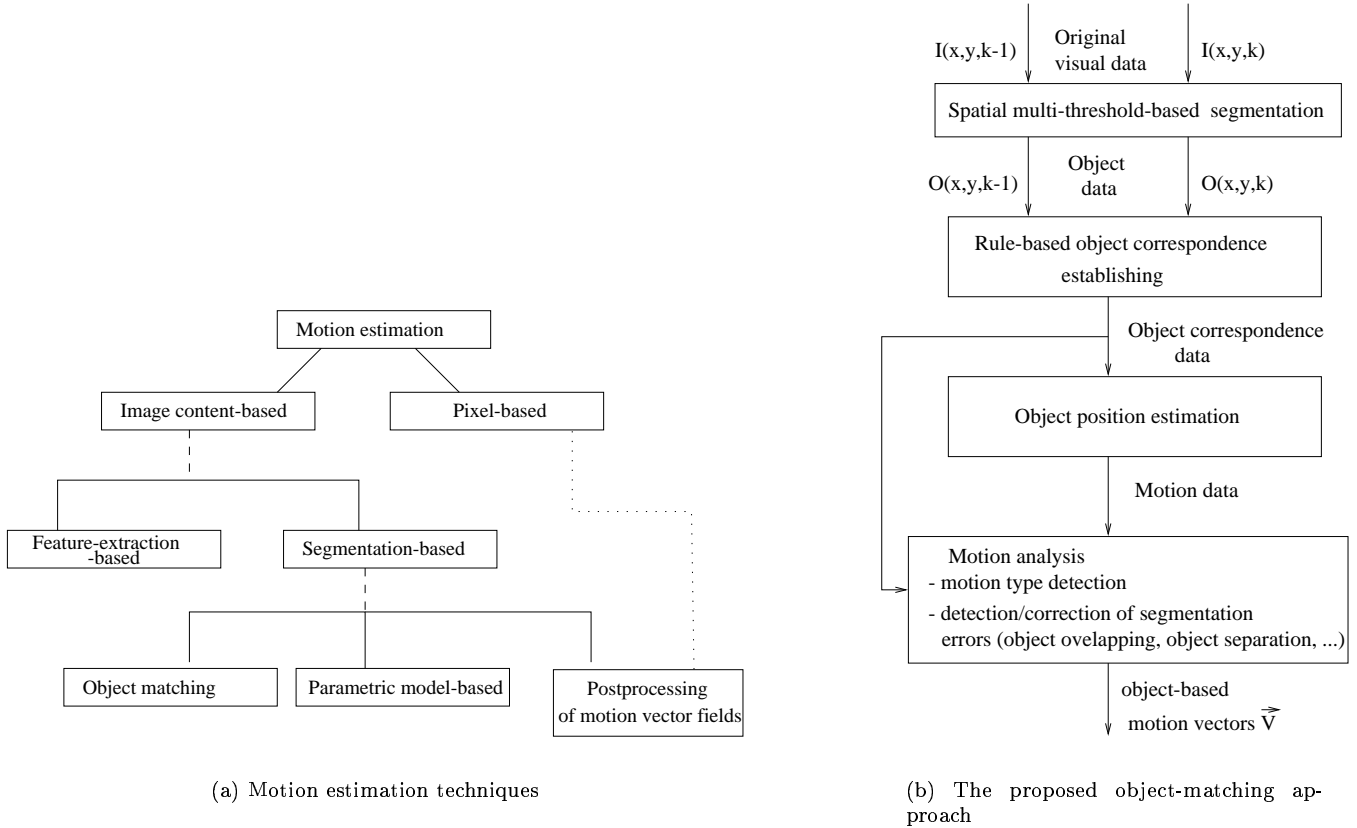


Figure 5.

of the current and new motion vector of the reference object. Second, *object frame* motion is estimated using the displacement of the corresponding object frames and the positions of the corresponding objects. To do this, a frame (a bounding box) surrounding each object is determined using pre-obtained object positions (rows, columns). Third, the estimated motion inside objects is analyzed and different motion types (translation, rotation, zoom, and acceleration) as well as image segmentation errors (*object-fusion* or *object-separation*) are detected. Finally, motion vectors from the second step are adapted to the detected motion types. As a result, depending on the motion type detection, one or more motion vectors for each recognized object are estimated. Due to the interlacing techniques object motion between two interlaced images is changed by one pixel. Thus, the object motion has to be corrected.

One of the main features of the new motion estimation technique is its tolerance against image segmentation errors such as the fusion or separation of objects. In addition, motion types inside recognized object are detected. Depending on the detected object motion types either ‘one object/one motion-vector’ relations or ‘one object/several motion-vectors’ relations are established. In the case of translation and rotation for example, objects are divided into different regions and a ‘one region/one motion-vector’ relation is achieved by interpolation of motion vector found in the object-bounding-box motion estimation step.

Common to all object-based motion estimation methods, is a huge amount of calculations because of the several refining, hierarchical, or interdependent steps. Therefore, within this concept, non-regular-structured and complex operations are avoided, particularly in the image segmentation. The object information is used to estimate the object motion by a homogenization process within the object regions in a non-hierarchical, but regular way without considerably raising the computational costs.

Another prerequisite for this algorithm is robustness. In the case of non-translatory motion, object-based algorithms often yield worse results than standard motion estimators. Because of this, non-translatory motion types are detected here in a second step by an analysis of the estimated motions of an object. In such a case, the estimation

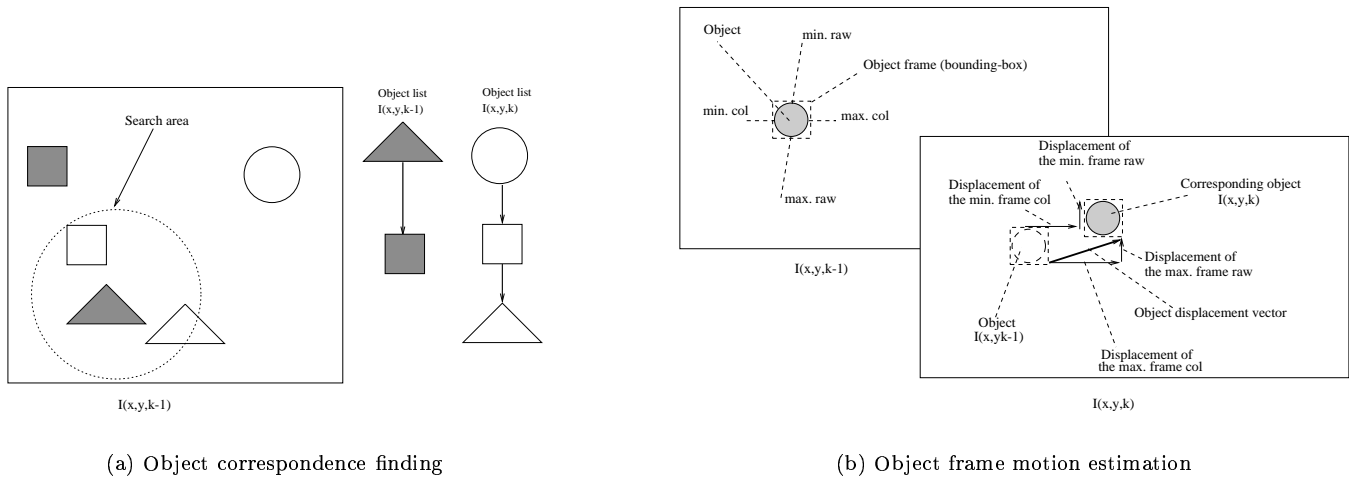


Figure 6.

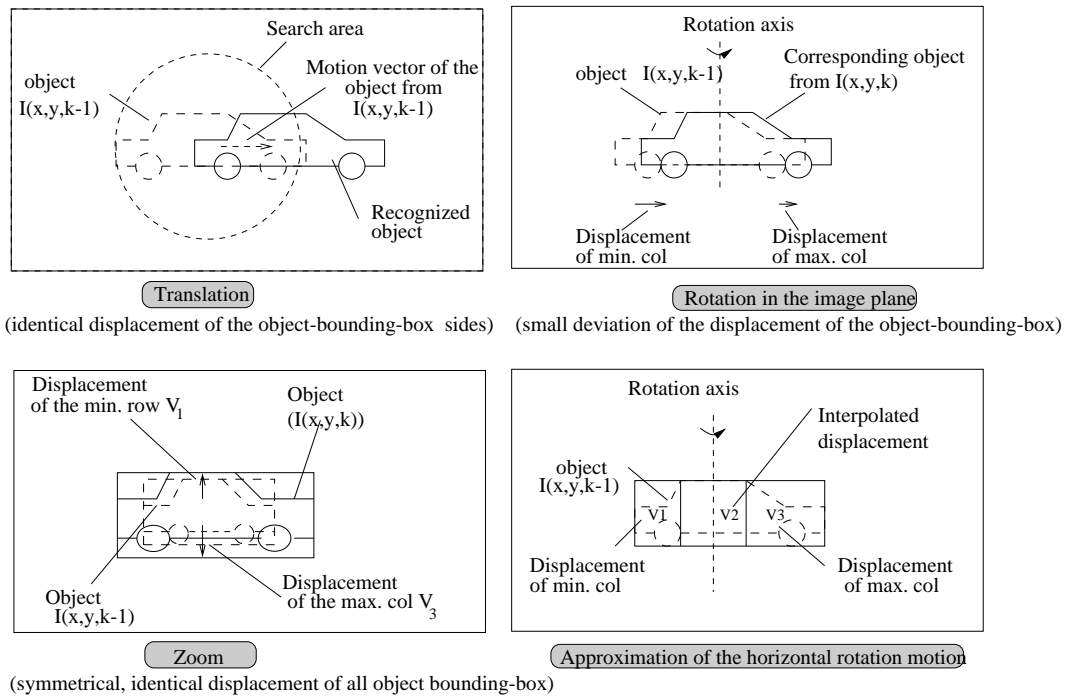


Figure 7. Detection of motion types

process is adapted to the detected motion form.

4. EXPERIMENTAL RESULTS

Experimental results are used to evaluate the performance of the proposed methods and to compare with a block-based motion estimation technique.

Computational costs: Table 1 depicts the computational costs for the segmentation and the object-matching in comparison to the computational costs required for parallel-predictive block-matching.³ The data given in the table are measured in seconds by the 'clock' command of 'C' on a SUN4 machine with a SPARC2 processor. The data are given for one single standard TV-field (720×288) and averaged over seven sequences: 'flower garden', noisy 'flower garden', 'BBC-car', 'train', 'prlcar', 'table tennis', and an artificial sequence 'arti'. Each of the sequences consists

of 60 images. As can be seen, the computational cost for the object-based motion estimation is about 1/38 of the

Algorithm	average execution time
Binarization	0.99
Morphological contour point detection	0.09
Contour reconstruction	0.5
Object reconstruction	2.1
Segmentation (total)	3.68
Object-matching	1.7
Object-based motion estimation	5.38
parallel predictive block-matching	209.5

Table 1. Average computational cost (in seconds) of the algorithm elements

computational cost of the block-matching method. This method has a complexity of about forty times lower than that of a Full-search algorithm.⁴ Further, regular operation (regular binarization, binary morphological operator, regular contour point tracking) are applied. Thus, the object-matching seems to be suitable for online video applications.

Robustness: The image-wise isolation of potential object regions was shown to be noise-robust by various sequence simulations. The developed morphological contour point detector (and the modified morphological erosion) has been compared to various methods (gradient- and morphological-based). It has shown noise robustness, accuracy of point positions, and it yields gap-free edges and it preserved the shape of objects, especially on corners (cf. 4). In particular, the contour point detector and the contour point tracking have very low calculation costs. The robustness of the whole segmentation method can also be demonstrated very noisy (white and impulsive) image sequences (cf. Fig. 8b, 8c)

Figures 8 to 12 show simulation results of the proposed segmentation and object-matching based motion estimation techniques and summarize their performance. In conclusion, 1) noise-robust segmentation (Fig. 8) can be achieved, 2) the homogeneity of the resulting motion vector fields (Fig. 10 and 11) can be distinctively increased, and 3) the interpolation quality (Fig. 12) especially in moving objects can be raised.

Due to the missing object points which are lost during the intensity-based binarization phase, the performance of the object-matching technique in intensity non-homogeneous regions shall be increased. The introduction of texture and color homogeneity criteria shall increase the performance of the proposed algorithm.

5. CONCLUSION AND OUTLOOK

In this paper, an approach to segmentation-based motion estimation in video sequences is introduced. This novel technique consists of two main strategies: a luminance-oriented, noise-robust, and fast image segmentation that generates visually relevant disjoint objects or object regions and an explicit matching of these arbitrary-shaped objects in order to estimate their motion. This estimation strategy takes account of 1) both translatory and non-translatory motion types and 2) image segmentation errors such as object-fusion or object-separation (*error-tolerance*).

The algorithmic performance of the algorithm is noise robust and improves the vector quality. The computational costs of the whole concept are only about 1/38 of the computational cost of a parallel-predictive block-matching algorithm. This block-matching algorithm itself has a complexity which is about forty times lower than that of Full-search algorithm.

Therefore, the proposed algorithms seems to be well suited for the demands of online video systems as they: 1) improve the quality of the motion-vector fields, 2) provide low computational cost, and 3) remain stable in the presence of noise.

Current research is oriented at introduction of color information and other statistical local and global properties of the object to enhance the segmentation quality. In addition, we are working on integrating inter-image segmentation and object correspondence dependencies, in order a) to reduce the complexity of the segmentation-based motion estimation and b) to increase the stability of the whole algorithm. Further, simulations have to be carried out to evaluate applications (e.g. image interpolation) gain of the new concept.

ACKNOWLEDGMENTS

The authors thank Arnold Gasch for helping by the simulations and for drawing some of the figures.

REFERENCES

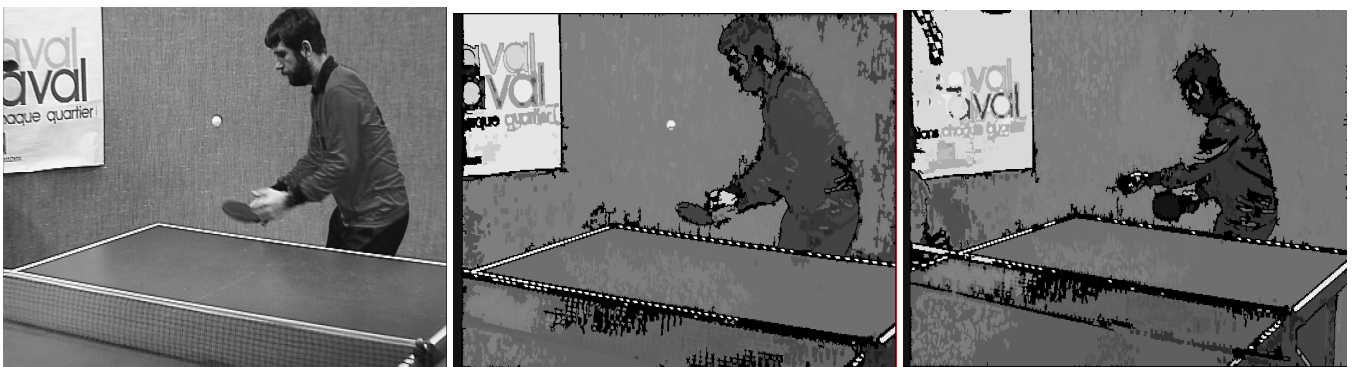
1. A. Amer and S. Reichert, A multilevel method for robust object recognition in video sequences, the 9th Aachen Symposium on 'Signaltheorie', Aachen, Germany, March 1997, ISBN 3-923219-10-5 (in German).
2. H. Blume, A. Amer, and H. Schröder, Vector-based postprocessing of MPEG-2 signals for digital TV-receivers, IS&T SPIE's Annual Symposium on Electronic Imaging, Conference in Visual Communication and Image Processing, vol. 3024, pp. 1176-1187, San Jose, USA, February 1997.
3. Blume, H.: Vector-Based Nonlinear Upconversion Applying Center Weighted Medians, IS&T/SPIE Symposium on Electronic Imaging, Nonlinear Image Processing VII, San Jose (USA): February 1996.
4. H. Blume and A. Amer, Parallel predictive motion estimation using object recognition methods, European Workshop and Exhibition on Image Format Conversion and Transcoding, Berlin, Germany, March 1995.
5. P. Correia, F. Pereira, The role of analysis in content-based video coding and indexing, Signal Processing, vol. 66, pp. 125-142, 1998.
6. G. de Haan, et al., IC for motion compensated 100 Hz TV with smooth movie motion mode, Proceedings of the ICCE, Chicago, USA, June 1995.
7. E. Dubois and J. Konrad, Estimation of 2-D motion fields from image sequences with application to motion compensated processing, in Motion Analysis and Image Sequence Processing (M. Sezan and R. Lagendijk, eds.), ch. 3, pp. 53-87, Kluwer Academic Publisher, 1993.
8. Haralick, R.M., Shapiro L.G.: Computer and Robot Vision, Volume I,II, Reading, Addison-Wesley 1992
9. A. Gash, Object-based vector analysis for restoration of video signals, Master's Thesis, Dept. of Elect. Eng., University of Dortmund, July 1997 (in German).
10. K. Jostschulte, A. Amer, M. Schu, H. Schröder, Perception Adaptive Temporal TV-Noise Reduction Using Contour Preserving Pre-filter Techniques, IEEE Trans. on Consumer electronics, vol. 44, no. 3, pp. 1091-1096, August 1998.
11. T. Pavlidis, Structural Pattern Recognition, Springer Verlag, Berlin 1977.



(a) Original 'flower garden' (img#1)

(b) Recognized objects (img#1)

(c) Recognized objects of noisy (30 dB & 1% impulsive noise) 'flower garden' (img#1)



(d) 'table tennis' (img#8)

(e) Recognized objects (img#8)

(f) Recognized objects (img#27)



(g) Original 'BBC-car' (img#1)

(h) Recognized objects (img#1)

(i) Recognized objects (img#16)

Figure 8. Segmentation results of 'flower garden' and 'BBC-car'

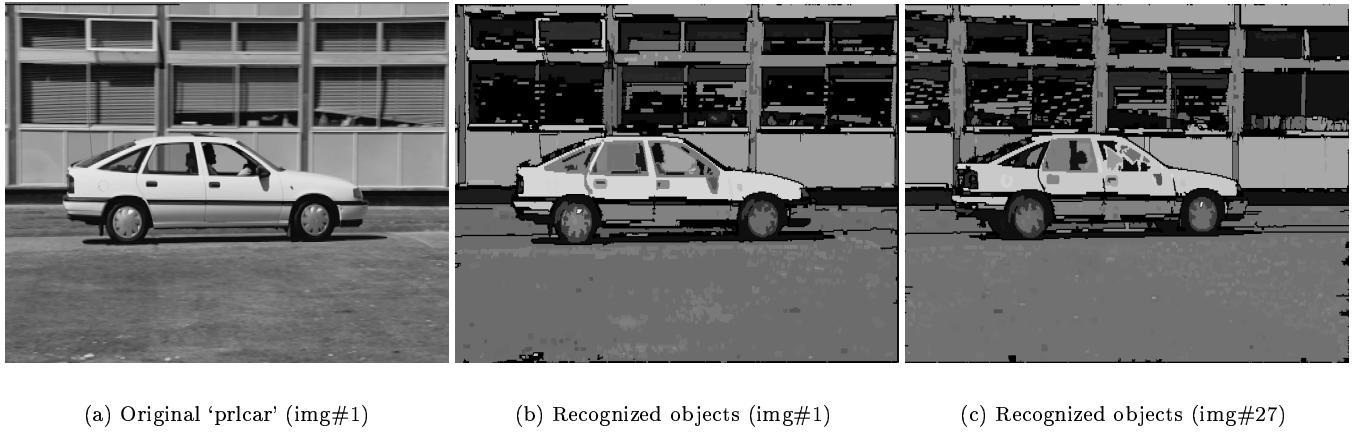


Figure 9. Segmentation results of 'prlcar' images

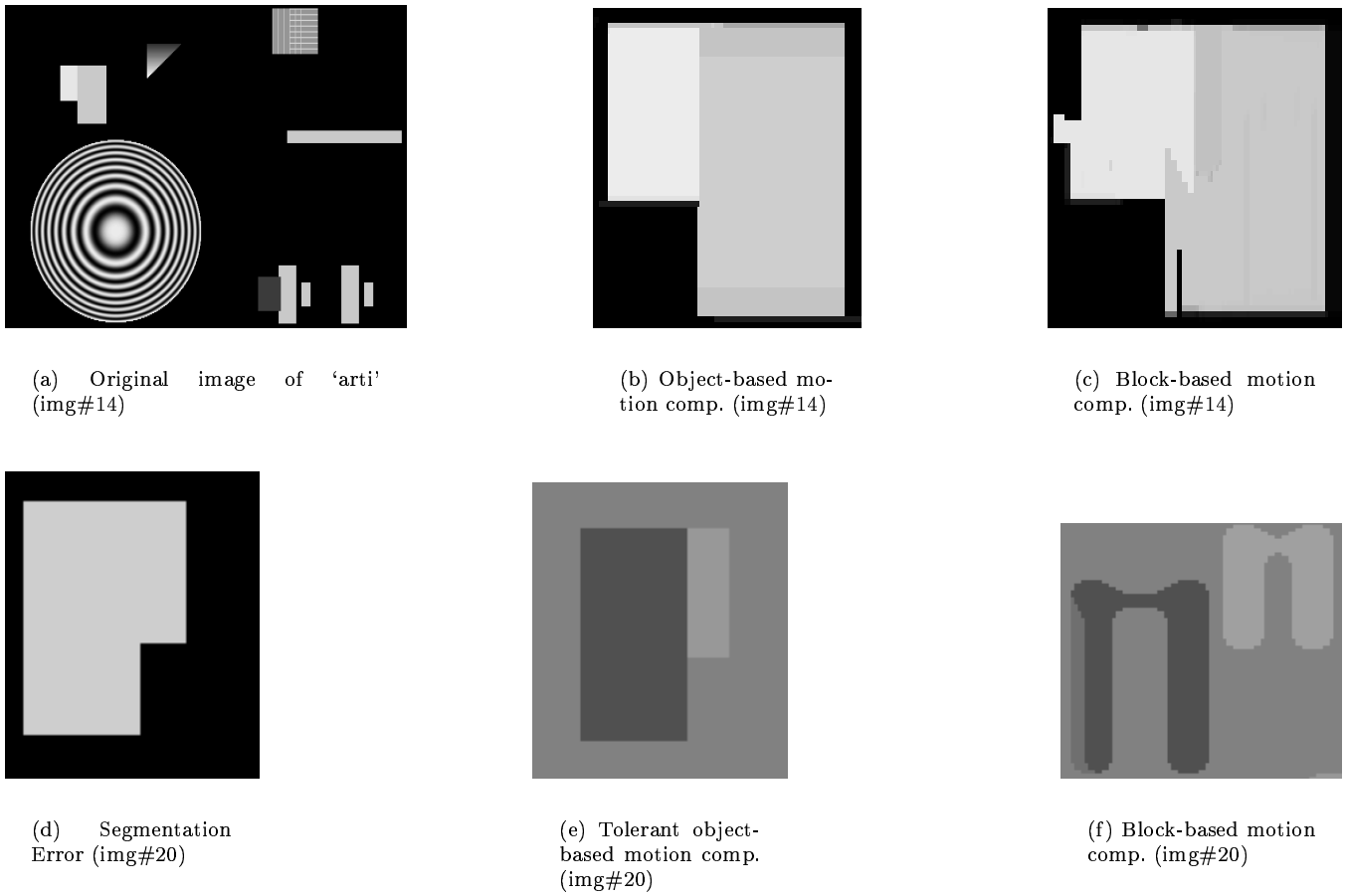
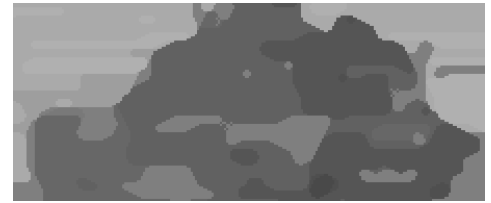


Figure 10. Behavior of the algorithm by object overlapping and segmentation errors



(a) 'BBC-car': object-based estimated motion (img#32)



(b) 'BBC-car': block-based estimated motion (img#32)



(c) 'BBC-car': object-based estimated motion (img#33)



(d) 'prlcar': object-based estimated motion (img#33)



(e) 'prlcar': object-based estimated motion (img#44)



(f) 'prlcar': block-based estimated motion (img#33)

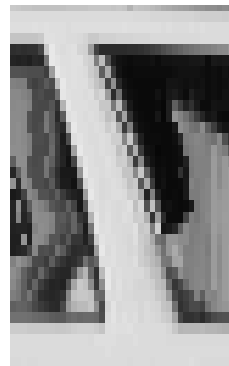
Figure 11. Approximation of rotation and translation motion of 'BBC-car' and 'prlcar' objects



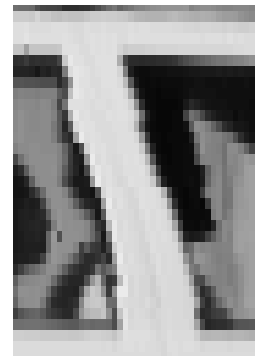
(a) Object-based mot. comp. (img#12)



(b) Block-based mot. comp. (img#12)



(c) Object-based Mot. comp. (img#13)



(d) Block-based mot. comp. (img#13)

Figure 12. Comparison of motion-compensated regions of objects of 'prlcar' images