Motion-Based Object Detection and Tracking in Color Image Sequences

Bernd Heisele[†]
Image Understanding Group (FT3/AB)
DaimlerChrysler Research Center
Ulm, D-89013, Germany

Abstract

In this paper we present an algorithm for detecting objects in a sequence of color images taken from a moving camera. The first step of our algorithm is the estimation of motion in the image plane. Instead of calculating optical flow, tracking single points, edges or regions over a sequence of images, we determine the motion of clusters, built by grouping of pixels in a color/position feature space. The second step is a motion-based segmentation, where adjacent clusters with similar trajectories are combined to build object hypotheses. Our application area is vision-based driving assistance. The algorithm has been successfully tested in traffic scenes containing objects, such as cars, motorcycles, and pedestrians.

Keywords: clustering, color, motion estimation, motion segmentation, tracking.

1. Introduction

A common approach to extracting moving objects from the background is segmentation based on image motion. Spatio-temporal changes in the image data are used to estimate image motion, which is the projection of 3D-motion in the image plane. In a subsequent motion segmentation, each image is divided into segments corresponding to objects with different motion properties. There are two important requirements for this approach: dense motion estimates must be calculated and motion discontinuities must be preserved.

Many algorithms estimate the optical flow (velocity field in the image plane) by calculating the spatio-temporal gradient of the local intensity distribution [1]. The main problem is that the optical flow cannot be reliably estimated in image parts with approximately uniform intensity. To overcome this problem, a smoothness constraint for the whole image is introduced in [1, 2]. It results in the estimation of a dense, smoothly varying velocity field. However, large estimation errors occur at discontinuities in the true velocity field. To preserve motion discontinuities, constraints can be applied to image regions, where a continuous velocity field can be assumed. In [3, 4], intensity-based segmentation techniques are used to determine such image regions.

Another group of motion estimation techniques is based on tracking image features over a sequence of images. Tracking points with prominent features is described in [5, 6]; tracking edges is proposed in [7]. However, the density of tracked points or edges is often not sufficient for motion-based segmentation. We have shown in [8], that tracking colored regions determined by color segmentation yield good results for detecting vehicles on highways. Despite the early reduction of data in the color segmentation, this approach nevertheless generates motion information across the *whole* image. Moreover, discontinuities in image motion, which are likely to coincide with border lines of color segments, are preserved. Finally, the task of motion segmentation is simplified to grouping of a few hundred color segments instead of thousands of pixels. However, performing color segmentation for each image independently, sometimes lead to unstable segmentation results over time. Assuming image sequences with small frame-to-frame changes, we suggested a different algorithm [9] for time consistent color segmentation based on clustering. Each image is clustered in a feature space, defined by the color (R, G, B) and the position (x, y) of a pixel. The final partitioning is fully described by the so called prototypes, which are the centroids of the clusters in the color/position feature space. The first image of

[†]Present address: Center for Biological and Computational Learning, M.I.T., Cambridge, MA 02142, USA, heisele@ai.mit.edu

a sequence is segmented by a divisive clustering algorithm, which only needs to be initialized with the number of clusters. In the consecutive frames the prototypes of the previous frame are shifted in the feature space by parallel k-means clustering to fit the new image data. Thus we obtain consistent segmentation results over time. Moreover no explicit matching of corresponding clusters is required. In this contribution we improve the approach by adding a Kalman filter, which predicts the position of each prototype, based on its given trajectory in the image plane (see Figure 1).

Once image motion has been estimated, a motion segmentation is applied to partition each image into object hypotheses. While most segmentation approaches are based on the instantaneous image motion, the proposed method uses motion information accumulated over several frames. It combines adjacent clusters with similar trajectories into object hypotheses. Since we are mainly interested in objects which are close to the observer and move relative to the camera, only those clusters are considered which show a significant displacement in the image plane.

The outline of the paper is as follows: In Sec. 2 we briefly describe the estimation of image motion by clustering. The motion-based segmentation is explained in Sec. 3. Our results are presented in Sec. 4. The paper is summarized in Sec. 5.

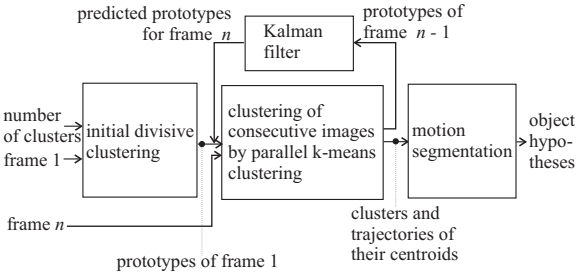

Fig. 1: Overview

2. Motion Estimation by Clustering

Each pixel n in the image is described by a feature vector $\mathbf{f}_n$, containing its color in the RGB space and its position in the image: $\mathbf{f}_n = (R_n, G_n, B_n, w{\cdot}x_n, w{\cdot}y_n)$, where x_n is the horizontal and y_n the vertical position of pixel n in the image plane. The weighting between color and position is determined by the factor w. The task of clustering is to find a given number R of prototypes $\mathbf{p}_r$, which minimize the sum of quantization errors: $\sum_n \|\mathbf{f}_n - \mathbf{p}_{r(n)}\|^2$, where $\mathbf{p}_{r(n)}$ is the prototype closest to the feature vector $\mathbf{f}_n$ in the color/position feature space. The weighting factor w has strong influence on the clustering result. As shown in Figure 2, large values of w result in compact, square shaped clusters. Decreasing w will diffuse the clusters in the image plane. For our purposes $w = 1$ seemed to be reasonable choice.

For the segmentation of the first image we use a divisive vector quantization [10] which partitions the image into a preselected number of clusters.

For each following image the predicted prototypes of the preceeding image serve as seeds for the parallel k-means clustering [11] in the current image. This clustering produces a new set of prototypes for the next image. Parallel k-means clustering consists of two steps per iteration:

$$\begin{aligned} \mathcal{C}_r\,(i+1) &= \{\text{pix } n \mid \|\mathbf{f}_n - \mathbf{p}_r(i)\| \leq \\ &\qquad \|\mathbf{f}_n - \mathbf{p}_s(i)\| \;\forall\; s\} \quad (1) \\ \mathbf{p}_r\,(i+1) &= \frac{1}{\text{size }[\mathcal{C}_r(i+1)\,]} \sum_{n \in \mathcal{C}_r(i+1)} \mathbf{f}_n \end{aligned}$$

Where $\mathcal{C}$ is the set of pixels in a cluster, i is the counter for the iterations, r is the index of a cluster and its corresponding prototype, and size $[\mathcal{C}]$ is the number of pixels in cluster $\mathcal{C}$.

In the partitioning step each pixel n, characterized by its feature vector $\mathbf{f}_n$, is assigned to the cluster $\mathcal{C}_r(i+1)$ with the closest prototype $\mathbf{p}_r(i)$. After that, the prototypes $\mathbf{p}_r(i+1)$ are recomputed as the average of the data in their clusters. Each of these two alternating steps reduces the sum of quantization errors until no further changes occur. At this stage a local minimum of the sum of quantization errors has been obtained.

Assuming a continuous motion behavior of clusters, prediction techniques can be used to increase the robustness of motion estimation. For each prototype we predict its positon x, y in the next frame with a standard Kalman filter. The color features are kept unchanged. Since there are various types of objects in traffic scenes, a rather general kinematic model [12] is chosen for the Kalman filter: The motion along the x and y axes are regarded to be decoupled, x and y motions are therefore predicted by two separate filters. The motion of the cluster is assumed to have approximately constant velocity. To account for slight changes in the velocity, the time continuous accelaration is modeled as white noise. The discrete state equation with a sampling period T is:

$$\mathbf{s}(k+1) = \mathbf{A}\,\mathbf{s}(k) + \mathbf{w}(k) \quad (2)$$

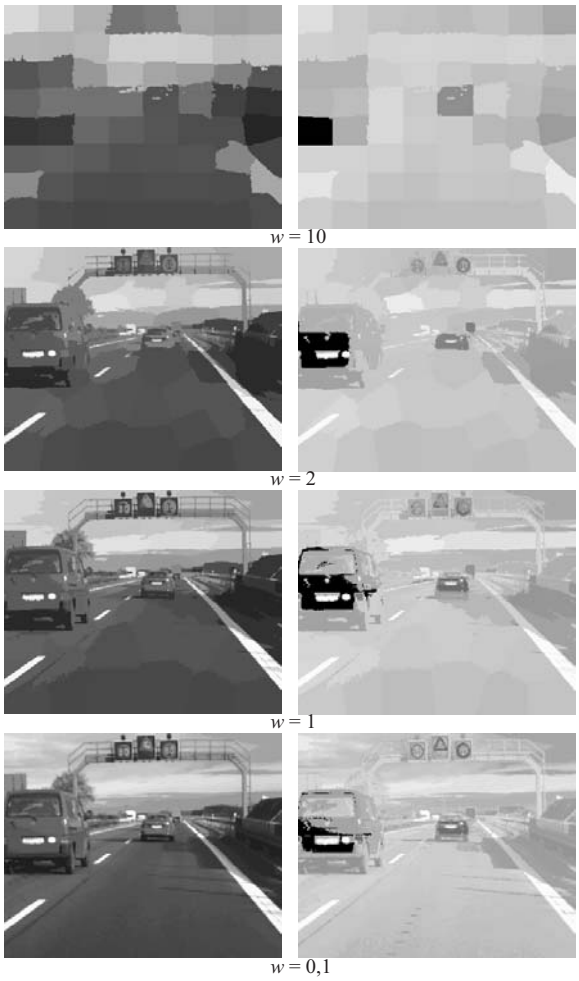

Fig. 2: Influence of weighting factor w on combined color/position clustering. The left column shows the clustered images. In the right column the cluster belonging to the van has been highlighted to illustrate the decreasing compactness of the cluster with decreasing w.

with

$$\mathbf{A} = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} \quad (3)$$

$$\mathbf{Q} = E\left(\mathbf{w}(k)\mathbf{w}(k)^T\right) \quad (4)$$

$$= \begin{pmatrix} \frac{1}{3}T^3 & \frac{1}{2}T^2 \\ \frac{1}{2}T^2 & T \end{pmatrix} \sigma_w^2$$

The measurement equation for the one-dimensional position is:

$$p(k) = \mathbf{C}\,\mathbf{s}(k) + n(k) \quad (5)$$

with

$$\mathbf{C} = \begin{pmatrix} 1 & 0 \end{pmatrix} \quad (6)$$

$$E\left(n^2(k)\right) = \sigma_n^2 \quad (7)$$

The parameters of the filter are the power spectral density of the process noise σ_w^2 and the measurement noise σ_n^2. An example for the improvement of tracking achieved by Kalman filtering is shown in Figure 3.

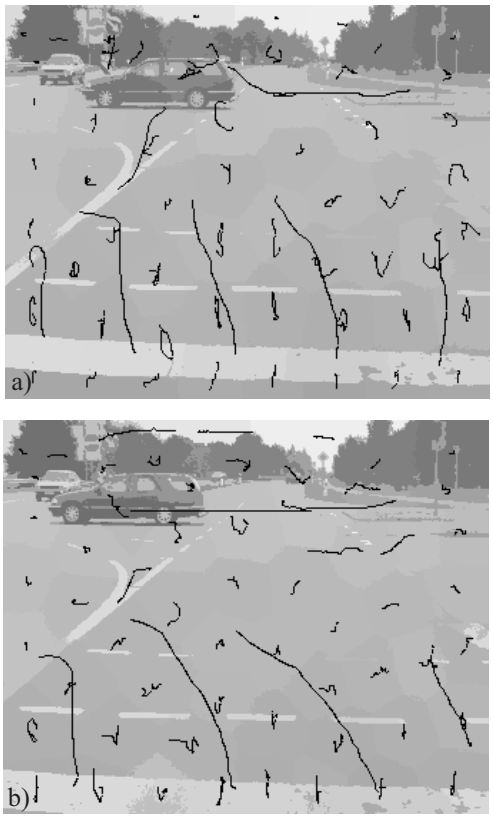

Fig. 3: Approaching a road crossing. The trajectories of the clusters are drawn as black lines. Without prediction, the fast moving car could not be tracked over the whole sequence (a). Tracking was successful when using Kalman filtering (b).

3. Object Detection by Motion Segmentation

The proposed method for motion segmentation combines adjacent clusters with similar trajectories into object hypotheses. It is based on the assumptions that image motion across a cluster can be approximated by the motion of the centroid of the cluster and that clusters belonging to the same object have roughly parallel trajectories. These assumptions have proven to be acceptable in most of the image sequences investigated in this paper. However, they are not valid in the presence of non-rigid objects, objects rotating about the optical axis, or large objects, moving along the optical axis.

In motion segmentation we only consider clusters which fulfill the following conditions: minimum length and minimum reliability of the trajectory. The first condition eliminates clusters, corresponding to objects which are far away from the camera or do not move relative to the observer. The second condition accounts for the well known correspondence problem. It is apparent in large, uniformly colored image areas (e.g. road, sky), which are partitioned into a set of similar clusters. To assign a low reliability to the trajectories of these clusters, we calculate the distances between clusters in the color/position feature space. The measure for the reliability of a trajectory increases linearly with the mean distance of a cluster centroid to its nearest neighbors. An example for the reliability measure is shown in Figure 4. Figure 4 (a) shows the result of clustering in color/position feature space, Figure 4 (b) shows the reliability measures for each cluster. Bright values indicate high reliability.

Fig. 4: Illustration of the measure for the reliability of trajectories. The result of clustering is shown in (a). The reliability measures for each cluster is shown in (b). Bright values indicate high reliability.

If two clusters $\mathcal{C}_1$ and $\mathcal{C}_2$ are adjacent to one another in the image plane and fulfill the above mentioned conditions, the similarity between their trajectories is determined by the following measure:

$$\rho(\mathcal{C}_1,\mathcal{C}_2) = \left(1 - \frac{|l_{\mathcal{C}_1} - l_{\mathcal{C}_2}|}{l_{\mathcal{C}_1} + l_{\mathcal{C}_2}}\right) \cdot \frac{\sum_{k=M}^{N} \left(\mathbf{x}_{\mathcal{C}_1}(k) - \bar{\mathbf{x}}_{\mathcal{C}_1}\right)^T \left(\mathbf{x}_{\mathcal{C}_2}(k) - \bar{\mathbf{x}}_{\mathcal{C}_2}\right)}{\sqrt{\sum_{k=M}^{N} \| \left(\mathbf{x}_{\mathcal{C}_1}(k) - \bar{\mathbf{x}}_{\mathcal{C}_1}\right) \|^2 \sum_{k=M}^{N} \| \left(\mathbf{x}_{\mathcal{C}_2}(k) - \bar{\mathbf{x}}_{\mathcal{C}_2}\right) \|^2}} \quad (8)$$

where

$$\bar{\mathbf{x}} = \frac{1}{1+N-M} \sum_{k=M}^{N} \mathbf{x}(k) \quad (9)$$

$$l = \sum_{k=M}^{N-1} \| \left(\mathbf{x}(k+1) - \mathbf{x}(k)\right) \| \quad (10)$$

The position of a centroid of a cluster in frame k is $\mathbf{x}(k) = (x(k), y(k))^T$. The interval $[M, N]$ is a time window for matching the trajectories. The first factor of Eq. 8 measures the difference in length. The second factor is the linear correlation between the trajectories. If the trajectories are parallel the correlation is 1; if they are perpendicular to each other it is 0. In the case of contra-rotating trajectories it is -1.

If $\rho(\mathcal{C}_1, \mathcal{C}_2)$ exceeds a given threshold ρ_{min}, both clusters $\mathcal{C}_1$ and $\mathcal{C}_2$ are combined into the same object hypothesis.

4. Results

Experiments were carried out on traffic scenes, taken on highways and in cities. A 3-chip CCD camera, connected to digital video recorder (Y:U:V, 720×576 pixels, 25 frames/sec.), was used for taking images. The resolution of the images was reduced to half the number of rows and columns (360×288 pixels). Using divisive vector quantization techniques, the preselected number of clusters should be close to 2^n in order to obtain clusters with similar deviations. In our experiments 128 clusters were sufficient to cover all relevant details of the images. Only one iteration was calculated in the k-means clustering—it took approximately 1.2 seconds on a Pentium II 300 MHz. The last 5 points of the trajectories were used in the motion segmentation and the minimum trajectory length over 5 frames was set to 10 pixels. The threshold ρ_{min} was set to 0.95.

Figure 5 shows results of the algorithm applied to four sequences. In sequence (a), our test-car was traveling at a speed of about 80 km/h; the car on the left lane passed us at a speed of approximately 100 km/h. It lasted 10 frames from the time the car entered the field of view until it was detected. The reason for the

late detection is that clusters jumped from parts of the background onto the car, when it entered the image. Consequently their trajectories differed from the real image motion. Sequence (b) was taken at a road crossing with a stationary camera. It shows a passing motorcycle. In sequence (c) our car slowly approached a pedestrian crossing. The pedestrians were combined into one object hypothesis, because they were walking close to each other with approximately the same speed. Sequence (d) illustrates the robustness of the algorithm in the face of partial occlusions and shape variations.

Surprisingly, the algorithm allows walking pedestrians to be detected. Due to the similarity in color, both legs or arms are often combined into the same cluster. The trajectory of such a cluster reflects the motion of the centroid of both legs or arms, which is similar to that of the other body parts. Combining both legs or arms into the same cluster violates our initially stated assumption of roughly constant image motion across a cluster. However, it enables us to segment pedestrians without modeling human motion.

5. Summary

In our approach we presented a new method for detecting moving objects from color image sequences taken with a moving camera. The algorithm estimates image motion by tracking clusters determined in the color/position feature space. Each image is divided into given number of clusters by grouping pixels of similar color and position. To achieve consistent clusters over time, clustering of each image is based on clustering results of previous images. In this context, Kalman filters are used to predict dynamic changes in cluster positions. Finally a motion segmentation builds object hypotheses, by combining adjacent clusters with significant image motion and nearby parallel trajectories. When tested in traffic scenes, the algorithm successfully segmented various types of moving objects, such as cars, motorcycles, and even pedestrians.

References

[1] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[2] H.-H. Nagel and W. Enkelmann. An investigation of smoothness constraints for the estimation of displacement vector fields from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(5):565–593, 1986.

[3] M. J. Black and A. D. Jepson. Estimating optical flow in segmented images using variable-ordered parametric models with local deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(10):972–986, 1996.

[4] M. Gelgon and P. Bouthemy. A region-level graph labeling approach to motion-based segmentation. In *Proc. Computer Vision and Pattern Recognition*, pages 514–519, San Juan, 1997.

[5] J. Weng, J. Ahuja, and N. Huang. Matching two perspective views. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(8):806–825, 1992.

[6] S. T. Barnard and W. B. Thompson. Disparity analysis of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(4):333–340, 1980.

[7] G. L. Foresti and V. Murino. Moving object recognition from an image sequence for autonomous driving. In V. Capellini, editor, *TVIP-MOR 4th International Workshop 1994*, pages 383–390, Florenz, Italien, 1994. Elsevier Science Amsterdam.

[8] B. Heisele and W. Ritter. Obstacle detection based on color blob flow. In *Proc. Intelligent Vehicles Symposium*, pages 282–286, Detroit, 1995.

[9] B. Heisele, U. Kressel, and W. Ritter. Tracking non-rigid, moving objects based on color cluster flow. In *Proc. Computer Vision and Pattern Recognition*, pages 253–257, San Juan, 1997.

[10] Y. Linde, A. Buzo, and R. Gray. An algorithm for vector quantizer design. *IEEE Transactions on Communications*, 28(1):84–95, 1980.

[11] R. O. Duda and P. E. Hart. *Pattern classification and scene analysis*. Wiley-Interscience, 1973.

[12] Y. Bar-Shalom and X.-R. Li. *Estimation and tracking: Principles, techniques, and software*. Artech House, Boston, 1993.

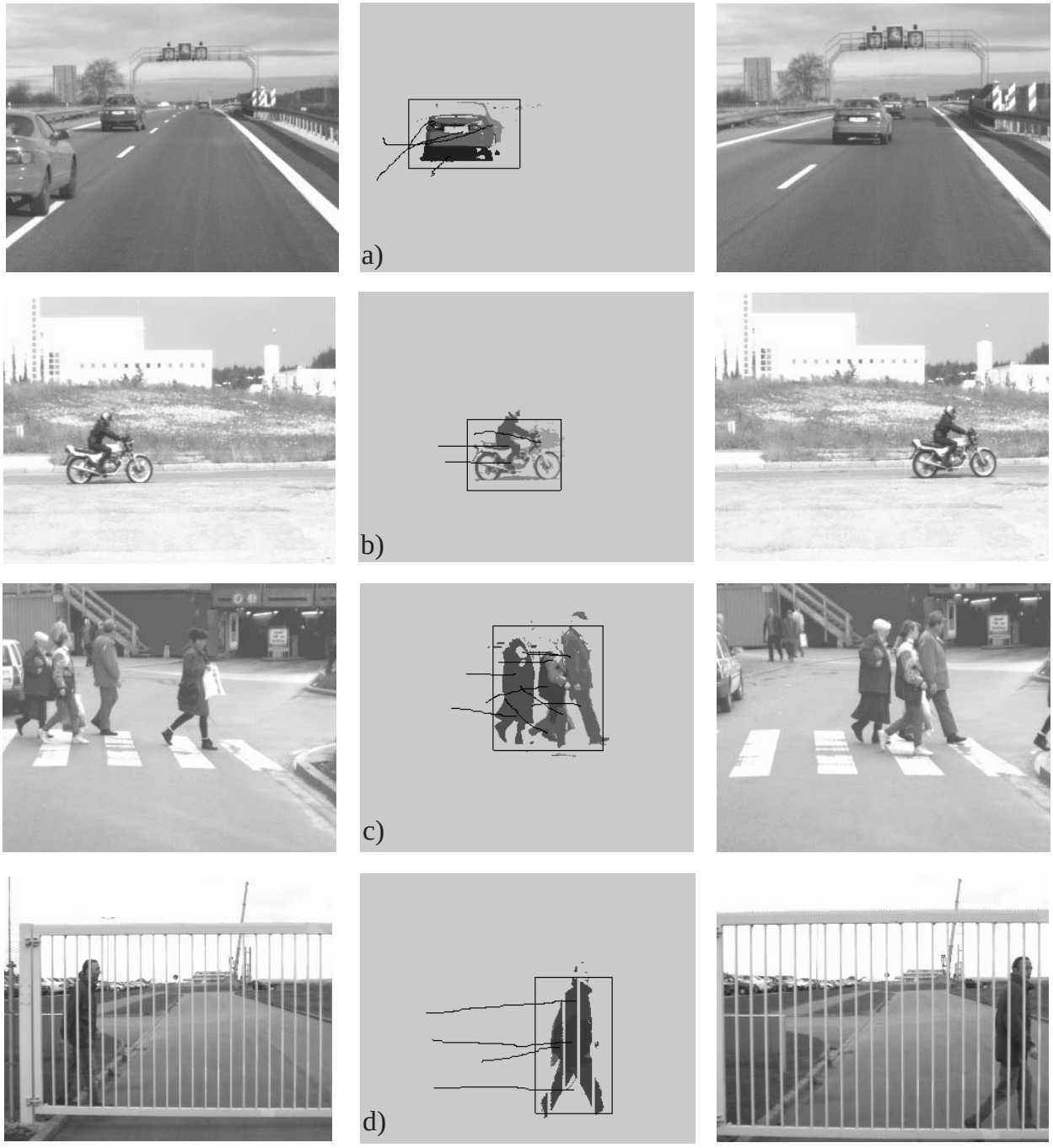

Fig. 5: The left and right images in each row are from the beginning and end of the original sequences. The images in the second column show the detected objects. Dark lines represent the bounding boxes of the object hypotheses and the trajectories of the clusters.