Real-Time Tracking of Pedestrians and Vehicles

N.T. Siebel and S.J. Maybank*
Computational Vision Group
Department of Computer Science
The University of Reading
Reading RG6 6AY, England

Abstract

We present results from a tracker which is part of the integrated surveillance system ADVISOR which is designed to operate in real-time in a distributed local network of off-the-shelf computers.

For PETS2001, our indoor people tracker has been modified to include the tracking of vehicles in outdoor scenes. An effort has been made only to use simple techniques in these modifications. Solutions include splitting and merging of regions which have been processed by the motion detector, as well as the temporal incorporation of static objects into the background image.

Figure 1: View from surveillance camera

1. Introduction

Real-time automated visual surveillance is a popular area for research and development. Recently, the Reading Computational Vision Group has teamed up with research groups from King's College London, INRIA Sophia Antipolis (France), Thales Research (UK), Bull (France) and Vigitec (Belgium) to build the ADVISOR[1] system. ADVISOR is an integrated system for automated surveillance of people in underground stations. People are tracked in real time and their behaviour is analysed. Video annotations and warnings will be archived together with the digitised video and also displayed in real-time to the human operator as necessary. Figure 2 shows the overall system layout for ADVISOR.

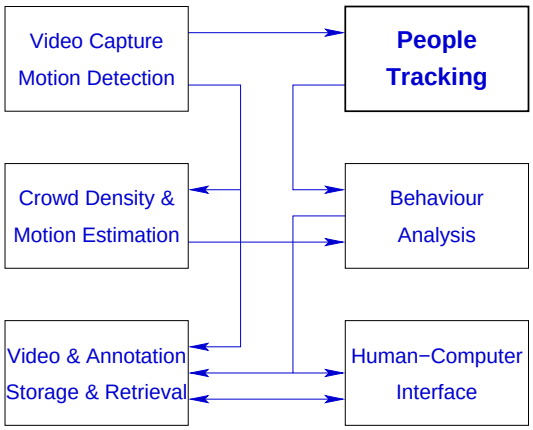

Figure 2: ADVISOR System Layout

It is the task of our group at Reading to provide the people tracking module for the integrated system. The tracker used by us is an extension of the Leeds People Tracker which was developed by Baumberg and Hogg [1]. Starting in 2000, it has been modified for use in the ADVISOR system. In the work reported in this paper the ADVISOR People Tracker is combined with a blob-based vehicle tracker, to allow for simultaneous tracking of people and vehicles in a single camera image. Vehicle tracking will not be needed for ADVISOR, it has been included only to meet the requirements of the PETS2001 data.

*This work is funded by the European Union, grant ADVISOR (IST-1999-11287)

[1]Annotated Digital Video for Intelligent Surveillance and Optimised Retrieval

1.1. People Tracking

The people tracker uses an active shape model [1] for the contour of a person in the image. A space of suitable models is learnt in a training stage using a set of video images containing walking pedestrians. Detected person outline shapes are represented using cubic B-splines. Each outline is specified by a point in a high-dimensional parameter space. Principal Component Analysis (PCA) is applied to the obtained set of points to generate a lower dimensional subspace S which explains the most significant modes of shape variation, and which is a suitable state space for the tracker.

People tracking is performed in multiple stages. The tracker maintains a background image which is automatically updated by median-filtering the sequence of video images over time. To detect new people, a motion detector subtracts the background image from the current video image. Thresholding of this difference image yields a binary image containing "foreground" regions where movement was detected. Those regions that match certain criteria for size and shape are classified as possible people, and are examined more closely by the people tracker. Their outline shape is approximated by a cubic B-spline and projected into the PCA space S of trained pedestrian outlines. The new shape obtained in this process is then used as a starting point for further shape analysis. Once people are recognised they are tracked using the trained shape model. Tracking is performed using Kalman filtering with second order models for the movement of people in 3D. The state of the tracker includes the current outline shape as a point in S. This point is updated as the observed outline changes during tracking.

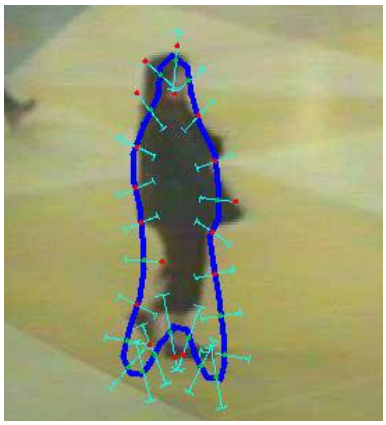

Figure 3: Edge search for shape fitting

In order to adjust the outline shape to each new image of a person we use an iterative optimisation method. The current estimate of the shape is projected onto the image. The shape is then fitted to the image in an optimisation loop by searching for edges of the person's outline in the neighbourhood of each spline control point around the shape. Figure 3 illustrates this process. The pale blue lines show the Mahalanobis optimal search direction [2] used in local edge search.

In previous work we have examined ways to improve local edge contrast in the presence of image noise coming from JPEG compression and electrical interference in the metro station [3]. With these improvements, we found that the shape fitting process is fairly robust provided the person is sufficiently isolated from other people. Tracking was successful even when part of the outline is occluded by static objects in the scene, or affected by image noise.

2. Modifications to the Tracker

We have made a number of modifications to the original tracker in order to use it within the ADVISOR system. Further modifications were necessary for PETS2001 since our tracker is specialised to track people in indoor scenes, and the PETS2001 data feature vehicles and bikes, moving outdoors.

2.1. Modifications Required by ADVISOR

In order to use the existing Reading People Tracking Software for the ADVISOR project, major modifications had to be made to the functionality and software implementation.

The original code made heavy use of **sgi**™'s specialised video hardware to achieve real-time tracking performance. For ADVISOR, the tracking software had to be ported to off-the-shelf PC hardware, in fact a GNU/Linux system, in order to make economic system integration feasible. A number of source-level optimisations and restructuring of the code made it possible to have a similar performance on the GNU/Linux system.

Integrating the system with the ADVISOR network (see Figure 2) meant we had to modify the software to accept from a local network video images in JPEG format and motion data in XML format. Also, video annotations are now written out in XML so the behaviour analysis module can use our output in a standardised format. All XML channels are well-defined using XML Schemas.

Further work included research into ways to increase the robustness of tracking in the presence of the strong image noise and using JPEG image compression. In [3] we have shown how image filtering methods can enhance local edge contrast, thereby improving the robustness of image analysis algorithms for people tracking.

2.2. Modifications for PETS2001

In order to run the tracker on the PETS2001 dataset, we had to modify the people tracker to track vehicles. To keep simplicity and to maintain real-time performance, we decided not to make use of the Reading Vehicle Tracker (RVT) which has been integrated with the people tracker in the

past [4]. Instead, we decided to use a simple frame-to-frame region tracker using output from out motion detector, with a few added extras.

One difficulty is that our motion detector is very simple. It is designed for indoor surveillance and thus it does not handle lighting changes very well. There has been a lot of research on methods to detect image motion more reliably in outdoor scenes. Many surveillance systems nowadays use complex statistical models for background modelling, based on a mixture of Gaussian distributions for each pixel pixel [5] or using a non-parametric model [6]. However, these methods need a considerable amount of CPU time, which in our case is already taken up by the people tracker.

The new region tracking algorithms implemented in our system are explained in more detail in section 3.

3. Tracking Objects Other than People

The tracker uses a simple frame-to-frame region tracker to track moving blobs which do not match the size of people, and to which no person shape could be fitted. In order to deal with overlapping blobs in the motion image, a few new features have been implemented in order not to lose tracks or confuse tracked objects in these situations. The development has been carried out using the "Training" sequence from the PETS2001 dataset 1, camera view 1. We have used 1 out of 7 images from the dataset, resulting in $\frac{25}{7}$ (roughly 3.6) processed frames per one second of video footage.

3.1. Temporal Background Integration

One problem when tracking regions in the image is the overlap of two (or more) blobs. If one of the blobs is still moving and the other one has become static only a few frames ago the motion detection image will normally still show both blobs as moving. Since the blobs overlap in the image, they are detected as one motion blob which makes it difficult to maintain the correct identification of both blobs. This problem can be reduced by maintaining an up to date background image. If one of the objects has become static it can be integrated into the background, thereby enabling correct detection and identification of the second object using the motion image.

Our background image is updated periodically, using a temporal median filter. This means all static objects are eventually incorporated into the background, making detection of other objects in the vicinity possible. However, if we incorporate detected, static objects too quickly into the background, we might not be able to identify and track them when they start moving again. Moreover, we might detect a "negative" of the object (for instance, the absence of a vehicle) once the object has moved on.

A simple procedure has been devised and implemented to incorporate static objects temporarily into the background, thereby

- resulting in correct detection of movement in the vicinity of objects which became static only a few frames ago (people getting out of vehicles etc)
- making it possible to restore the "empty" original background when the object starts moving again
- enabling us to re-gain track and identification of the object since we keep a copy of the necessary data (position, size, identity record).

Figure 4: Car 1 (yellow) integrated into background

Figure 4 shows frame 1134, when the blue VW Polo (left) comes very close to the green Peugeot (right) only a few frames after the latter becomes static. In this situation, the Peugeot is identified as being static and incorporated into the background. This is marked in the image by drawing the yellow box around the vehicle with a dashed line. As a result, the motion image clearly shows the Polo which is identified and tracked successfully. In frame 1190, the motion detector detects motion near the known static Peugeot, correctly assumes this movement is due to the Peugeot moving again, removes the image of the vehicle from the background and keeps on tracking the vehicle with the original identity.

One problem remains the situation where neither of the objects is static, as in frame 1393 when the Peugeot drives past the Polo while the latter is still reversing into the parking lot (see the motion image in Figure 5). Another difficult situation occurs when two parts of the same object, separated in the motion image because of occlusion, are extracted as two separate regions and therefore wrongly identified, classified or matched. An example is frame 1260, where the back of the Peugeot is partly occluded by the

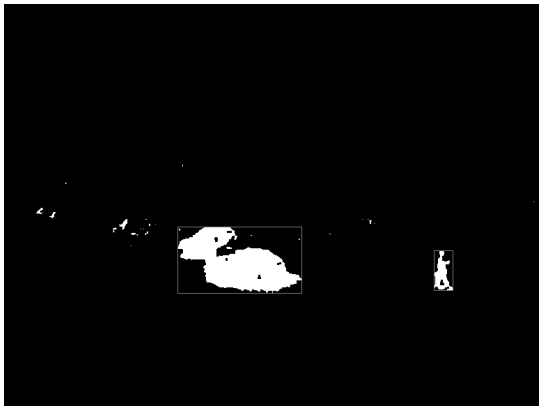

Figure 5: Problem with two close objects, both moving

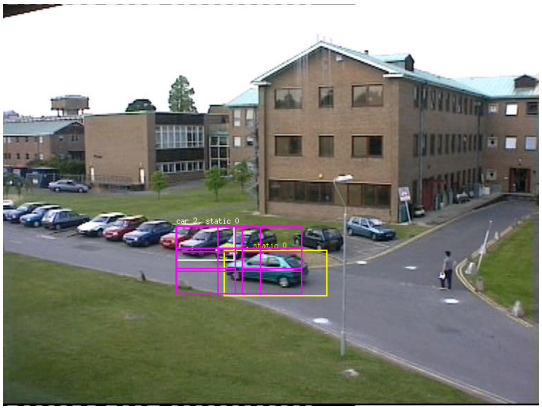

Figure 6: Region Splitting with Multiple Guesses

lamppost, resulting in a splitting of its motion blob as the vehicle reverses out of the parking lot. These problems will be addressed section 3.2.

3.2. Merging and Splitting Regions

To solve the problems due to overlapping or split up blobs in the motion image we use a simple region splitting and merging algorithm. Depending on the size of the region (more precisely, the size of the bounding box of the motion blob detected in the motion image), we

- split large motion regions into 4 vehicle-sized regions, aligned with the upper left, upper right, lower right and lower left corner of the large bounding box, respectively.
- merged nearby regions where at least one of them does not match the minimum size of a vehicle, creating one large region which contains the merged regions.

The rationale behind the splitting is that the two blobs with a larger common bounding box are likely to have their bounding boxes in opposite corners of the common bounding box. Merging was inspired by the problem of partial occlusion by the lamppost in frame 1260, mentioned in section 3.1 above.

The new regions created with these algorithms are added to the list of moving regions detected in the current frame, however, they are clearly marked as being “synthetic”, ie not directly extracted from the motion image. After the standard region identification based on size and predicted new position of previously tracked objects, the synthetic regions which were not matched are assumed to be erroneous and removed.

Figure 6 shows frame 1386 where a large region (the biggest magenta coloured rectangle) is split into a number smaller ones, guessing possible positions of vehicles. Overlayed are the previous detected positions of the two moving vehicles, and guesses (also in magenta) of possible vehicle configurations. In this situation, those guesses which match most closely our prediction about the size and location of the vehicles in this frame are accepted. We keep tracking the two moving regions with correct identifications as they part, as can be seen in Figure 7 (frame 1421).

3.3. Overall Performance During Training

With the additions to the tracker described in this section, we were able to track all people and all vehicles within the “Training” sequence reliably. There were no lost or missed tracks and the position and identity of all people and objects was correctly established at each point in time.

In section 4, we will assess the performance of the tracker when run on the “Testing” sequence.

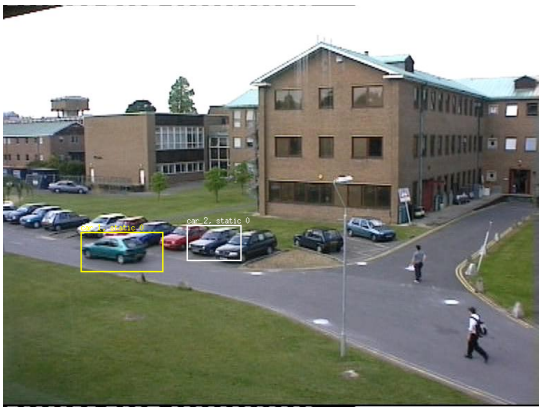

Figure 7: Recovered Tracks after the Objects Part

4. Performance Evaluation

In this section we validate tracking performance again using the PETS2001 dataset 1, camera view 1, this time with the "Testing" sequence. As for the training procedure, we have used 1 out of 7 images from the video footage, resulting in $\frac{25}{7}$ (roughly 3.6) processed frames per one second of imagery.

All people tracking is performed using the active shape tracker described in section 1.1. Moving regions which cannot be identified as people are classified according to size and then tracked using the frame-to-frame region tracker introduced in section 3.

4.1. Tracking Results

Figures 8 and 9 show key frames in the sequence. We will give short explanations here and present the conclusions from tracking in section 4.2 below.
Images in Figure 8 (from top to bottom):

Frame 562 The tracker has tracked the green Peugeot and the pedestrian correctly so far, even in this situation where a significant part of the person's outline is occluded by the vehicle.

Frame 604 The occlusion of the pedestrian was not modelled by the simple blob tracker. As a consequence, a large part of the person outline was not detectable although the people tracker expected it to be so we lost track. When we re-gain track of the person a few frames later we do not recognise this is the same person as before.

Frame 863 The white van has driven past the parked Peugeot. The Peugeot had been detected as static and temporarily incorporated into the background. We therefore track the van correctly although it gets very close to the Peugeot. The person at the left hand side is detected and tracked, however, the group of people at the right is not classified correctly as the people tracker cannot easily extract single silhouettes within the group.

Frame 933 The tracker did not predict the trajectories of the group of people and the white van correctly when they overlapped and fused in the motion image. Although the region tracker breaks up the large region into two smaller regions it does not establish their true identities but instead mixes up the tracks. The person in the front is accurately tracked, and the driver getting out of the parked Peugeot is detected at an early stage.

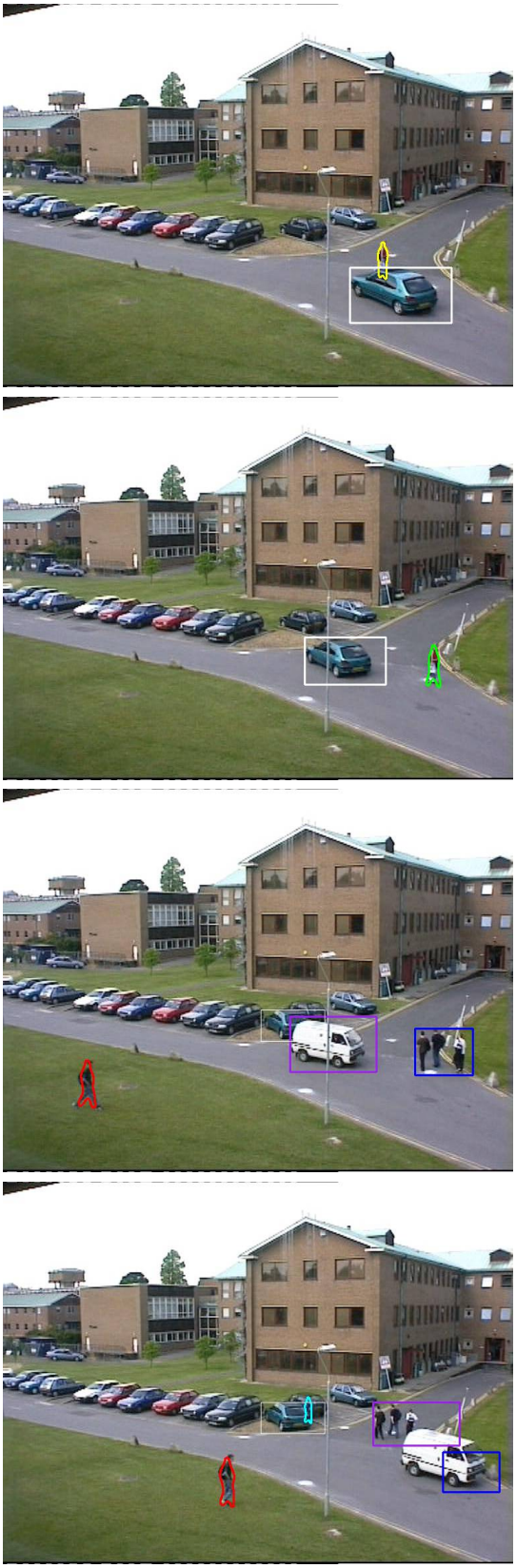

Figure 8: Frames **562**, **604**, **863** and **933**

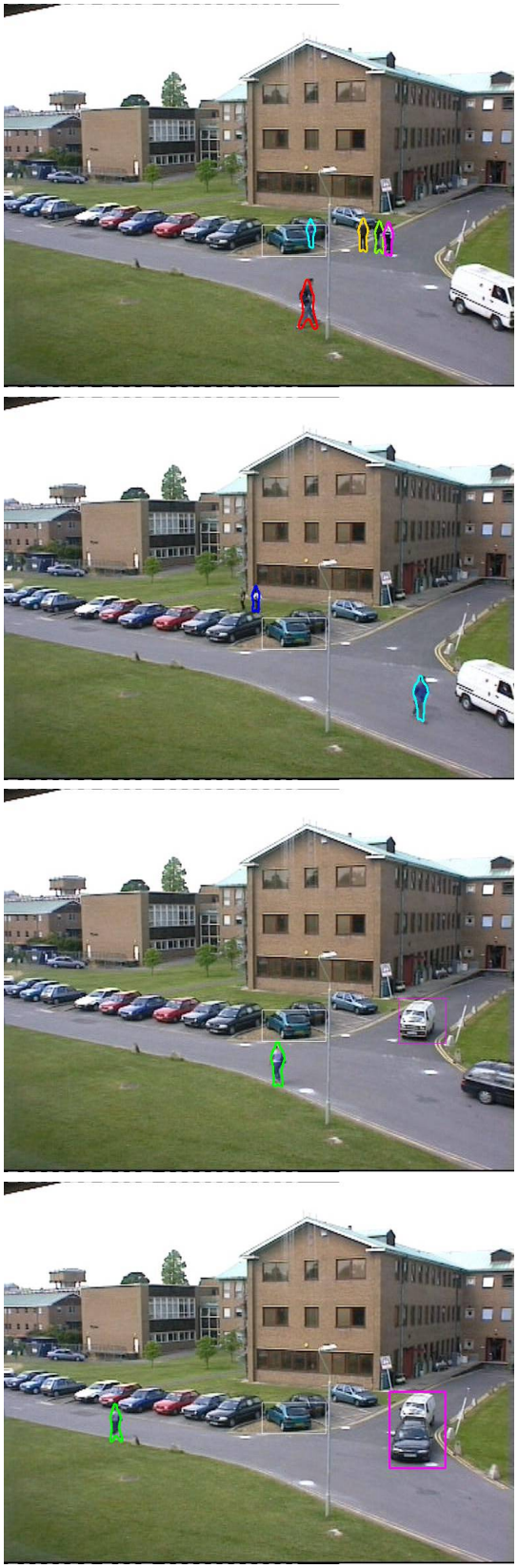

Figure 9: Frames **975**, **1213**, **2193** and **2382**

Images in Figure 9 (from top to bottom):

Frame 975 All 5 people in the image have been precisely detected and they are tracked for a few frames. However, a little later in the sequence the 3 individuals at the right hand side will merge and build a close group again which means that we cannot keep track of the individuals making up the group.

Frame 1213 We have kept track of the pedestrian in the front until they have left the image. The driver of the Peugeot is leaving the field of view quickly towards the right, tracked all the time. Until the group of people leaves the visible area, we detect people within the group when they are sufficiently isolated from each other. However, these tracks are not reliable and are not kept over an extended period of time.

Frame 2193 A pedestrian has entered the field of view from the right and is tracked correctly until they leave. The white van has reversed back into the image and is detected as static. However, reflections on the windscreen are not modelled and therefore show up in the motion image. These motion blobs are filtered out as noise because they do not match the size of the van.

Frame 2382 At this point in time, the lighting has started to change, resulting in noisy measurements from the motion detector. While the people tracker is not affected strongly yet, the motion image-based region tracker shows up erroneous measurements. Most of these are filtered out as noise because they are too small or they have not enough temporal consistency to be accepted as tracks. The reliability of tracks, however, is much reduced. An estate car has come close to the parked van, at the right hand side of the image. Their motion blobs merge and the region tracker cannot distinguish the two vehicles any more.

4.2. Overall Performance During Testing

Vehicle Tracking: Although our new region tracker did very well in the "Training" sequence, the "Testing" sequence posed problems which it had not been trained for. In particular, there is a strong lighting variation towards the end of the sequence, causing severe problems for the motion detector. This effect is demonstrated in figure 10 which shows the motion image in frame 2617, one of the last frames in the sequence. The reason for this problem is the motion detector's background updating process which does not adapt fast enough to the lighting variation. As a consequence, our new region tracker is much less reliable as it relies on correct output from the motion detector.

In conclusion, we have established that our motion detector in its current state is not suitable for use in an outdoor surveillance application.

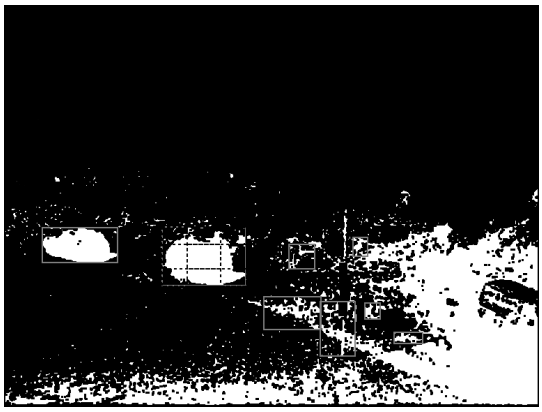

Figure 10: Motion Image During Strong Lighting Variation

People Tracking: The People Tracker has been shown to work reliably through the whole sequence, as long as people are sufficiently isolated from other people. Groups of people cannot be handled well when people are so close that they significantly occlude each other's outlines in the image.

Speed: The tracker runs in real time, more specifically at approximately 5 frames per second, when run on full PAL-sized images. Considering we have used only approximately 3.6 images per second of video we can say real-time performance has been more than established. The computing time includes software JPEG decompression and annotation output in XML, but no display (this is done by the Human Computer Interface in the ADVISOR system). The computer uses an 800 MHz Pentium III^{TM} CPU, running under GNU/Linux.

5. Discussion

In this paper we have presented modifications to the Reading People Tracking System in order to implement simple tracking of vehicles and other objects not dealt with by the original people tracker. Emphasis has been put on keeping the new algorithms simple and efficient. We have implemented a simple frame-to-frame region matching algorithm, including region merging and splitting according to rules based on the size of regions extracted from the motion image. Multiple guesses as to the position and size of vehicles are matched to predictions from the previous frame, aiming at recovery from problems where a single motion region contains more than one, or only part of a vehicle. All tracking can be performed in real time on a standard off-the-shelf computer.

5.1. Performance

The validation of the tracker shows that although the new *Vehicle Tracking* routines deal well with simple circumstances, they lack the ability to deal with multiple moving regions when the regions overlap and fuse. A problem here is the heavy reliance of the region tracking on a correct motion image. In the "Testing" sequence from dataset 1 the lighting variation towards the end happens so fast that our motion detector cannot recover, resulting in lost and false tracks.

Our *People Tracking* generally gives good results, and the new algorithms to incorporate regions (like vehicles) into the background when they have been static for only a few frames improves tracking of people walking past or getting out of vehicles which have just been parked. Groups of people still pose a problem for the people tracker, because of its design to use the outline of a single person.

5.2. Future Work

We are currently working on ways to improve tracking of groups of people, within the ADVISOR project. Ideas include the use of lower-level image cues from simple human feature detection, and new statistical models for shape fitting.

Another area of research is the identification of people when we have lost track of them and re-gain track at a later time. Appearance models based on colour and possibly more information have been established and will be implemented.

The validation also shows that if we are to run our tracking system within an outdoor surveillance application, the use of a better background modelling algorithm will be required. One aspect which we have to keep in mind is the computational cost usually involved with more complex background modelling. The CPU usage of such a motion detector must not be too high, in order to keep system integration costs economical.

References

[1] A. M. Baumberg, *Learning Deformable Models for Tracking Human Motion*. PhD thesis, School of Computer Studies, University of Leeds, October 1995.

[2] A. Baumberg, "Hierarchical shape fitting using an iterated linear filter," in *Proceedings of the Seventh British Machine Vision Conference (BMVC96)*, pp. 313–322, BMVA Press, 1996.

[3] N. T. Siebel and S. Maybank, "The application of colour filtering to real-time person tracking," in *Proceedings of the 2nd European Work-*

shop on Advanced Video-Based Surveillance Systems (AVBS'2001), pp. 227–234, September 2001.

[4] P. Remagnino, A. Baumberg, T. Grove, T. Tan, D. Hogg, K. Baker, and A. Worrall, "An integrated traffic and pedestrian model-based vision system," in *Proceedings of the Eighth British Machine Vision Conference (BMVC97)* (A. Clark, ed.), pp. 380–389, BMVA Press, 1997.

[5] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'99)*, vol. 2, pp. 246–252, 1999.

[6] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *ECCV 2000, 6th European Conference on Computer Vision* (D. Vernon, ed.), pp. 751–767, Springer Verlag, 2000.