Matthias Zobel, Joachim Denzler, Heinrich Niemann

Binocular 3-D Object Tracking with Varying Focal Lengths

appeared in:
Proceedings of the IASTED International Conference on Signal Processing,
Pattern Recognition, and Application, (SPPRA 2002)
Crete, Greece
p. 325–330
2002

Binocular 3-D Object Tracking with Varying Focal Lengths

Matthias Zobel,[*] Joachim Denzler, Heinrich Niemann
Chair for Pattern Recognition, Universität Erlangen-Nürnberg
Martensstraße 3, 91058 Erlangen, Germany
{zobel,denzler,niemann}@informatik.uni-erlangen.de

ABSTRACT

In this paper we discuss some practical facets of active camera control during 3-D object tracking. The basis of our investigation is an active binocular camera system looking onto a scene to track a moving object in 3-D. Tracking is done in a data driven manner using an extented version of the region based method that was proposed earlier by Hager and Belhumeur. Triangulation of the extracted objects in the two image planes lead to an estimation of the 3-D position of the moving object. To keep the moving object in the center of the image the tilt and vergence axis of the binocular camera system is controlled. The important question for such an experimental setup is which parameters do influence the quality of the final 3-D estimation. The main effects we are concentrating on is the accuracy of the 2-D localization in the image plane depending on the focal length of the camera. Also the consequences of errors in the synchronization between image acquisition and motor control of the camera system are shortly discussed. All considerations are verified in real-time experiments.

KEY WORDS

Real-time 3-D Object Tracking, Image Sequence Processing, Zooming

1 Introduction

Active control of a camera system during object tracking has been proposed earlier [3, 8]. The main motivation is to keep the moving object in the image. Recently, also the control mechanism for moving the camera has been discussed [9]. In their control theoretic investigation the authors present a two mode controller, one performing smooth pursuit, while in the second mode saccades are performed to keep the object of interest in the image.

The goal of our research is object tracking in 3-D by optimally selecting the focal lengths of a binocular camera system. For focal length selection we have to deal with the tradeoff between a small focal length that reduces the induced image flow and a large focal length that might increase accuracy in the 2-D localization in the image plane and therefore later in the 3-D estimation of the position and velocity of the moving object.

In consideration of this goal, i.e. to improve tracking of a moving object in 3-D, several factors must be taken into account that influence the quality and accuracy of the 3-D position estimation. First, the tracking algorithm must handle changing focal lengths during tracking without losing the object. Second, the real-time constraints that we inherently have to deal with if we change camera parameters during tracking enforce tight constraints on the synchronization between the acquisition of the image data on the one side and the motor positions of the binocular camera system on the other side. Up to now, no system is known to the authors that has perfect synchronization between the controller of the camera system and the framegrabber. This is not only a problem in practice: a strategy must be found that reduces errors induced by a misalignment of the camera parameters with the image data if synchronization is difficult to achieve or even impossible.

In this paper we mainly tackle the first of the problems, namely the question, how a certain tracking algorithm behaves if the focal length is adapted during tracking. The tracking algorithm we use is a region based method presented in [5]. The advantage of this algorithm is that tracking can be done in a data driven manner without having a 3-D or 2-D model of the object. The estimation of the position, the scaling, and the rotation of the object is done during an optimization step. We shortly summarize the region based tracking method of [5] in Section 2. Additionally, we present a hierarchical extension of this method in order to improve the handling of faster object motions.

In Section 3 we present our binocular vision system, and we discuss how 3-D information about the moving object can be acquired with this setup using triangulation. Also, the problems to be expected are mentioned.

Real-time experiments are presented together with quantitative results in Section 4 to investigate how the accuracy of the 3-D position estimation depends on the chosen focal length, assuming the 2-D positions in the images being computed by the region based tracking approach.

Based on the gained results, we finally give a suggestion in Section 5 on how the focal lengths of the binocular system should be controlled using the presented tracking algorithm.

[*] This work was supported by the “Deutsche Forschungsgemeinschaft” under grant SFB603/TP B2. Only the authors are responsible for the content.

2 Binocular Object Tracking

If at some discrete time t a region in an image can be defined that contains the projection of a real-world object that moves through the field of view of the camera, the problem of 2-D object tracking is to find the corresponding region in the image taken at time $t+1$. Due to motion of the object, changes in illumination, occlusions, or changes in the internal and external camera parameters, the region at time $t+1$ will be a somehow transformed version of the region of the previous timestep.

Hager and Belhumeur have presented an algorithm for efficiently tracking an object's image region through a sequence of images [5]. They have provided impressive results on tracking a human face even on large changes in illumination and partial occlusions. For the investigations in this contribution we consider the case that the variability in the images is caused only by object motion or changes in the projection parameters.

2.1 Region Based Tracking

In the framework of Hager and Belhumeur, object tracking is formulated as a parameter estimation problem according to the minimization of the sum-of-squared differences between two image regions at subsequent timesteps.

The region of the tracked object, the *target region*, is defined by a set

$$R = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\} \tag{1}$$

of N image locations $\boldsymbol{x} = (x, y)^{\mathrm{T}}$. The set of the intensity values $f(\boldsymbol{x}_i, t_0), i = 1, \ldots, N$ of these locations at an initial time t_0 is referred to as the *reference template*.

During tracking the target region transforms due to the reasons mentioned above. The transformation can be modeled by a parametric *motion model* $\boldsymbol{\xi}(\boldsymbol{x}; \boldsymbol{\mu})$ with parameter $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_n)^{\mathrm{T}}, n < N$, obeying

$$\boldsymbol{\xi}(\boldsymbol{x}; \boldsymbol{\mu} = \boldsymbol{0}) = \boldsymbol{x} \quad .$$

$\boldsymbol{\mu}$ is referred to as the time dependent *motion parameter vector*; $\boldsymbol{\mu}^*(t)$ denotes the true values of the motion parameters at time t and $\boldsymbol{\mu}(t)$ the estimated ones.

Under the assumption of *image constancy*, i.e. for any time $t > t_0$ there exists a motion parameter vector $\boldsymbol{\mu}^*(t)$ such that

$$f(\boldsymbol{x}, t_0) = f(\boldsymbol{\xi}(\boldsymbol{x}; \boldsymbol{\mu}^*(t)), t) \quad \forall \boldsymbol{x} \in R \quad ,$$

we gain an estimate for the motion parameter vector at time t from solving

$$\min_{\boldsymbol{\mu}(t)} \left(\sum_{\boldsymbol{x} \in R} \left(f(\boldsymbol{\xi}(\boldsymbol{x}; \boldsymbol{\mu}(t)), t) - f(\boldsymbol{x}, t_0) \right)^2 \right) \quad .$$

This optimization can be efficiently solved at every timestep by the recursive algorithm that Hager and Belhumeur have developed in their article. As examples they consider different kinds of motion models starting with a linear one that models pure translation up to a more complicated case of a special family of nonlinear motions. In our experiments we have mainly used the so called RM+S model, i.e. planar rigid motion plus scaling

$$\boldsymbol{\xi}(\boldsymbol{x}; \boldsymbol{\mu}) = s\boldsymbol{R}(\theta)\boldsymbol{x} + \boldsymbol{u} \tag{2}$$

with a 2×2 rotation matrix $\boldsymbol{R}$ and motion parameter vector $\boldsymbol{\mu} = (\boldsymbol{u}, \theta, s)^{\mathrm{T}}$ being $\boldsymbol{u}$ a translation vector, θ a rotation angle, and s a scaling factor.

It should be noticed, that if a movable camera is used, for example a pan/tilt camera, the estimated motion parameter vector can be further used to derive appropriate motion commands for the camera's axes to keep the object in the middle of the image, i.e. to fixate onto the object.

2.2 Tracking Fast Motions

One drawback of the proposed method is, as the authors pointed out by themselves, that only small image motions of about a few pixels can be handled without losing the object. This means that tracking with a certain focal length restricts the speed of the object's motions to a corresponding maximal value. One possibility to overcome this situation is to decrease the focal length of the tracking camera, i.e. to zoom out of the scene, to reduce the image flow that is induced by the object's motions. As already pointed out in the introduction and also experimentally investigated in Section 4 this may result in an increasing instability in 2-D tracking and therefore in 3-D position estimation, because in general the motion parameter vector estimation will become more difficult due to a smaller projection of the object in the image.

Another approach to handle fast object motions that we have implemented — and that works independent of the ability to vary the focal length — is to perform a hierarchical estimation of the motion parameter vector at different levels of resolutions from coarse to fine. The same motion of an object induces less image motion as the resolution of the image decreases.

In the original non-hierarchical case, the estimated motion parameter vector $\boldsymbol{\mu}(t-1)$ from time $t-1$ is used recursively as the initial value for the estimation at time t (cf. Figure 1, top).

In the hierarchical case, as it is depicted in Figure 1 (bottom), every one of the k levels of the resolution hierarchy actually runs its own tracker referring to its own scaled version of the reference template. At time t we start the estimation process on the highest level, initialized with the solution ${}^{0}\boldsymbol{\mu}(t-1)$ of level 0 from the last timestep. This results in an estimate ${}^{k-1}\boldsymbol{\mu}(t)$ on level $k-1$ that is propagated down the hierarchy to the next level $k-2$ and so on until level 0 is reached.

While the estimated parameter vectors become propagated down the levels they must of course be adapted appropriately to the new level. In our case, for example, if the

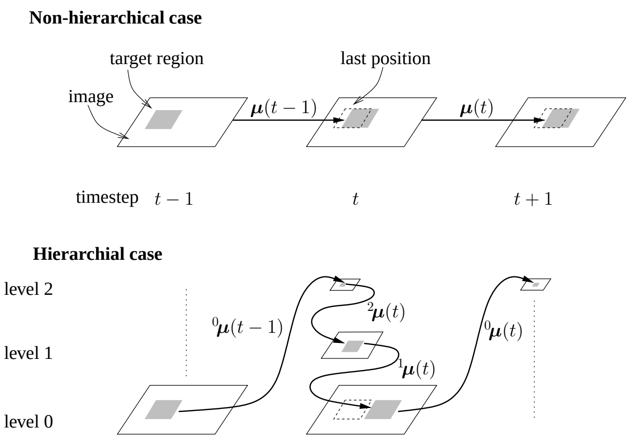


Figure 1. Propagation of the estimated motion parameter vector $\boldsymbol{\mu}$ over time. In the non-hierarchical case (top) only small image motions can be tracked properly. In the hierarchical case (bottom) larger motions can be handled. Here, it is depicted for a hierarchy with three levels.

resolution is doubled from one level down to the next and using the RM+S motion model the translational component $\boldsymbol{u}$ has to be multiplied by the factor two. The scaling factor s and the rotation angle θ need not to be modified, because these parameters are invariant to image scaling. Going the other direction from level 0 to the highest level $k-1$ at the beginning of each timestep, we have to divide $\boldsymbol{u}$ by 2^{k-1}.

By the use of such a hierarchical processing the range of the possible image motion is doubled with each new level that is introduced in the image pyramid. On the other hand one must be aware of the fact, that computation time and memory usage is increased, too. This may be a critical point for real-time applications. For our experiments we gained best results for $k = 2$ and $k = 3$ levels.

3 Gaining 3-D Information

The hierarchical extended framework of Hager and Belhumeur that was described in the last section provides only pure 2-D tracking in the image plane. No information can be gained about the 3-D position of the object in the world using just one single camera. This looks different if we are using two cameras.

In this section we shortly summarize facts of the two-camera vision system that we have used for our experiments. Afterwards, we describe the technique that was applied for deriving 3-D information from the two 2-D trackers, and finally, we address some aspects inherent to the currently used tracking system that influence the accuracy of the 3-D position estimation.

3.1 Binocular Setup

The binocular vision system that has been utilized for our experiments is a TRC bisight/unsight unit (”TRC head” for short) that can be seen in Figure 2. It has four axes to control pan, tilt, and left and right vergence. For each of the lenses the zoom, focus, and iris settings are adjustable. This results in a total of 10 degrees of freedom. In our experiments five out of these 10 degrees of freedom were used. Pan, focus, and iris remain at constant settings. The motors of the axes and cameras can be accessed via a PMAC servo controller that is connected to a workstation using a RS232 serial interface. Except for the iris all other motors work in a closed loop, so it is possible to query the current position of the axes and camera parameters.

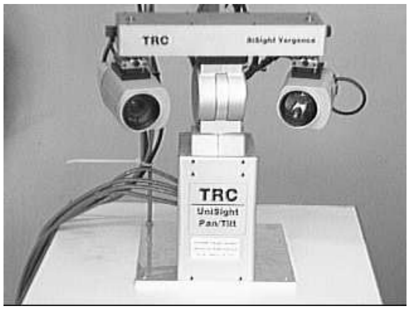


Figure 2. The TRC bisight/unisight binocular camera system.

There exists a coordinate system for each of the two cameras, whereby it is assumed that the origins of the coordinate systems coincide with the intersection of the tilt axis and the corresponding vergence axis (cf. [1]). It is also assumed that if both cameras were aligned to build a perfect stereo system their x-axes coincide with the mechanical tilt axis of the TRC head.

3.2 3-D from Triangulation

As already pointed out, one camera is not enough to gain information about the 3-D position of the tracked object. Therefore, we used the two cameras of the previously described TRC head to track the object with each of them separately. While tracking we use a triangulation method (cf. [6] on different techniques) for computing the 3-D position from the two 2-D target region position estimates. This is done by finding the intersection point of the two lines of sight corresponding to the 2-D image coordinates of the center of the target region.

Because in general these lines do not intersect in a common point, we have used the *midpoint* method as an approximation. It is searched for those points on both lines of sight at which the distance between the lines takes its minimal value. The midpoint of the line that connects these two points is considered to be the intersection point. The coordinates of this intersection point are treated as the 3-D position of the tracked object with respect to a given world coordinate system.

To perform triangulation it is required to know the properties of the imaging process in each camera, i.e. the cameras need to be calibrated with respect to a common world coordinate system. We use the method of Tsai [10] to calibrate the internal and external parameters of the cameras of the TRC head by means of a coplanar calibration pattern. The origin of the world coordinate system is defined to be one of the points of the calibration pattern.

Since it is our goal to vary zoom while tracking, for proper triangulation those camera parameters need to be used that correspond to the current focal length of the camera. It is practically not feasible to calibrate the cameras for each possible focal length. Therefore, we have calibrated the cameras at different equidistant zoom motor positions and stored the resulting parameter values. During tracking we use those stored camera parameters that are closest to the current zoom motor position. The error that is produced by this selection scheme can be reduced if the distance between two calibration positions is shortened. Another possibility would be to interpolate the camera parameters for the current focal length from the two neighboring calibration data records.

3.3 Critical Aspects

In addition to the errors in 3-D position estimation that come from deviations in the calibration data, there are two main factors that influence the accuracy of the estimation even if using the correct camera parameters.

The first point we want to mention is the correspondence problem between the two cameras and the object. Currently, there is no mechanism that ensures that the centers of both of the target regions correspond to the same 3-D point on the object. In fact, the trackers are initialized manually by clicking with the mouse on the object in the live video streams of the cameras one after the other. Hence, although visually the same object is tracked, the prerequisite for proper triangulation is not given in general resulting in an absolute error in 3-D position estimation. It should be noticed that even if there is an automatic initialization mechanism, the problem still remains.

The correspondence problem still exists even if we use static cameras. But one major problem arises when we try to fixate the object with movable cameras. This means, deviations of the target region's center from the image center need to be compensated by appropriate camera motions. As a consequence, if it takes longer than 40 ms (assuming a framerate of 25 fps) to complete the whole motion, the next image will be taken while the camera still moves. This would cause no problems if you could ensure to query the current position of the camera exactly at the same time as you grab the new frame and therefore providing the triangulation routine the correct values. But to our knowledge there exists no such system that provides this synchronicity. In general, due to delays in communication, especially over the serial interface, the position information received presents an old camera configuration. The same problem occurs if you cannot query the camera's position at the same frequency as frames are grabbed. For example, the TRC head we use can provide position information only every 50 ms.

It shows up that the error that is introduced due to these delays depends on the current speed of the image motion, i.e. if there is less motion to compensate, the error will be small. In contrast, if the cameras need to move fast, the error will be increased.

4 Zooming while Tracking

The long term research goal of our work is the optimal selection and fusion of sensor data of n static and m moving cameras for 3-D object tracking. Currently we restrict our investigations to $n = 2$ spatially fixed active cameras as described earlier. While pan and tilt control has been studied in detail earlier (cf. for example a recent control theoretic paper [9]) as well as the use of active focus for depth estimation [7], the adjustment of focal length for improved tracking has not been considered in detail so far. Literature can be found on active zoom control for depth reconstruction and imaging an object with maximum resolution.

In [4] for a single camera a framework for controlling the focal length is presented to keep an object that is moving along the optical axis at constant size. Such a setting allows to apply scale variant tracking algorithms, like correlation techniques. Here we explicitly like to figure out how a certain algorithm behaves if the focal length is changed during tracking, while we allow — in contrast to the work of [4] — an arbitrary motion of the object in 3-D.

The reason for such an investigation is, that the optimal focal length during tracking not only depends on the distance of the object to the camera but also on the uncertainty in the estimation of the whole state of a moving object. Besides of depth, additionally the state includes velocity and acceleration. A consequence is that it might be necessary to change the focal length already if the uncertainty in the state estimation is changed during tracking. If the uncertainty in the state estimation is large, zooming toward the object is risky since in the next image the object might disappear from the image.

For our experiments we have extended the tracking algorithm of [5], summarized in Section 2. With respect to zooming while tracking the algorithm is perfectly suited since the change in scale of the target is implicitly estimated in the RM+S model (compare Eq. (2)). One of the free parameters of the algorithm that is of special interest for the quality of zooming while tracking is the initial target region, i.e. the size of the reference template.

LC	RC	x	y	z
17.1	17.4	3.48	13.08	6.46
24.0	23.4	1.93	6.91	3.50
31.6	30.8	0.90	3.33	1.75

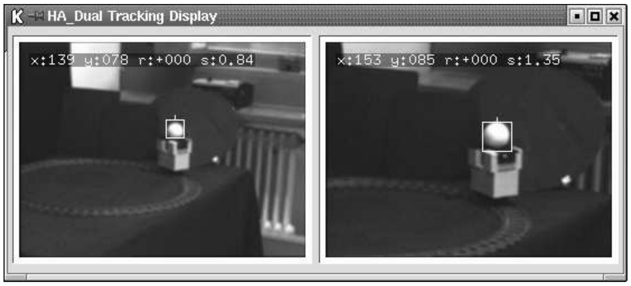
Table 1. Standard deviation in 3-D position estimation (world coordinates) for different focal lengths for the left (LC) and right (RC) camera (in mm). Initialization of the reference template was done with a focal length of 17.1 mm and 17.4 mm for the left and the right camera, respectively.

LC	RC	x	y	z
19.3	19.1	6.38	22.32	11.13
24.0	23.4	3.17	10.50	5.29
31.6	30.8	1.95	7.06	3.59

Table 2. Standard deviation in 3-D position estimate (world coordinates) for different focal lengths for the left (LC) and right (RC) camera (in mm). Initialization of the reference template was done with a focal length of 31.6 mm and 30.8 mm for the left and the right camera, respectively. For the smallest focal length, extraction of the object was not possible any longer.

4.1 3-D Object Localization

First we tested the sensibility in 3-D position estimation using the triangulation technique described in Section 3.2 for a static setting. A static setting means that the object was fixed in 3-D. In both camera images the object was tracked over time while simultaneously varying the camera parameters. Based on the 2-D image coordinates of the extracted target region the 3-D position is computed. The quality in 3-D localization was estimated for varying resolutions of the object in the image (i.e. varying focal lengths) and different sizes of the reference template (i.e. different zoom settings for the initial image, in which the reference template has been selected). The quality in 3-D localization is measured by the standard deviation in the 3-D position estimate after triangulation. Table 1 and Table 2 summarize the results. In both cases, i.e. when zooming in and zooming out with respect to the reference template size, we got the smallest standard deviation for the largest focal length for both cameras. This means, that the algorithm performs best in either case for maximum resolution of the image. It is both interesting and plausible that the most stable results in the 3-D estimation by triangulation are achieved if the reference template has low resolution. It should be mentioned, that due to the pose of the world coordinate system, errors in depth estimation correspond mainly to errors in the y-coordinate.



```
x:139 y:078 r:+000 s:0.84
x:153 y:085 r:+000 s:1.35
```

Figure 3. Screen dump during live experiment. Left and right camera image with focal length being 19.5 mm and 31.4 mm respectively. Tracked object is indicated by white rectangle.

4.2 3-D Object Tracking

The results that we got for a static setting are now verified for an object moving in 3-D on a circular path at a distance of about 3 m in front of the cameras. The 3-D pose of the circle was estimated during the camera calibration step in order to have ground truth information. During the experiments the mean distance to this reference path was computed. A screen dump during one real-time experiment is shown in Figure 3. It shows the camera image of both cameras, the tracked object (white ball) and some information on the estimated position of the moving object, the rotation angle (set to zero in our experiments for stability reasons), and the computed scaling factor. The focal length was 19.5 mm for the left camera and 31.4 mm for the right.

The tracking algorithm runs on an Athlon 1GHz with a framerate of 25 fps, including the image acquisition, the extraction of the position in the image plane for both camera images, the control of the binocular camera system, and the triangulation to return 3-D position estimates for the object.

In Table 3 the results for different combinations of focal lengths (initial focal length to define the reference template and focal length during tracking for both cameras) are shown. Comparable with the results for pure localization of the object in the image plane (compare Section 4.1) the best results are achieved for a small focal length during initialization of the template and a large focal length during tracking.

In Figure 4 one example for tracking the moving object in 3-D is shown. The focal length has been set to 23.7 mm and 23.2 mm for the left and right camera. In the left image of Figure 4 one can see a perfect ellipse confirming that tracking in the image plane was successful. Examining the returned path of the object in 3-D (Figure 4, right) one can see, that there are two main parts on the 3-D path, where the error in depth is large. One should see a circle. The reason for this systematic error that could be observed for nearly every experiment is caused by the error

init	work	min	max	mean	stddev
19.5	19.5	1.8	212.8	71.8	44.9
19.5	23.7	1.8	169.6	65.9	37.7
19.5	32.0	1.2	133.4	54.6	33.7
23.7	19.5	0.8	358.3	64.4	59.8
23.7	23.7	0.9	235.3	59.7	50.2
23.7	32.0	3.7	243.1	95.1	60.4
32.0	19.5	1.2	513.3	84.3	69.2
32.0	23.7	0.4	314.4	63.5	55.6
32.0	32.0	0.4	212.5	68.6	47.2

Table 3. Minimum (min), maximum (max), mean error (mean), and standard deviation for tracking the moving objects using different focal lengths (init: focal length used to define the reference template. work: focal length during tracking for both cameras). All quantities are given in mm.

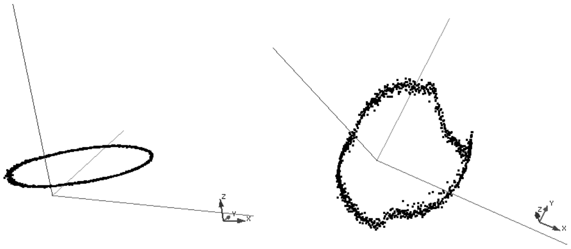


Figure 4. Reconstructed motion path of the tracked object in 3-D based on the 3-D estimation using triangulation. Left: ellipse as seen from the camera. Right: view of the trajectory from above (this should be a circle).

in the synchronization between the image acquisition and the transfer of the camera data (tilt and vergence angles). Similar experiences have been noted by [9] and explicitly modeled for the control of a monocular active camera system.

5 Conclusion

In the paper we have presented results from real-time experiments with a binocular active camera system. The goal has been to report on different aspects that influence the quality of the estimation of the 3-D position of an arbitrarily moving object. The investigations have been concentrated on the selected focal length for the cameras, with special focus on the applied tracking algorithm. Without taking the uncertainty in the position estimate into account we could show that

1. the region based tracking algorithm can handle varying focal length and is thus suited for adaptive focal length control during tracking
2. the smallest expected error in the 3-D arises if the reference template is extracted from a low resolution image taken with small focal length and subsequent tracking should be done using large focal length.

Especially the scond point is the typical scenario when a moving object is detected in a low resolution overview image of the scene and the following tracking is done with large focal length.

One important experience has been collected more or less incidentally. From geometrical considerations it is obvious that the errors in the 3-D estimation that come from wrong assumptions for the rotation angles of the binocular camera system cannot be compensated by selecting a certain focal length (cf. Section 3.3). This fact especially holds for errors in the angles of the vergence axes.

In further research we will investigate the influence of the synchronization delays on the resulting 3-D estimation at different focal lengths in more detail. In our opinion the incorrect synchronization is the main reason for the deformations of the perfect object trajectory as they can be seen in Figure 4, right. Also, we will transfer the framework for optimal sensor data selection presented in [2] for object recognition to the dynamical case of object tracking.

References

[1] K. Daniilidis, C. Krauss, M. Hansen, and G. Sommer. Real-time tracking of moving objects with an active camera. *Real-Time Imaging*, 4:3–20, 1998.

[2] J. Denzler and C.M. Brown. Information theoretic sensor data selection for active object recognitionand state estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2), 2002.

[3] J. Denzler and H. Niemann. Real–time pedestrian tracking in natural scenes. In G. Sommer, K. Daniliidis, and J. Pauli, editors, *CAIP'97*, pages 42–49, Berlin, 1997.

[4] J. Fayman, O. Sudarsky, and E. Rivlin. Zoom tracking. In *Proceedings of International Conference on Robotics and Automation*, pages 2783–2788, Leuven, Belgium, 1998.

[5] G.D. Hager and P.N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(10):1025–1039, 1998.

[6] R. Hartley. Triangulation. *Computer Vision and Image Understanding*, 68(2):146–157, 1997.

[7] E.P. Krotkov. *Active Computer Vision by Cooperative Focus and Stereo*. Springer, Berlin, Heidelberg, 1989.

[8] J.E.W. Mayhew, Y. Zheng, and S.A. Billings. Layered architecture for the control of micro saccadic tracking of a stereo camera head. In D. Hogg and R. Boyle, editors, *British Machine Vision Conference 1992*, London, 1992. Springer.

[9] E. Rivlin and H. Rotstein. Control of a camera for active vision: Foveal vision, smooth tracking and saccade. *International Journal on Computer Vision*, 39(2):81–96, 2000.

[10] R.Y. Tsai. A versatile camera calibration technique for high–accuracy 3d machine vision metrology using off–the–shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323–344, 1987.