

Extending GKLT Tracking—Feature Tracking for Controlled Environments with Integrated Uncertainty Estimation

Michael Trummer¹, Christoph Munkelt², and Joachim Denzler¹

¹ Friedrich-Schiller University of Jena, Chair for Computer Vision
Ernst-Abbe-Platz 2, 07743 Jena, Germany

{michael.trummer, joachim.denzler}@uni-jena.de

² Fraunhofer Society, Optical Systems
Albert-Einstein-Straße 7, 07745 Jena, Germany
christoph.munkelt@iof.fraunhofer.de

Abstract. Guided Kanade-Lucas-Tomasi (GKLT) feature tracking offers a way to perform KLT tracking for rigid scenes using known camera parameters as prior knowledge, but requires manual control of uncertainty. The uncertainty of prior knowledge is unknown in general. We present an extended modeling of GKLT that overcomes the need of manual adjustment of the uncertainty parameter. We establish an extended optimization error function for GKLT feature tracking, from which we derive extended parameter update rules and a new optimization algorithm in the context of KLT tracking. By this means we give a new formulation of KLT tracking using known camera parameters originating, for instance, from a controlled environment. We compare the extended GKLT tracking method with the original GKLT and the standard KLT tracking using real data. The experiments show that the extended GKLT tracking performs better than the standard KLT and reaches an accuracy up to several times better than the original GKLT with an improperly chosen value of the uncertainty parameter.

1 Introduction

Three-dimensional (3D) reconstruction from digital images requires, more or less explicitly, a solution to the correspondence problem. A solution can be found by matching and tracking algorithms. The choice between matching and tracking depends on the problem setup, in particular on the camera baseline, available prior knowledge, scene constraints and requirements in the result.

Recent research [1,2] deals with the special problem of active, purposive 3D reconstruction inside a controlled environment, like the robotic arm in Fig. 1, with active adjustment of sensor parameters. These methods, also known as next-best-view (NBV) planning methods, use the controllable sensor and the additional information about camera parameters endowed by the controlled environment to meet the reconstruction goals (e.g. no more than n views, defined reconstruction accuracy) in an optimal manner.

Matching algorithms suffer from ambiguities. On the other hand, feature tracking methods are favored by small baselines that can be generated in the context of NBV planning methods. Thus, KLT tracking turns into the method of choice for solving the correspondence problem within NBV procedures. Previous work has shown that it is worth to look for possible improvements of the KLT tracking method by incorporating prior knowledge about camera parameters. This additional knowledge may originate from a controlled environment or from an estimation step within the reconstruction process. Using an estimation of the camera parameters implicates the need to address the uncertainty of this information explicitly.

Originally, the formulation of feature tracking based on an iterative optimization process is the work of Lucas and Kanade [3]. Since then a rich variety of extensions to the original formulation has been published, as surveyed by Baker and Matthews [4]. These extensions may be used independently from the incorporation of camera parameters. For example, Fusiello et al. [5] deal with the removal of spurious correspondences by using robust statistics. Zinsser et al. [6] propose a separated tracking process by inter-frame translation estimation using block matching followed by estimating the affine motion with respect to the template image. Heigl [7] uses an estimation of camera parameters to move features along their epipolar line, but he does not consider the uncertainty of the estimation. Trummer et al. [8,9] give a formulation of KLT tracking, called *Guided KLT tracking (GKLT)*, with known camera parameters regarding uncertainty, using the traditional optimization error function. They adjust uncertainty manually and do not estimate it within the optimization process.

This paper contributes to the solution of the correspondence problem by incorporating known camera parameters into the model of KLT tracking under explicit treatment of uncertainty. The resulting extension of GKLT tracking estimates the feature warping together with the amount of uncertainty during the optimization process. Inspired by the EM approach [10], the extended GKLT tracking algorithm uses alternating iterative estimation of hidden information and result values.

The remainder of the paper is organized as follows. Section 2 gives a repetition of KLT tracking basics and defines the notation. It also views the adaptations of GKLT tracking. The incorporation of known camera parameters into the KLT framework with uncertainty estimation is presented in Sect. 3. Section 4 lists experimental results that allow the comparison between the standard KLT, GKLT and the extended GKLT tracking presented in Sect. 3. The paper is concluded in Sect. 5 by summary and outlook to future work.

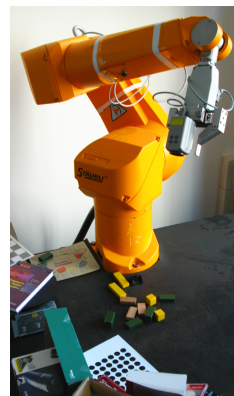


Fig. 1. Robotic arm *Staubli RX90L* as an example of a controlled environment

2 KLT and GKLT Tracking

For the sake of clarity of the explanations in the following sections, we first review the basic KLT tracking and the adaptations for GKLT tracking. The complete derivations can be found in [3,4] (KLT) and [8] (GKLT).

2.1 KLT Tracking

Given a feature position in the initial frame, KLT feature tracking aims at finding the corresponding feature position in the consecutive input frame with intensity function $I(\mathbf{x})$. The initial frame is the template image with intensity function $T(\mathbf{x})$, $\mathbf{x} = (x, y)^T$. A small image region and the intensity values inside describe a feature. This descriptor is called feature patch P . Tracking a feature means that the parameters $\mathbf{p} = (p_1, \dots, p_n)^T$ of a warping function $W(\mathbf{x}, \mathbf{p})$ are estimated iteratively, trying to minimize the squared intensity error over all pixels in the feature patch. A common choice is affine warping by

$$W^a(\mathbf{x}, \mathbf{p}^a) = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \quad (1)$$

with $\mathbf{p}^a = (\Delta x, \Delta y, a_{11}, a_{12}, a_{21}, a_{22})^T$. The error function of the optimization problem can be written as

$$\epsilon(\mathbf{p}) = \sum_{\mathbf{x} \in P} (I(W(\mathbf{x}, \mathbf{p})) - T(\mathbf{x}))^2, \quad (2)$$

where the goal is to find $\arg \min_{\mathbf{p}} \epsilon(\mathbf{p})$. Following the additional approach (cf. [4]), the error function is reformulated yielding

$$\epsilon(\Delta \mathbf{p}) = \sum_{\mathbf{x} \in P} (I(W(\mathbf{x}, \mathbf{p} + \Delta \mathbf{p})) - T(\mathbf{x}))^2. \quad (3)$$

To resolve for $\Delta \mathbf{p}$ in the end, first-order Taylor approximations are applied to clear the functional dependencies of $\Delta \mathbf{p}$. Two approximation steps give

$$\epsilon'(\Delta \mathbf{p}) = \sum_{\mathbf{x} \in P} (I(W(\mathbf{x}, \mathbf{p})) + \nabla I \nabla_{\mathbf{p}} W(\mathbf{x}, \mathbf{p}) \Delta \mathbf{p} - T(\mathbf{x}))^2 \quad (4)$$

with $\epsilon(\Delta \mathbf{p}) \approx \epsilon'(\Delta \mathbf{p})$ for small $\Delta \mathbf{p}$. The expression in (4) is differentiated with respect to $\Delta \mathbf{p}$ and set to zero. After rearranging the terms it follows that

$$\Delta \mathbf{p} = \mathbf{H}^{-1} \sum_{\mathbf{x} \in P} (\nabla I \nabla_{\mathbf{p}} W(\mathbf{x}, \mathbf{p}))^T (T(\mathbf{x}) - I(W(\mathbf{x}, \mathbf{p}))) \quad (5)$$

using the first-order approximation \mathbf{H} of the Hessian,

$$\mathbf{H} = \sum_{\mathbf{x} \in P} (\nabla I \nabla_{\mathbf{p}} W(\mathbf{x}, \mathbf{p}))^T (\nabla I \nabla_{\mathbf{p}} W(\mathbf{x}, \mathbf{p})). \quad (6)$$

Equation (5) delivers the iterative update rule for the warping parameter vector.

2.2 GKLT Tracking

In comparison to standard KLT tracking, GKLT [8] uses knowledge about intrinsic and extrinsic camera parameters to alter the translational part of the warping function. Features are moved along their respective epipolar line, but allowing for translations perpendicular to the epipolar line caused by the uncertainty in the estimate of the epipolar geometry. The affine warping function from (1) is changed to

$$W_{EU}^a(\mathbf{x}, \mathbf{p}_{EU}^a, \mathbf{m}) = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \frac{-l_3}{l_1} - \lambda_1 l_2 + \lambda_2 l_1 \\ \lambda_1 l_1 + \lambda_2 l_2 \end{pmatrix} \quad (7)$$

with $\mathbf{p}_{EU}^a = (\lambda_1, \lambda_2, a_{11}, a_{12}, a_{21}, a_{22})^T$; the respective epipolar line $\mathbf{l} = (l_1, l_2, l_3)^T = \mathbf{F}\tilde{\mathbf{m}}$ is computed using the fundamental matrix \mathbf{F} and the feature position (center of feature patch) $\tilde{\mathbf{m}} = (x_m, y_m, 1)^T$. In general, the warping parameter vector is $\mathbf{p}_{EU} = (\lambda_1, \lambda_2, p_3, \dots, p_n)^T$. The parameter λ_1 is responsible for movements along the respective epipolar line, λ_2 for the perpendicular direction. The optimization error function of GKLT is the same as the one from KLT (2), but using substitutions for the warping parameters and the warping function. The parameter update rule of GKLT derived from the error function,

$$\Delta \mathbf{p}_{EU} = \mathbf{A}_w \mathbf{H}_{EU}^{-1} \sum_{\mathbf{x} \in P} (\nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}))^T (T(\mathbf{x}) - I(W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}))), \quad (8)$$

also looks very similar to the one of KLT (5). The difference is the weighting matrix

$$\mathbf{A}_w = \begin{pmatrix} w & 0 & 0 & \dots & 0 \\ 0 & 1-w & 0 & & \\ 0 & 0 & 1 & & \vdots \\ \vdots & & & \ddots & 0 \\ 0 & & \dots & 0 & 1 \end{pmatrix}, \quad (9)$$

which enables the user to weight the translational changes (along/perpendicular to the epipolar line) by the parameter $w \in [0, 1]$ called *epipolar weight*. In [8] the authors associate $w = 1$ with the case of a perfectly accurate estimate of the epipolar geometry, since only feature translations along the respective epipolar line are realized. The more uncertain the epipolar estimate the smaller is w said to be. The case of no knowledge about the epipolar geometry is linked with $w = 0.5$, when translations along and perpendicular to the respective epipolar line are realized equally weighted.

3 GKLT Tracking with Uncertainty Estimation

The previous section briefly reviewed a way to incorporate knowledge about camera parameters into the KLT tracking model. The resulting GKLT tracking

requires manual adjustment of the weighting factor w that controls the translational parts of the warping function and thereby handles an uncertain epipolar geometry. For practical application, it is questionable how to find an optimal w and whether one allocation of w holds for all features in all sequences produced within the respective controlled environment. Hence, we propose to estimate the uncertainty parameter w for each feature during the feature tracking process.

In the following we present a new approach for GKLT where the warping parameters and the epipolar weight are optimally computed in a combined estimation step. Like the EM algorithm [10], our approach uses an alternating iterative estimation of hidden information and result values. The first step in deriving the extended iterative optimization procedure is the specification of the optimization error function of GKLT tracking with respect to the uncertainty parameter.

3.1 Modifying the Optimization Error Function

In the derivation of GKLT from [8], the warping parameter update rule is constructed from the standard error function and in the last step augmented by the weighting matrix \mathbf{A}_w to yield (8). Instead, we suggest to directly include the weighting matrix in the optimization error function. Thus, we reparameterize the standard error function to get the new optimization error function

$$\epsilon(\Delta\mathbf{p}_{EU}, \Delta w) = \sum_{\mathbf{x} \in P} (I(W_{EU}(\mathbf{x}, \mathbf{p}_{EU} + \mathbf{A}_{w, \Delta w} \Delta\mathbf{p}_{EU}, \mathbf{m})) - T(\mathbf{x}))^2. \quad (10)$$

Following the additional approach for the matrix \mathbf{A}_w from (9), we substitute $w + \Delta w$ instead of w to reach the weighting matrix $\mathbf{A}_{w, \Delta w}$ used in (10). We achieve an approximation of this error function by first-order Taylor approximation applied twice,

$$\epsilon'(\Delta\mathbf{p}_{EU}, \Delta w) = \sum_{\mathbf{x} \in P} (I(W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m})) + \nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}) \mathbf{A}_{w, \Delta w} \Delta\mathbf{p}_{EU} - T(\mathbf{x}))^2 \quad (11)$$

with $\epsilon(\Delta\mathbf{p}_{EU}, \Delta w) \approx \epsilon'(\Delta\mathbf{p}_{EU}, \Delta w)$ for small $\mathbf{A}_{w, \Delta w} \Delta\mathbf{p}_{EU}$. This allows for direct access to the warping and uncertainty parameters.

3.2 The Modified Update Rule for the Warping Parameters

We calculate the warping parameter change $\Delta\mathbf{p}_{EU}$ by minimization of the approximated error term (11) with respect to $\Delta\mathbf{p}_{EU}$ in the sense of steepest descent, $\frac{\partial \epsilon'(\Delta\mathbf{p}_{EU}, \Delta w)}{\partial \Delta\mathbf{p}_{EU}} \stackrel{!}{=} \mathbf{0}$. We get as the update rule for the warping parameters

$$\Delta\mathbf{p}_{EU} = \mathbf{H}_{\Delta\mathbf{p}_{EU}}^{-1} \sum_{\mathbf{x} \in P} (\nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}) \mathbf{A}_{w, \Delta w})^T (T(\mathbf{x}) - I(W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}))) \quad (12)$$

with the approximated Hessian

$$\mathbf{H}_{\Delta\mathbf{p}_{EU}} = \sum_{\mathbf{x} \in P} (\nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}) \mathbf{A}_{w, \Delta w})^T (\nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}) \mathbf{A}_{w, \Delta w}). \quad (13)$$

3.3 The Modified Update Rule for the Uncertainty Estimate

For calculating the change Δw of the uncertainty estimate we again perform minimization of (11), but with respect to Δw , $\frac{\partial \epsilon'(\Delta \mathbf{p}_{EU}, \Delta w)}{\partial \Delta w} \stackrel{!}{=} \mathbf{0}$. This claim yields

$$\sum_{\mathbf{x} \in P} \left(\frac{\partial}{\partial \Delta w} (\nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}) \mathbf{A}_{w, \Delta w} \Delta \mathbf{p}_{EU}) \right) \cdot (I(W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m})) + \nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}) \mathbf{A}_{w, \Delta w} \Delta \mathbf{p}_{EU} - T(\mathbf{x})) \stackrel{!}{=} \mathbf{0}. \quad (14)$$

We specify

$$\frac{\partial}{\partial \Delta w} (\nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}) \mathbf{A}_{w, \Delta w} \Delta \mathbf{p}_{EU}) = \nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}) \frac{\partial \mathbf{A}_{w, \Delta w}}{\partial \Delta w} \Delta \mathbf{p}_{EU}. \quad (15)$$

By rearrangement of (14) and using (15) we get

$$\begin{aligned} & \overbrace{\sum_{\mathbf{x} \in P} (\nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}) \frac{\partial \mathbf{A}_{w, \Delta w}}{\partial \Delta w} \Delta \mathbf{p}_{EU}) (\nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m})) \mathbf{A}_{w, \Delta w} \Delta \mathbf{p}_{EU}}^{\mathbf{h}_{\Delta w}} \\ &= \underbrace{\sum_{\mathbf{x} \in P} (\nabla I \nabla_{\mathbf{p}_{EU}} W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m}) \frac{\partial \mathbf{A}_{w, \Delta w}}{\partial \Delta w} \Delta \mathbf{p}_{EU}) (T(\mathbf{x}) - I(W_{EU}(\mathbf{x}, \mathbf{p}_{EU}, \mathbf{m})))}_{e}, \end{aligned}$$

$$\text{i.e.} \quad \mathbf{h}_{\Delta w} \mathbf{A}_{w, \Delta w} \Delta \mathbf{p}_{EU} = e. \quad (16)$$

Since e is real-valued, (16) provides one linear equation in Δw . With $\mathbf{h}_{\Delta w} = (h_1, \dots, h_n)^T$ and $\Delta \mathbf{p}_{EU} = (\Delta \lambda_1, \Delta \lambda_2, \Delta p_3, \dots, \Delta p_n)^T$ we reach the update rule for the uncertainty estimate,

$$\Delta w = \frac{e - h_2 \Delta \lambda_2 - h_3 \Delta p_3 - \dots - h_n \Delta p_n}{h_1 \Delta \lambda_1 - h_2 \Delta \lambda_2} - w. \quad (17)$$

3.4 The Modified Optimization Algorithm

In comparison to the KLT and GKLT tracking, we now have two update rules: one for \mathbf{p}_{EU} and one for w . These update rules, just as in the previous KLT versions, compute optimal parameter changes in the sense of least-squares estimation found by steepest descent of an approximated error function. We combine the two update rules in an EM-like approach. For one iteration of the optimization algorithm, we calculate $\Delta \mathbf{p}_{EU}$ (using $\Delta w = 0$) followed by the computation of Δw with respect to the $\Delta \mathbf{p}_{EU}$ just computed in this step. Then we apply the change to the warping parameter using the actual w .

The modified optimization algorithm as a whole is:

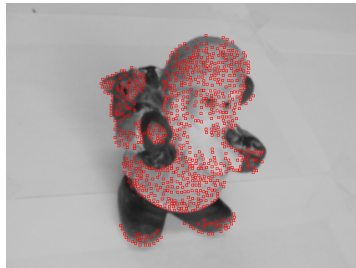
1. initialize \mathbf{p}_{EU} and w
2. compute $\Delta \mathbf{p}_{EU}$ by (12)
3. compute Δw by (17) using $\Delta \mathbf{p}_{EU}$

4. update \mathbf{p}_{EU} : $\mathbf{p}_{EU} \leftarrow \mathbf{p}_{EU} + \mathbf{A}_{w, \Delta w} \Delta \mathbf{p}_{EU}$
5. update w : $w \leftarrow w + \Delta w$
6. if changes are small, stop; else go to step 2.

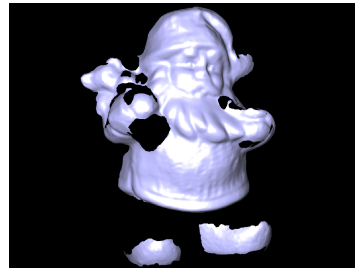
This new optimization algorithm for feature tracking with known camera parameters uses the update rules derived from the extended optimization error function (12) for GKLT tracking. Most importantly, these steps provide a combined estimation of the warping and the uncertainty parameters. Hence, there is no more need to adjust the uncertainty parameter manually as in [8].

4 Experimental Evaluation

Let us denote the extended GKLT tracking method shown in the previous section by GKLT₂, the original formulation [8] by GKLT₁. In this section we quantitatively compare the performances of the KLT, GKLT₁ and GKLT₂ feature tracking methods with and without the presence of noise in the prior knowledge about camera parameters. For GKLT₁, we measure its performance with respect to different values of the uncertainty parameter w .



(a) Initial frame of the test sequence with 746 features selected.



(b) View of the set of 3D reference points. Surface mesh for illustration only.

Fig. 2. Test and reference data

As performance measure we use *tracking accuracy*. Assuming that accurately tracked features lead to an accurate 3D reconstruction, we visualize the tracking accuracy by plotting the mean error distances μ_E and standard deviations σ_E of the resulting set of 3D points, reconstructed by plain triangulation, compared to a 3D reference. We also note mean trail lengths.

Figure 2 shows a part of the data we used for our experiments. The image in Fig. 2(a) is the first frame of our test sequence of 26 frames taken from a Santa Claus figurine. The little squares indicate the positions of 746 features initialized for the tracking procedure. Each of the trackers (KLT, GKLT₁ with $w = 0, \dots$, GKLT₁ with $w = 1$, GKLT₂) has to track these features through the following

frames of the test sequence. We store the resulting trails and calculate the mean trail length for each tracker. Using the feature trails and the camera parameters, we do a 3D reconstruction by plain triangulation for each feature that has a trail length of at least five frames. The resulting set of 3D points is rated by comparison with the reference set shown in Fig. 2(b). This yields μ_E , σ_E of the error distances between each reconstructed point and the actual closest point of the reference set for each tracker. The 3D reference points are provided by a highly accurate (measurement error below $70\mu\text{m}$) fringe-projection measurement system [11]. We register these reference points into our measurement coordinate frame by manual registration of distinctive points and an optimal estimation of a 3D Euclidean transformation using dual number quaternions [12]. The camera parameters we apply are provided by our robot arm *Stäubli RX90L* illustrated in Fig. 1. Throughout the experiments, we initialize GKLT₂ with $w = 0.5$.

The extensions of GKLT₁ and GKLT₂ affect the translational part of the feature warping function only. Therefore, we assume and estimate pure translation of the feature positions in the test sequence.

Table 1. Accuracy evaluation by mean error distance μ_E (mm) and standard deviation σ_E (mm) for each tracker. GKLT₁ showed accuracy from 9% better to 269% worse than KLT, depending on choice of w relative to respective uncertainty of camera parameters. GKLT₂ performed better than standard KLT in any case tested. Without additional noise accuracy of GKLT₂ was 5% better than KLT.

	KLT	GKLT ₁ , w equals:											GKLT ₂
		0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0	
<i>Using camera parameters without additional noise:</i>													
μ_E (mm)	2.68	9.90	3.52	3.15	2.93	2.77	2.77	2.65	2.62	2.51	2.45	3.90	2.56
σ_E (mm)	3.70	6.99	4.65	4.08	3.63	3.38	3.63	3.55	3.41	3.17	2.77	5.12	3.36
<i>Using disturbed camera parameters:</i>													
μ_E (mm)	2.68	5.09	2.76	2.68	2.75	2.76	2.77	2.78	2.88	3.05	3.35	7.98	2.66
σ_E (mm)	3.70	5.60	3.40	3.37	3.60	3.71	3.63	3.50	4.05	4.08	4.30	6.90	3.61

Throughout the experiments GKLT₂ produced trail lengths that are comparable to standard KLT. The mean runtimes (Intel Core2 Duo, 2.4 GHz, 4 GB RAM) per feature and frame were 0.03 ms for standard KLT, 0.14 ms for GKLT₁ with $w = 0.9$ and 0.29 ms for GKLT₂.

The modified optimization algorithm presented in the last section performs two non-linear optimizations in each step. This results in larger runtimes compared to KLT and GKLT₁ which use one non-linear optimization in each step. The quantitative results of the tracking accuracy are printed in Table 1.

Results using camera parameters without additional noise. GKLT₂ showed a mean error 5% less than KLT, standard deviation was reduced by 9%. The results

of GKLT₁ were scattered for different values of w . The mean error reached from 9% less at $w = 0.9$ to 269% larger at $w = 0$ than with KLT. The mean trail length of GKLT₁ was comparable to KLT at $w = 0.9$, but up to 50% less for all other values of w . An optimal allocation of $w \in [0, 1]$ for the image sequence used is likely to be in $]0.8, 1.0[$, but it is unknown.

Results using disturbed camera parameters. To simulate serious disturbance of the prior knowledge used for tracking, the camera parameters were selected completely random for this test. In the case of fully random prior information, GKLT₂ could adapt the uncertainty parameter for each feature in each frame to reduce the mean error by 1% and the standard deviation by 2% relative to KLT. Instead, GKLT₁ uses a global value of w for all features in all frames. Again it showed strongly differing performance with respect to the value of w . In the case tested GKLT₁ reached the result of KLT at $w = 0.2$ considering mean error and mean trail length. For any other allocation of the uncertainty parameter the mean reconstruction error was up to 198% larger and the mean trail length up to 56% less than with KLT.

5 Summary and Outlook

In this paper we presented a way to extend the GKLT tracking model for integrated uncertainty estimation. For this, we incorporated the uncertainty parameter into the optimization error function resulting in modified parameter update rules. We established a new EM-like optimization algorithm for combined estimation of the tracking and the uncertainty parameters.

The experimental evaluation showed that our extended GKLT performed better than standard KLT tracking in each case tested, even in the case of completely random camera parameters. In contrast the results of the original GKLT varied seriously. An improper choice of the uncertainty parameter caused errors several times larger than with standard KLT. The fitness of the respectively chosen value of the uncertainty parameter was shown to depend on the uncertainty of prior knowledge, which is unknown in general.

Considering the experiments conducted, there are few configurations of the original GKLT that yield better results than KLT and the extended GKLT. Future work is necessary to examine these cases of properly chosen values of the uncertainty parameter. This is a precondition for improving the extended GKLT to reach results closer to the best ones of the original GKLT tracking method.

References

1. Wenhardt, S., Deutsch, B., Angelopoulou, E., Niemann, H.: Active Visual Object Reconstruction using D-, E-, and T-Optimal Next Best Views. In: Computer Vision and Pattern Recognition, CVPR 2007, June 2007, pp. 1–7 (2007)
2. Chen, S.Y., Li, Y.F.: Vision Sensor Planning for 3D Model Acquisition. IEEE Transactions on Systems, Man and Cybernetics – B 35(4), 1–12 (2005)

3. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proceedings of 7th International Joint Conference on Artificial Intelligence, pp. 674–679 (1981)
4. Baker, S., Matthews, I.: Lucas-Kanade 20 Years On: A Unifying Framework. *International Journal of Computer Vision* 56, 221–255 (2004)
5. Fusiello, A., Trucco, E., Tommasini, T., Roberto, V.: Improving feature tracking with robust statistics. *Pattern Analysis and Applications* 2, 312–320 (1999)
6. Zinsser, T., Graessl, C., Niemann, H.: High-speed feature point tracking. In: Proceedings of Conference on Vision, Modeling and Visualization (2005)
7. Heigl, B.: Plenoptic Scene Modelling from Uncalibrated Image Sequences. PhD thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg (2003)
8. Trummer, M., Denzler, J., Munkelt, C.: KLT Tracking Using Intrinsic and Extrinsic Camera Parameters in Consideration of Uncertainty. In: Proceedings of 3rd International Conference on Computer Vision Theory and Applications (VISAPP), vol. 2, pp. 346–351 (2008)
9. Trummer, M., Denzler, J., Munkelt, C.: Guided KLT Tracking Using Camera Parameters in Consideration of Uncertainty. *Lecture Notes in Communications in Computer and Information Science (CCIS)*. Springer, Heidelberg (to appear)
10. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data. *Journal of the Royal Statistical Society* 39, 1–38 (1977)
11. Kuehmstedt, P., Munkelt, C., Matthies, H., Braeuer-Burchardt, C., Notni, G.: 3D shape measurement with phase correlation based fringe projection. In: Osten, W., Gorecki, C., Novak, E.L. (eds.) *Optical Measurement Systems for Industrial Inspection V*, vol. 6616, p. 66160B. SPIE (2007)
12. Walker, M.W., Shao, L., Volz, R.A.: Estimating 3-D location parameters using dual number quaternions. *CVGIP: Image Understanding* 54(3), 358–367 (1991)