Unscented KLT: nonlinear feature and uncertainty tracking

Leyza Baldo Dorini Siome Klein Goldenstein
Unicamp - Universidade Estadual de Campinas
Instituto de Computação
Caixa Postal 6176, 13083-971, Campinas, SP, Brasil
{ldorini, siome}@ic.unicamp.br

Abstract

Accurate feature tracking is the foundation of several high level tasks, such as 3D reconstruction and motion analysis. Although there are many feature tracking algorithms, most of them do not maintain information about the error of the data being tracked. In this paper, we propose a new generic framework that uses the Scaled Unscented Transform (SUT) to augment arbitrary feature tracking algorithms, by introducing Gaussian Random Variables (GRV) for the representation of features' locations uncertainties. Here, we apply the framework to the well-understood Kanade-Lucas-Tomasi (KLT) feature tracker, giving birth to what we call Unscented KLT (UKLT). It tracks probabilistic confidences and better rejects errors, all on-line, and leads to more robust computer vision applications. We also validade the experiments with a bundle adjustment procedure, using real and synthetic sequences.

1. Introduction

Several problems in computer vision depend on the determination of correspondences through a sequence of images. Feature tracking, finding correspondences along image sequences, tracks selected image features as they move through the image frames. It is an instance of the general problem of optical flow, that represents the motion estimate of a sequence frame at sparse image positions.

It is important to select reliable features to track, that do not suffer from the aperture problem effect [15]. There are many feature tracking algorithms, each relying on different assumptions and objectives.

The Kanade-Lucas-Tomasi feature tracker (KLT), used in this paper, is based initially on the work of Lucas e Kanade [10], and was developed by Tomasi and Kanade [15]. The matching criteria between consecutive frames is based on the minimization of the sum of squared differences (SSD) of window intensities, assuming a translational movement model.

Shi and Tomasi [12] extended this algorithm in order to take into account more complex displacements, considering the affine model. They have proposed a technique to monitor the quality of the features being tracked. If the residual of the match between image regions in the first and the current frame exceeds a threshold, the feature is considered unreliable and is rejected. Later, some works presented an extension of Shi-Tomasi tracker that take into account changes in illumination and reflection [5, 6].

Robust matching approaches automatically detect features to be rejected. Examples include Torr et al. [16] that adopts a RANSAC [2] approach to eliminate outliers and Fusiello et. al. [3] that propose an extension to the KLT, by introducing an automatic scheme for reject features based on a rejection rule called X84.

The Scale-Invariant Feature Transform (SIFT) [9] is a method to feature selection that extracts distinctive invariant features from images so that they can be used to matching between different frames. The features are scale and rotation invariant, and are shown to provide reliable matching through a wide range of affine distortion, changes in 3D viewpoint and are partially invariant to illumination changes. It uses a cascade filtering approach, and operations with high computational cost are applied only in features that satisfy initial requirements.

Unfortunately, none of these algorithms consider the uncertainties of the data being tracked, the information about the estimate's reliability, as does the statistical optical flow [13].

In Section 2, we discuss types of tracking errors and the importance of an uncertainty measure related to each feature point location. Section 3 presents the UKLT formalization. In Section 4, we show experimental results using real sequences and ground truth data, and an application on a bundle adjustment procedure. Finally, we discuss our conclusions and future work.

2. Tracking errors and uncertainty estimation

Reliable feature tracking is the core for several problems in the field of computer vision. Structure from motion algorithms, for example, can reconstruct the 3D scene depth and camera motion from a set of feature points tracked through a sequence of frames. The quality of the reconstruction depends on the accuracy of the feature tracking.

There are two main categories of tracking errors: location imprecision and false matches [20]. In the former, feature points are in a location differing of a few pixels from the true position. False matches occur when a feature is mapped to a different location, causing gross mistakes.

In this paper, we introduce a new generic framework that augments arbitrary feature tracking algorithms to track the first two moments of a probability distribution (mean and covariance). Here, for simplicity, we focused on one of the most commonly used methods, the vanilla KLT [15], to demonstrate the potential and benefits of our method.

Representing the features' locations as Gaussian Random Variables (GRVs), completely described by the first two moments of a probability distribution, we associate to each location an uncertainty measure, represented by the second central moment. At this way, we have a confidence region that represents the location errors, information that can be used in a great variety of applications.

If we have a reliable estimate of uncertainty, we can improve the parameters fit to noisy data, using a weighted least squares fit that takes into account the uncertainty infomation, by minimizing the Mahalonobis distance between the data and the predicted model [17]. This is used in several computer vison problems, such as bundle adjustment and epipolar geometry estimation.

Although the use of covariance matrices for the features' locations uncertainty representation have been discussed in the literature [8], with our method we have obtained good results in bundle adjustment for structure from motion and in the improvement of the feature location estimates. Although estimation of uncertainties also have errors, their use is valid and provides specially good results with our method.

Chowdhurry [1] has derived an explicit expression for the error covariance in motion and structure estimates as a function of the error covariance in the feature positions in the images, but his work does not consider the effects of outliers. Steele and Jaynes [14] proposed a method to improve the uncertainty estimates of the features' locations, propagating the covariance through the Jacobian of the feature location estimator. Their estimate does not take into account the uncertainty of the feature tracking algorithm. In the same way, Zhu et al. [21] have used a confidence measure based on the gray level difference, without considering the inherent error of the tracking algorithm.

3. Tracking random variables

In this paper, our main goals are: feature uncertainty representation and to improve the accuracy in the feature location's estimates. To accomplish this, we need to formulate the problem with appropriate mathematical models.

The feature uncertainty representation is done through the modeling of the feature locations as GRVs. This allows us to use a probability distribution (in fact, we use only its first two moments) to represent the correspondence.

When using concepts of predictive filters, a family of parameter estimation techniques, we can obtain better estimates. These filters propagate the parameters and their uncertainties through a system dynamics, and combine this preliminary estimative with data obtained from the system's observations [4]. There are different types of predictive filters, each relying on different assumptions and objectives.

The linearity constraints required by the well-known Kalman Filter (KF) are not satisfied in many practical applications, and suitable extensions have to be used. The Extended Kalman Filter (EKF) is an estimator for nonlinear systems that linearize all the nonlinear models, so that we can use the traditional linear KF equations. However, the EKF has some drawbacks, usually leading to poor representations of the nonlinear functions and probability distributions of interest, that results in incorrect estimates.

The Unscented Kalman Filter (UKF) [7], based on the Unscented Transform (UT), uses the true nonlinear model, thus surpassing the EKF limitations. We use the UT in this paper to estimate transformed GRVs.

3.1. The Scaled Unscented Transform

The Unscented Transform (UT) calculates the statistics of a random variable that undergoes a non-linear transformation. The key idea of the UT is that is easier to approximate a Gaussian distribution than an arbitrary nonlinear function/transformation [7]. In this paper, we will use the Scaled Unscented Transform (SUT), an extension of the Unscented Transform that ensures the positive definiteness condition of the transformed covariance matrices [18].

Consider a random variable $\vec{x}$ (dimension n) that undergoes a nonlinear transformation $\vec{y} = g(\vec{x})$. Let $\bar{\vec{x}}$ and Σ_x be the mean and covariance matrix of $\vec{x}$, respectively. To calculate the mean and covariance of $\vec{y}$, we generate a deterministic set $\mathcal{X}$ of $2n+1$ sigma points $\mathcal{X}_i$ as follows [18]:

$$\begin{aligned} &\mathcal{X}_0 = \bar{\vec{x}}, \\ &\mathcal{X}_i = \bar{\vec{x}} + (\sqrt{(n+\lambda)\Sigma_x})_i, \quad i = 1, \ldots, n, \\ &\mathcal{X}_i = \bar{\vec{x}} - (\sqrt{(n+\lambda)\Sigma_x})_{(i-n)}, \quad i = n+1, \ldots, 2n, \end{aligned} \tag{1}$$

whit $\lambda = \alpha^2(n+\kappa) - n$, where α determines the spread of the sigma points around $\bar{\vec{x}}$ and κ is a scale parameter. Finally, $(\sqrt{(n+\lambda)\Sigma_x})_i$ is the ith row of the matrix square root. This deterministic choice of the sigma points guarantees that they completely capture the true mean and covariance of the prior random variable $\vec{x}$. We use $\alpha = 0.9$ and $\kappa = 0$ for all sequences. The value of α was chosen to minimize the particle spread, decreasing the occurrence of false outliers.

Each sigma point has a associated weight W_i:

$$\begin{aligned} W_0^m &= \frac{\lambda}{n+\lambda}, \\ W_0^c &= \frac{\lambda}{n+\lambda} + 1 - \alpha^2 + \beta, \\ W_i^m = W_i^c &= \frac{1}{2(n+\lambda)} \qquad i = 1, \ldots, 2n, \end{aligned} \tag{2}$$

where β is used for incorporate knowledge of the higher order moments of the distribution. We use $\beta = 2$ for all sequences, since this should always be the value for Gaussian priors. The subscript m indicates the weight for the mean calculation, and c for the covariance calculation. The sigma points are propagated through a nonlinear transformation

$$\mathcal{Y}_i = f(\mathcal{X}_i).$$

The mean and covariance are approximated as

$$\bar{\vec{y}} = \sum_{i=0}^{2n} W_i^m \mathcal{Y}_i, \tag{3}$$

and

$$\Sigma_y = \sum_{i=0}^{2n} W_i^c \, (\mathcal{Y}_i - \bar{\vec{y}})(\mathcal{Y}_i - \bar{\vec{y}})^T. \tag{4}$$

Figure 1 illustrates the method. The SUT approach results in an approximation that is accurate to at least the second order. The third and higher order moments accuracy is determined by the choice of α and β [19]. Just for comparison, the Extended Kalman Filter calculates the posterior mean and covariance accurately only to the first order [7].

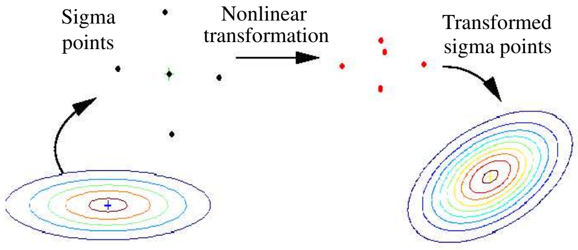


Figure 1. The Unscented Transform.

3.2. Unscented KLT: UKLT

When analyzing the feature tracking uncertainty, the existing algorithms usually consider only the relationship between the noise models and the feature points' covariances, emphasizing the local image characteristics and ignoring the inherent error of the tracking algorithm. In this paper, when we use the SUT to propagate the GRVs through a nonlinear transformation, represented here by the KLT feature tracking algorithm, we are taking into account its inherent error.

Let $u(\mu_i, \Sigma_i)_k$ be the state vector of our system, where for each discrete time step k we have the mean and covariance (that describes a GRV) of each feature point i. At time step k, we apply the SUT (Algorithm 1) to each feature point i in the state vector $u(\mu_i, \Sigma_i)_k$: generate the sigma points (Equation 1), propagate them using the KLT tracker, and finally calculate the corresponding mean and covariance (Equations 3 and 4). Note that with the use of only the first two moments, we have a trade off between flexibility of representation and computational cost.

Algorithm 1 The Scaled Unscented Transform

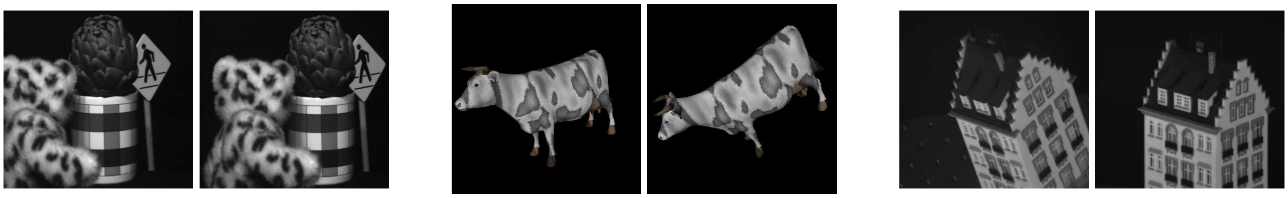
```
function SCALED UNSCENTED TRANSFORM
    given $n$ feature points selected by the KLT algorithm
    for each feature point do
        generate $2L+1$ sigma points, where $L$ is the RV dimension;
        propagate the sigma points using the KLT tracker;
        calculate the mean (Equation 3);
        calculate the covariance matrix (Equation 4);
    end for
end function
```

For each time step k, we also make an observation of the system, denoted by $v(\mu_i, \Sigma_i)_k$, that considers the local image characteristics. This gives extra information to combine with the first GRV. We do this through the generation of a new GRV, whose covariance is estimated based on the gray level variation. The covariance is given by the inverse of the matrix

$$C = \begin{bmatrix} \nabla^2 x & \nabla x \nabla y \\ \nabla x \nabla y & \nabla^2 y \end{bmatrix},$$

where ∇x and ∇y represent the gradient in the x and y directions, respectively. The mean is the coordinate estimated by the KLT tracker.

On the context of predictive filters, we can see the SUT estimate as the state prediction, and this new GRV as an observation of the system. To obtain a better estimate of the next state, we make the fusion of this two partial estimates using the Maximum Likelihood Estimation (MLE),



(a) *Artichoke* synthetic sequence.

(b) *Cow* synthetic sequence.

(c) *Hotel* real sequence.

Figure 3. First and last frame of each sequence. The sequences (a) and (b) have a ground truth.

an inference strategy that consists on choose the world parameters that maximize the observed measured probabilities. Figure 2 illustrates the ideas of our method.

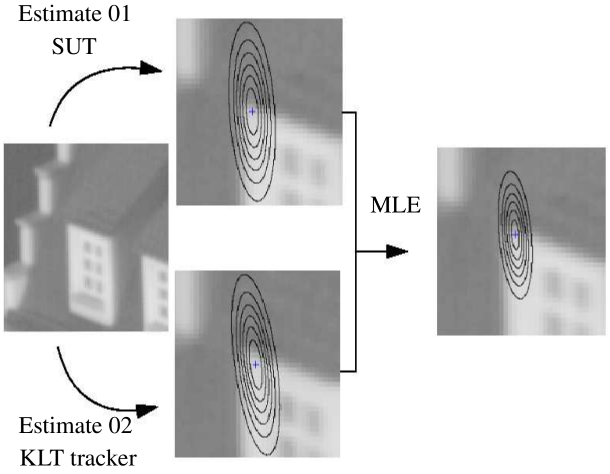


Figure 2. Our algorithm.

The MLE finds the better estimate of the features' positions and covariances, and the algorithm is reiterated using this new locations as inputs. The UKLT algorithm is summarized in Algorithm 2.

Algorithm 2 Unscented KLT

```
function UNSCENTED KLT (UKLT)
    given an image sequence;
    select feature points and calculate their initial Gaussian distribution
    for each feature point do
        compute the SUT using algorithm 1;
        use the KLT feature tracker to obtain the observation;
        fuse this estimates using MLE;
    end for
end function
```

3.3. Rejecting outliers

Due to the difficulty of the feature tracking problem, the existing matching methods commonly output bad correspondences, making outlier rejection an essential step. Using the problem formalization of Section 3, outliers are rejected on-line, during the tracking itself.

Our algorithm propagates five points and calculates a covariance matrix for each selected feature point. When one of the five sigma points is mapped to a wrong location (i.e. far from its "real" position), our method discards the respective feature point. This usually happens when the main sigma point (the KLT estimate for the original location) is mapped to a region that is very close of the limit between a rich and a poor texture pattern. One or more sigma points falls on the poor texture and KLT looses the tracking.

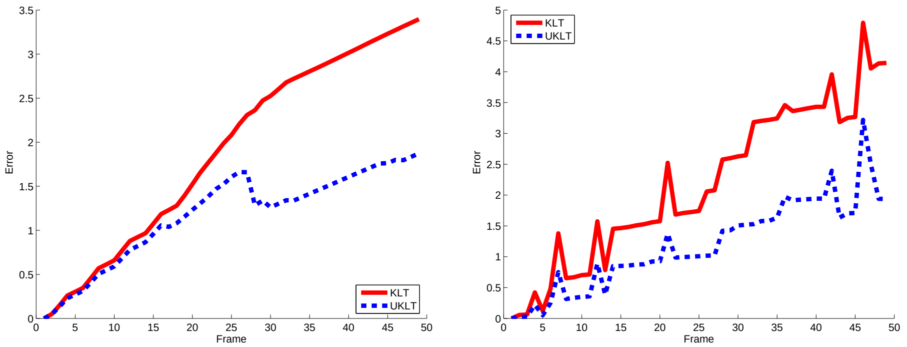
When the Equation 4 or the result of the MLE fusion is an non-positive semi-definite matrix (that is, it is not a covariance matrix), the feature point is also rejected. This aspect still needs a deeper mathematical analysis.

4. Results

In this paper, we compare the tracking accuracy of our algorithm against the standard KLT on three different sequences. We generate the first synthetic sequence, Figure 3(a), by a sequence of controlled translational warpings of the known *Artichoke* image. For the second synthetic sequence, Figure 3(b), we rendered the animation of a textured 3D model of a cow – we know the 3D coordinates, and their 2D projected image coordinates, for every frame. The real sequence is the *Hotel*[1], a static scene observed by a moving camera rotating and translating. The *Cow* sequence has an image resolution of 512×512 pixels, and the other two sequences of 512×480 pixels. The three sequences have 50 frames.

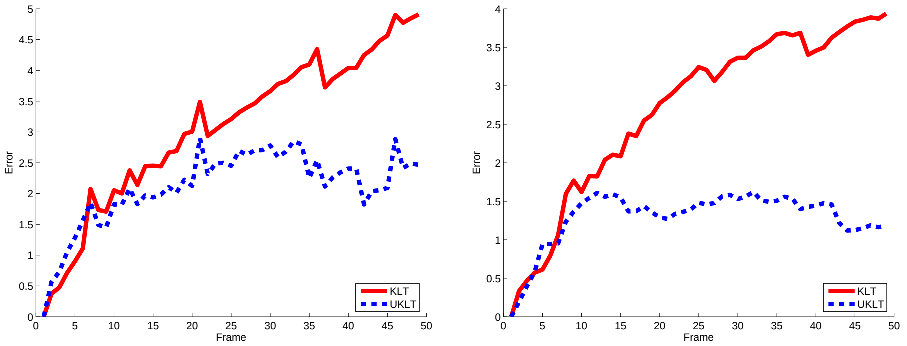
To analyze the results of our algorithm, we evaluate its performance in two steps: feature tracking and 3D reconstruction. Shortly, we apply both KLT and UKLT in each image sequence and (a) measure the accuracy of the estimated locations using different metrics and (b) use the cor-

[1] CMU/VASC Image Database. http://vasc.ri.cmu.edu/idb/



(a) *Artichoke* sequence (b) *Cow* sequence

Figure 4. Euclidean distance of the estimated feature positions from the real ones (synthetic sequences) (lower is better).



(a) *Cow* sequence (b) *Hotel* sequence

Figure 5. RMS distance of the feature points to the corresponding epipolar lines (lower is better).

respondence set on a bundle adjustment procedure and measure the reprojection error. The results confirm that the use of the random variables obtained by our method (Section 3) leads to a improved feature tracking and consequently to a more accurate 3D reconstruction.

4.1. Feature tracking analysis

As we have the ground truth for synthetic sequences, we can measure the difference between the estimated and the “real” position of each feature point. We compare the difference frame by frame using the estimates of the KLT and UKLT algorithms.

Figure 4(a) shows a plot of the distance error magnitude for the synthetic Artichoke sequence and Figure 4(b) for the Cow sequence. Note that our method gets better estimates in both sequences. This lower error is not only because of better estimates, but also because our method reject features that are not well tracked, while KLT algorithm stills preserves them (Section 3).

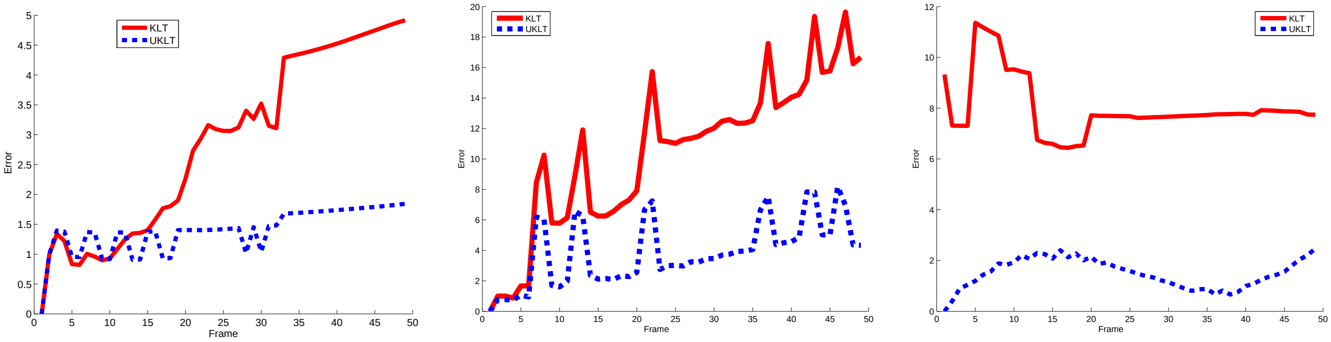
As a second validation procedure, we compute the fundamental matrix between the first and last frames of each sequence, and then compute the Root Mean Square (RMS) distance of the tracked points from the corresponding epipolar lines. If the epipolar geometry is estimated exactly, all points should lie on epipolar lines. Figure 5 shows the RMS distance of the feature points from the corresponding epipolar lines in each sequence frame. We compute the fundamental matrices using the UKLT and KLT correspondences. Figure 5(a) shows the results for the synthetic Cow sequence and Figure 5(b) for the real Hotel sequence. Note how our method reaches better results than the standard KLT.

We also compare the fundamental matrix estimated from the ground truth correspondence set with the fundamental matrix estimated from the correspondences tracked. We used the algorithm proposed by Zhang [20]. Table 1 presents the results.

Based on the experimental results, we can conclude that our method tracks features more robustly than KLT, getting better estimates and rejecting outliers. Our method also propagates a GRV that represents the features’ locations.

Sequence	UKLT			KLT		
	dEpLine	dGT	Zhang	dEpLine	dGT	Zhang
Artichoke	1.31	1.87	57.10	2.26	3.40	178.66
Cow	1.98	2.59	28.11	4.81	10.01	56.53
Hotel	1.18	–	–	3.94	–	–

Table 1. Error metrics. dEpLine: RMS distance to the epipolar line; dGT: distance between the real and the estimated position; and Zhang: fundamental matrix comparison using the Zhang's method. Measurements for the last frame (lower is better).



(a) *Artichoke* sequence (b) *Cow* sequence (c) *Hotel* sequence

Figure 6. Reprojection error magnitude.

This information is useful in various applications, as in a bundle adjustment procedure.

4.2. Application to bundle adjustment

The tracked features are used to estimate camera motion and 3D scene depth using a structure from motion algorithm. To estimate the unknown 3D feature and camera parameters from the observations, and reconstruct the scene, one minimizes some measure of their total prediction error.

Bundle adjustment is the model refinement part of this, refining the visual reconstruction to obtain both 3D structure and viewing optimal parameters estimates. We say optimal because bundle adjustment involves the minimization of a cost function, related to the model fitting error. The choice of the cost function is an important step [17].

One of the most basic parameter estimation methods is nonlinear least squares, a classic formulation of bundle adjustment computations.

Suppose that we have vectors of observations v_i predicted by a model $z_i = z_i(x)$, where x is a vector of model parameters. Nonlinear least squares takes as estimates the parameters values that minimize the weighted Sum of Squared Error (SSE) cost function [17]:

$$f(x) \equiv \frac{1}{2} \sum_i \Delta z_i(x)^T W_i \Delta z_i(x), \qquad (5)$$

where $\Delta z_i(x)$ is the feature prediction error and W_i is an arbitrary symmetric positive definite (SPD) weight matrix. In this paper, we use W as the covariance matrix obtained by our method, giving us an uncertainty measure of the structure and motion parameters.

In this paper, we use the structure from motion and bundle adjustment procedures described in [11]. To evaluate the quality on the 3D reconstruction, we reproject the estimated feature points and measure the error magnitude to the ground truth, in the case of the synthetic sequences, and the RMS distance to the epipolar line for the real sequence.

Figure 6(a-b) illustrates the results for the synthetic sequences and Figure 6(c) for the Hotel sequence. Note the superior performance of our method. In the Artichoke sequence, the reconstruction obtained by our method is better in all frames after frame 12. In the Cow sequence, the rotation cause the instabilities.

4.3. Implementation issues

We have implemented our algorithm using Matlab. The running times on a 2GHz Pentium 4 with 256Mb of memory are in Table 2.

Our algorithm takes 107.79 seconds for the Artichoke sequence, 143.59 seconds for the Cow sequence and 111.04 for the Hotel sequence. The *KLT part* refers to the time

Sequence	UKLT		KLT
	Total time	KLT part	
Artichoke Sequence	107.79	75%	21.26
Hotel Sequence	111.04	76%	24.62
Cow Sequence	143.59	79%	27.31

Table 2. Running times.

taken to propagate the feature sigma points. Note that the cost of the UKLT algorithm its higher, since it involves the tracking of five times more points. The standard KLT algorithm takes 21.26 seconds for the Artichoke Sequence, 27.31 seconds for the Cow sequence, and 24.62 second for the Hotel sequence. As the outlier rejection is done on-line, we do not need additional steps (and time) for this.

5. Conclusions

We have devised a method to estimate the random variable that undergoes a non-linear transformation, and combined this method with the KLT feature tracker. In this way, it is possible to use the standard KLT algorithm to propagate the associated uncertainty of each feature point. We emphasize that any other feature tracking algorithm could be used instead of KLT.

Our approach represents the feature locations as a GRVs, which enables us to propagate the uncertainty of the feature points. Using this representation, we know in what direction the error is high, making possible to use a more adequate error minimization measure. This information is important in a wide range of applications, such as bundle adjustment.

The results confirm that our method better discard features that should be considered outliers. This significantly improves the quality/accuracy of the tracking, and avoid large errors in high level tasks. The on-line outlier rejection avoids post-processing steps, increasing the process robustness. Future work includes the use of other trackers and the extension of the results to 3D algorithms.

Acknowledgements

The authors are grateful to Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq, for the financial support of this work.

References

[1] A. K. R. Chowdhury. *Statistical Analysis of 3D Modeling from Monocular Video Streams*. PhD thesis, University of Maryland, 2002.

[2] M. Fischler and R. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. In *Communications of the ACM*, volume 24(6), pages 381–395, 1981.

[3] A. Fusiello, E. Trucco, T. Tommasini, and V. Roberto. Improving feature tracking with robust statistics. *Pattern Analysis and Applications*, 2:312–320, 1999.

[4] S. Goldenstein. A gentle introduction to predictive filters. In *Revista de Informatica Teórica e Aplicada (RITA)*, volume XI (1), pages 61–89, Oct. 2004.

[5] G. Hager and P. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *PAMI*, 20:1025–1039, 1998.

[6] H. Jin, P. Favaro, and S. Soatto. Real-time feature tracking and outlier rejection with changes in illumination. In *ICCV*, pages 684–689, 2001.

[7] S. Julier and J. Uhlmann. A new extension of the kalman filter to nonlinear systems. In *In SPIE*, 1997.

[8] Y. Kanazawa and K. Kanatani. Do we really have to consider covariance matrices for image features? In *ICCV*, pages 301–306, 2001.

[9] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[10] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI81*, pages 674–679, 1981.

[11] Y. Ma, S. Soatto, J. Kosecka, and S. Sastry. *An Invitation to 3D Vision - From Images to Geometric Models*. Springer, 2004.

[12] J. Shi and C. Tomasi. Good features to track. In *CVPR*, pages 593–600, 1994.

[13] E. Simoncelli. *Handbook of Computer Vision and Applications*, volume II, chapter Bayesian Multi-scale Differential Optical Flow, pages 397–422. Academic Press, 1999.

[14] P. M. Steele and C. Jaynes. Feature uncertainty arising from covariant image noise. In *CVPR*, volume 1, pages 1063–1070, 2005.

[15] C. Tomasi and T. Kanade. Detection and tracking of point features. Technical Report CMU-CS-91-132, Carnegie Mellon University, April 1991.

[16] P. Torr, A. Zisserman, and M. S. Robust detection of degeneracy. In *ICCV*, pages 1037–1044, 1995.

[17] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment - a modern synthesis. In *Proceedings of the International Workshop on Vision Algorithms: Theory and Practice*, pages 298–372, 1999.

[18] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan. The unscented particle filter. Technical Report CUED/F-INFENG/TR380, Cambridge University, August 2000.

[19] E. Wan and R. van der Merwe. *Kalman Filtering and Neural Networks*. Wiley Publishing, 2001.

[20] Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. *IJCV*, 27(2):161–198, 1998.

[21] J. Zhu, S. Schwartz, and B. Liu. Object tracking: Feature selection and confidence propagation. In *CRV*, pages 18–21, 2004.