PFG 2013 / 4, 0323 – 0332
Stuttgart, August 2013

Article

A Robust Iterative Kalman Filter Based On Implicit Measurement Equations

Richard Steffen, Mülheim

Keywords: Kalman filter, implicit iterative update, estimation

Summary: In the field of photogrammetry, computer vision and robotics recursive estimation of time dependent processes is an important task. Usually Kalman filter based techniques are used which rely on explicit model functions that directly and explicitly describe the effect of the parameters on the observations. However, some problems naturally result in implicit constraints between the observations and the parameters, for instance all those resulting in homogeneous equation systems. By implicit we mean that the constraints are given by equations that are not easily solvable for the observation vector. We propose an iterative extended Kalman filter based on implicit measurement equations. The derived filter is useful for various applications, where the possibility to use implicit constraints simplifies the modeling. As an extension, we introduce a robustification technique similar to Ting et al. (2007) and Huber (1981) which down-weights the influence of potential outliers. The feasibility of the proposed framework is demonstrated at a number of typical computer vision applications.

Zusammenfassung: *Robuster iterativer Kalman-Filter mit implizierten Beobachtungsgleichungen.* In der Photogrammtrie, der Computer Vision und der Robotik finden rekursive Schätzungen ein weites Anwendungsspektrum. Üblicherweise werden in diesem Zusammenhang Kalman-Filter-basierte Techniken angewendet, welche auf expliziten Beobachtungsmodellen basieren, die den Effekt der Beobachtungen auf die Parameter direkt und explizit beschreiben. Einige Probleme sind jedoch aufgrund ihrer Natur als implizite Bedingungen zwischen den Parametern und den Beobachtungen formuliert, wie zum Beispiel Bedingungen unter Verwendung homogener Koordinaten. Unter impliziten Bedingungen verstehen wir Gleichungen, welche nicht trivial nach einem Beobachtungsvektor aufgelöst werden können. Diese Arbeit präsentiert einen iterativen erweiterten Kalman-Filter, welcher die Verwendung impliziter Beobachtungsgleichungen ermöglicht. Als eine Erweiterung führen wir ein Schema zur Robustifizierung nach Ting et al. (2007) and Huber (1981) ein, welches den Einfluss potenzieller Ausreißer reduziert. Die Nützlichkeit dieses Werkzeuges wird an einigen typischen Beispielen aus dem Bereich der Bildverarbeitung demonstriert.

1 Introduction

Recursive estimation and Kalman filtering is a classical technique (Kalman 1960) and has been widely used in robotics and computer vision (Welch & Bishop 1995) to tackle problems such as positioning, object reconstruction, object tracking or calibration tasks. So far, the applications of Kalman filter techniques was limited to problems where the observations are represented by an explicit function in the unknown parameters.

However, many problems encountered in computer vision naturally result in implicit constraints between the observations and the parameters (Hartley & Zisserman 2000, Heuel 2004, Perwass et al. 2005, Heuel 2001). An example is the iterative estimation of a 2d line from 2d point observations (section 3).

Although it is always possible to reduce the solution of an implicit problem to the solution of an explicit problem (Koch 1999, p.231ff) by introducing pseudo parameters, it is often much easier and straightforward to specify the measurement equations as implicit constraints relating the state vector to the observation vector.

DOI: 10.1127/1432-8364/2013/0180 1432-8364/13/0180 $ 2.50

The linear Kalman filter using explicit functions has been extended to the case of non-linear functions known as the extended Kalman filter (EKF) to deal with non-linear models and an iterative scheme (IEKF) to take into account the change of the linearization point. The only publication the author knows considering implicit constraints within Kalman filtering is SOATTO et al. (1994). However, the method only allows for a single iteration, equivalent to the classical EKF without taking into account the advantage of the IEKF.

The focus of this paper is on providing the foundations for integrating implicit constraints into a Kalman filter using the iterative scheme of the IEKF in order to enrich the application domain of the popular Kalman filter approach. As an enhancement to STEFFEN & BEDER (2007), we show that this novel framework is a generalization of the classical iterative Kalman filter and we demonstrate the benefit of the introduced approach for a variety of applications.

The traditional Kalman filter consists of two steps, a time update and a measurement update. Since the challenges for implicit constraints are in applying the measurement update, we mainly investigate the integration of implicit constraints to the update step.

Recently, the unscented Kalman filter (JULIER & UHLMANN 1997) has obtained a lot of attention which aims at improving the stochastic properties of the filter. Instead, our work aims at simplifying the specification of measurement equations instead.

This work is structured as follows: first we derive the prerequisites for the recursive estimation algorithm based on implicit measurement functions in section 2.1. Then we show, how outliers can be detected and the algorithms robustness can be improved. The final algorithm is given in section 2.3. Finally we present examples of computer vision tasks benefitting from the proposed algorithm in section 3. This comprises different tasks such as pose estimation, point cloud fitting and structure from motion.

2 Kalman Filter Update using Implicit Constraints

In this section we first derive the classical Kalman filter update using explicit measurement equations. Based on this concept, we show how to incorporate implicit constraints and adress the treatment of outliers. Section 2.3 provides a summary of the final algorithm.

2.1 *Update Estimation*

The Kalman filter consists of a dynamic model (prediction step) and a correction step (update step). The dynamic model is given by a non-linear function $\boldsymbol{h}$ which provides a predicted state vector $\bar{\boldsymbol{p}}$ by $\bar{\boldsymbol{p}} = \boldsymbol{h}(\boldsymbol{p}) + \epsilon$, with $\boldsymbol{p}$ as the previous state vector and ϵ as additional Gaussian random noise that influences the state vector. In addition, the state vectors uncertainty is given by its covariance matrix C_{pp} that has to be computed for the predicted state vector $\bar{\mathsf{C}}_{pp}$ by error propagation from the previous time step. After the prediction, we usually obtain measurements to update the predicted state vector. Note that this process can be interpreted as a weighted mean (2), (3) of the predicted state itself and the state implied by the measurements. The measurement equation is given by

$$\boldsymbol{z} + \boldsymbol{v} = \boldsymbol{f}(\boldsymbol{p}), \tag{1}$$

with $\boldsymbol{z}$ as the observation vector, $\boldsymbol{v}$ as the estimated residuals and $\boldsymbol{f}(\boldsymbol{p})$ as an explicit function of the state vector. In general, $\boldsymbol{f}$ is non-linear and we have to update the state vector iteratively by $\boldsymbol{p}^{\nu} = \boldsymbol{p}^{\nu-1} + \Delta\boldsymbol{p}^{\nu}$ until the convergence point has been reached ($\Delta\boldsymbol{p}^{\nu} = 0$). The index ν indicates the iteration counter. The update vector $\Delta\boldsymbol{p}^{\nu}$ can be obtained by an iterative maximum likelihood estimate using

$$\begin{aligned} \Delta\boldsymbol{p}^{\nu} &= \mathsf{F}\boldsymbol{v}^{\nu} \\ &= \mathsf{F}\left(\boldsymbol{z} - \boldsymbol{f}(\boldsymbol{p}^{\nu-1}) - \mathsf{A}(\bar{\boldsymbol{p}} - \boldsymbol{p}^{\nu-1})\right) \end{aligned} \tag{2}$$

with A as the Jacobian of $\boldsymbol{f}$ w.r.t. the updated state vector $\boldsymbol{p}^{\nu}$ and F as the influence or gain matrix by

$$\mathsf{F} = \bar{\mathsf{C}}_{pp}\mathsf{A}^{\mathsf{T}}(\mathsf{C}_{zz} + \mathsf{A}\bar{\mathsf{C}}_{pp}\mathsf{A}^{\mathsf{T}})^{-1}, \tag{3}$$

where C_{zz} denotes the covariance matrix of the observations. The last term $A(\bar{\boldsymbol{p}} - \boldsymbol{p}^{\nu-1})$ introduces the correction in the iterative extended Kalman filter (WELCH & BISHOP 1995). The covariance matrix of the estimated state vector can be obtained by the error propagation of (2) via

$$C_{pp} = (I - FA)\bar{C}_{pp}. \tag{4}$$

See WELCH & BISHOP (1995) or THRUN et al. (2001) for further details.

In the following, we derive a recursive estimation scheme for the case of non-linear implicit measurement constraints using the weighted mean idea. In complete analogy to the classical explicit Kalman filter we start with a parameter vector $\bar{\boldsymbol{p}}_1$ (the state vector) and its covariance matrix $\bar{C}_1$ resulting from some prediction step. This state shall be updated according to a newly acquired measurement vector $\boldsymbol{z}$ that implicitly constrains the parameter vector. By implicit we mean that the measurement model is given by a non-linear implicit function

$$\boldsymbol{g}(\boldsymbol{p}, \boldsymbol{z}) = \boldsymbol{0} \tag{5}$$

relating the unknown parameter vector $\boldsymbol{p}$ to the observation vector $\boldsymbol{z}$. Such an implicit observation model equation is often much easier to obtain than the explicit function $\boldsymbol{z} = \boldsymbol{f}(\boldsymbol{p})$ required by the classical Kalman filter. Note that every explicit function is easily made implicit by subtraction of the measurement vector $\boldsymbol{0} = \boldsymbol{g}(\boldsymbol{p}, \boldsymbol{z}) = \boldsymbol{f}(\boldsymbol{p}) - \boldsymbol{z}$.

We start by analyzing how a new parameter vector can be estimated from those observations alone by looking at the Taylor expansion of the observation model equation in (5)

$$\begin{aligned}\boldsymbol{0} &\approx \boldsymbol{g}(\boldsymbol{p}^{\nu}, \boldsymbol{z}^{\nu}) + A(\boldsymbol{p} - \boldsymbol{p}^{\nu}) + B^{\mathsf{T}}(\widehat{\boldsymbol{z}} - \boldsymbol{z}^{\nu}) \\ &= \boldsymbol{g}(\boldsymbol{p}^{\nu}, \boldsymbol{z}^{\nu}) + A\Delta\boldsymbol{p} + B^{\mathsf{T}}(\widehat{\boldsymbol{z}} - \boldsymbol{z} + \boldsymbol{z} - \boldsymbol{z}^{\nu})\end{aligned} \tag{6}$$

containing the Jacobians

$$A = \left.\frac{\partial \boldsymbol{g}(\boldsymbol{p}, \boldsymbol{z})}{\partial \boldsymbol{p}}\right|_{\boldsymbol{z}^{\nu}, \boldsymbol{p}^{\nu}} \quad B = \left.\frac{\partial \boldsymbol{g}(\boldsymbol{p}, \boldsymbol{z})^{\mathsf{T}}}{\partial \boldsymbol{z}}\right|_{\boldsymbol{z}^{\nu}, \boldsymbol{p}^{\nu}}. \tag{7}$$

By rearranging this equation, we obtain the linear (left) and non-linear (right) contradiction part

$$A\Delta\boldsymbol{p} + B^{\mathsf{T}}(\widehat{\boldsymbol{z}} - \boldsymbol{z}) = -\boldsymbol{g}(\boldsymbol{p}^{\nu}, \boldsymbol{z}^{\nu}) - B^{\mathsf{T}}(\boldsymbol{z} - \boldsymbol{z}^{\nu}), \tag{8}$$

which are equal at the convergence point with the final estimated observation vector $\widehat{\boldsymbol{z}}$. Given enough such observations, the maximum likelihood estimate of the parameter vector $\boldsymbol{p}$ is obtained by iteratively updating (FÖRSTNER & WROBEL 2004) similarly to the classical Kalman filter by

$$\boldsymbol{p}^{\nu} = \boldsymbol{p}^{\nu-1} + \Delta\boldsymbol{p}^{\nu}, \tag{9}$$

with the non-linear contradiction $\boldsymbol{c}_g$ in

$$\Delta\boldsymbol{p}^{\nu} = CA^{\mathsf{T}}(B^{\mathsf{T}}C_{zz}B)^{-1}\boldsymbol{c}_g^{\nu} \tag{10}$$

using the covariance matrix

$$C = (A^{\mathsf{T}}(B^{\mathsf{T}}C_{zz}B)^{-1}A)^{-1} \tag{11}$$

and the non-linear contradiction vector

$$\boldsymbol{c}_g^{\nu} = -\boldsymbol{g}(\boldsymbol{p}^{\nu-1}, \boldsymbol{z}^{\nu-1}) - B^{\mathsf{T}}(\boldsymbol{z} - \boldsymbol{z}^{\nu-1}). \tag{12}$$

We can also compute the residuals of the observations (MIKHAIL & ACKERMANN 1976)

$$\begin{aligned}\boldsymbol{v}^{\nu} &= \widehat{\boldsymbol{z}} - \boldsymbol{z} \\ &= C_{zz}B(B^{\mathsf{T}}C_{zz}B)^{-1}(\boldsymbol{c}_g^{\nu} - A\Delta\boldsymbol{p}^{\nu})\end{aligned} \tag{13}$$

yielding the linearization point for the next iteration $\boldsymbol{z}^{\nu+1} = \boldsymbol{z} + \boldsymbol{v}^{\nu}$ and $\boldsymbol{p}^{\nu}$ from (9).

This estimation scheme for the computation of a parameter vector from a given observation set using implicit constraints is also known as the Gauss-Helmert model. Now we combine this estimation scheme with the state vector from the prediction step to achieve an iterative recursive update. To do so, we interpret the predicted state vector $\bar{\boldsymbol{p}}$ as a direct observation $\boldsymbol{z}_1$ of the new state vector, which fits into the above estimation scheme using the model equation

$$\boldsymbol{0} = \boldsymbol{g}_1(\boldsymbol{p}, \boldsymbol{z}_1) = \boldsymbol{p} - \boldsymbol{z}_1 \tag{14}$$

and the observations $\boldsymbol{z}_1 = \bar{\boldsymbol{p}}$ having the covariance matrix $C_{zz_1} = C_{pp_1} = \bar{C}_{pp}$. Because this constraint is linear, the Jacobians are in this case simply $A_1 = I$ and $B_1 = -I$ and independent of the linearization point. Considering the prediction of the state vector alone, we would obtain $\boldsymbol{p} = \bar{\boldsymbol{p}}$ so that $\boldsymbol{c}_{g_1} = \mathbf{0}$. As the measurement update is supposed to influence the state vector and thereby the joint linearization point, we have to cope with the change of the contradiction incurred by this change, which we call $\Delta\boldsymbol{c}_{g_1}$ in the following to reflect this important property. Plugging the direct measurement (14) and its Jacobians into (12) yields the contradiction

$$\Delta\boldsymbol{c}_{g_1} = \bar{\boldsymbol{p}} - \boldsymbol{p}^{\nu} \tag{15}$$

because $\boldsymbol{g}_1(\boldsymbol{p}^{\nu}, \boldsymbol{z}_1^{\nu}) = \mathbf{0}$ is immediately fulfilled for the direct observation. Therefore, also the equation

$$C_{pp_1}^{-1}\Delta\boldsymbol{p}_1 = C_{pp_1}^{-1}\Delta\boldsymbol{c}_{g_1} \tag{16}$$

holds, which will become useful in the following.

We are now ready to formulate the recursive estimation as a weighted mean process of two variables being the predicted state $\boldsymbol{p}_1$ on the one hand and the estimated state from the novel observations $\boldsymbol{p}_2$ on the other hand. Hence, the state update is given by

$$\Delta\boldsymbol{p} = (C_{pp_1}^{-1} + C_{pp_2}^{-1})^{-1}(C_{pp_1}^{-1}\Delta\boldsymbol{p}_1 + C_{pp_2}^{-1}\Delta\boldsymbol{p}_2). \tag{17}$$

Substituting (16) and (10) into this weighted mean update, we obtain

$$\Delta\boldsymbol{p} = \overbrace{(C_{pp_1}^{-1} + C_{pp_2}^{-1})^{-1}}^{C_{pp}} (C_{pp_1}^{-1}\Delta\boldsymbol{c}_{g_1} + A_2^{\mathsf{T}}(B_2^{\mathsf{T}} C_{zz} B_2)^{-1}\boldsymbol{c}_{g_2}). \tag{18}$$

Using the well known matrix inversion (WOODBURY 1950) identity

$$(K + LN^{-1}M)^{-1} = K^{-1} - K^{-1}L(N + MK^{-1}L)^{-1}MK^{-1},$$

we can reformulate (18) and finally get

$$\Delta\boldsymbol{p}^{\nu} = F\boldsymbol{c}_{g_2} + (I - FA_2)\Delta\boldsymbol{c}_{g1}, \tag{19}$$

with the substitution

$$F = C_{pp_1}A_2^{\mathsf{T}}(B_2^{\mathsf{T}} C_{zz} B_2 + A_2 C_{pp_1} A_2^{\mathsf{T}})^{-1}, \tag{20}$$

yielding the iterative update $\boldsymbol{p}^{\nu+1} = \boldsymbol{p}^{\nu} + \Delta\boldsymbol{p}^{\nu}$. The residuals are computed using (13)

$$\boldsymbol{v}_1 = -\Delta\boldsymbol{c}_{g_1} + \Delta\boldsymbol{p}^{\nu} \tag{21}$$

$$\boldsymbol{v}_2 = C_{zz}B_2(B_2^{\mathsf{T}} C_{zz} B_2)^{-1}(\boldsymbol{c}_{g_2} - A_2\Delta\boldsymbol{p}^{\nu}) \tag{22}$$

allowing to compute the contradiction for the next iteration

$$\boldsymbol{c}_{g_2} = -\boldsymbol{g}_2(\boldsymbol{p}^{\nu}, \boldsymbol{z}_2 + \boldsymbol{v}_2) + B_2^{\mathsf{T}}\boldsymbol{v}_2. \tag{23}$$

Finally, note that the new covariance matrix of the state vector is given by

$$C_{pp} = (I - FA_2)\bar{C}_{pp}. \tag{24}$$

As already mentioned, the explicit model of the classical Kalman filter can be transformed to an implicit model into the form $\mathbf{0} = \boldsymbol{f}(\boldsymbol{p}) - \boldsymbol{z}$. Applying this to the novel framework, the Jacobian w.r.t. the observation becomes B_2 becomes the negative identity matrix $-I$. Therefore, the gain matrix in (20) boils down to the classical gain matrix in (3). The contradiction in (23) becomes

$$\begin{aligned} \boldsymbol{c}_{g_2} &= -(\boldsymbol{f}(\boldsymbol{p}) - \boldsymbol{z} + \boldsymbol{v}) + \boldsymbol{v} \\ \boldsymbol{c}_{g_2} &= \boldsymbol{z} - \boldsymbol{f}(\boldsymbol{p}) \end{aligned} \tag{25}$$

Substitute c_{g_2} and (15) into the update equation (19) we obtain

$$\begin{aligned} \Delta\boldsymbol{p}^{\nu} &= F(\boldsymbol{z} - \boldsymbol{f}(\boldsymbol{p}^{\nu-1})) \\ &\quad + (I - FA_2)(\bar{\boldsymbol{p}} - \boldsymbol{p}^{\nu-1}) \\ &= (\bar{\boldsymbol{p}} - \boldsymbol{p}^{\nu}) \\ &\quad + F\left(\boldsymbol{z} - \boldsymbol{f}(\boldsymbol{p}^{\nu-1}) - A_2(\bar{\boldsymbol{p}} - \boldsymbol{p}^{\nu-1})\right) \end{aligned}$$

as the iterative update to the approximate values $\boldsymbol{p}^{\nu} = \boldsymbol{p}^{\nu-1} + \Delta\boldsymbol{p}^{\nu}$. To be equal to the classical iterative Kalman filter, this can be rewritten into (2) using the predicted state at $\nu = 0$ as the reference state. As a conclusion, the classical iterative Kalman filter is just a specialization of the novel framework derived here. Therefore, there is no need to compare the novel Kalman filter with the classical Kalman filter as from theory both are equal. A full proof can be found in (STEFFEN 2009).

The presented algorithm is based on a least squares optimization, which is known to be very sensitive to outliers. In the following section, we show how the robustness of the presented method can be increased by re-weighting the observations.

2.2 *Robustification by Re-Weighting*

The classical Kalman filter as well as the estimation scheme presented so far minimizes the squared residuals of the observations, which is known to be sensitive to outliers. We now show how outliers may be detected by considering the plausibility of the computed residuals with respect to the expected uncertainty. By reducing the influence of such observations on the estimation, the robustness can be increased.

The weighted mean process is mainly influenced by two error effects. First, an erroneous dynamic model results in an erroneous prediction. Second, noisy observations yield a correction effect to the estimated state.

In TING et al. (2007) a robust outlier detection is presented for the classical Kalman filter. We will adapt this technique using an alternative re-weighting method proposed in HUBER (1981). Assuming an error free prediction, the update of the observations in $\boldsymbol{v}_2$ is normal distributed with zero mean. In this case, we are able to detect outliers by simply normalizing v_2 with the inverse observation covariance $\boldsymbol{C}_{zz}$ and reweight the observations accordingly.

However, in realistic applications the prediction model does not always hold true. Its effect on the improvement of the observations in $\boldsymbol{v}_2$ can not be modeled in general and depends on the system noise of the dynamic model. For instance, in the structure-from-motion problem an error in the camera position orthogonal to the viewing direction results in a consistent translation fraction in the image coordinates.

One common way to solve this problem is to approximate the complex deformation of the estimated observation $\widehat{\boldsymbol{z}} = \boldsymbol{z} + \boldsymbol{v}$. This can be done by choosing an approximation function depending on the expected deformations. In the case of image observations, a homography could be a good choice. The robust estimation of this function and the detection of the outliers can then be done by a RANSAC based approach or by a robustified least square solution. However, such a procedure for outlier detection is often quite expensive.

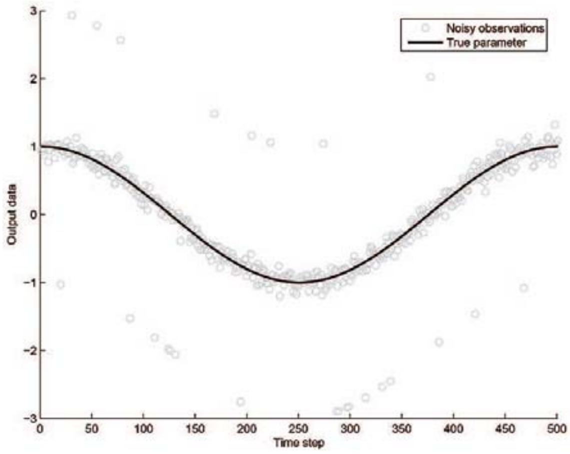

Fig. 1: Full cosine wavelength 2π, sampled with 500 samples, noise is 0.05, system noise 0.01, 5 percent outliers with strength of 2, iteration to convergence.

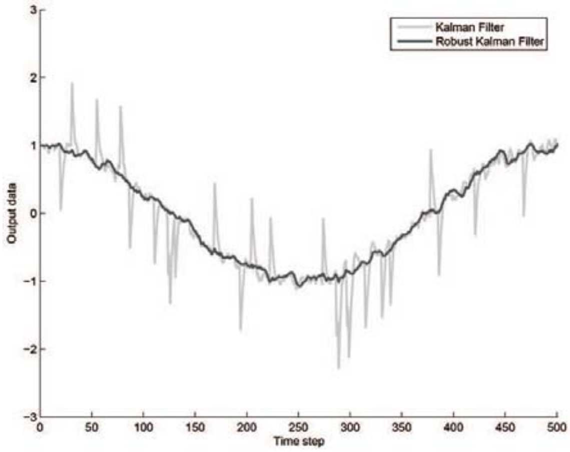

Fig. 2: Recursive estimation using the non-robustified and the robustified version of the Kalman filter.

From another point of view, the influence of the erroneous prediction is small if the system noise is large enough to compensate for the prediction error, which should be the case for a well approximating dynamic model. Thus, we are able to robustify the update by reweighting the observations in the following sense.

We first normalize the residuals $\boldsymbol{v}_{2_j}$ with the uncorrelated observation standard deviation to get standard normal distributed test values

$$c_j = \frac{v_{2_j}}{\sigma_j}. \tag{26}$$

One can argue that the normalization should be done using σ_v from C_{vv}. However, in the case that all observations can be assumed to have the same influence (equal redundancy number) to the parameter vector, it is suitable to use σ_z instead. The absolute test values c_j allow to decide for each single observation, whether there is a reason to consider it as an outlier. We then compute a variance factor w_j for each observation according to HUBER (1981)

$$w_j = \begin{cases} 1 & \text{if } \|c_j\| \leq k \\ \frac{\|c_j\|}{k} & \text{if } \|c_j\| > k \end{cases} , \qquad (27)$$

which does not alter the observations within the range of k times the expected standard deviation and reduces the effect of observations outside this range on the estimation. To perform the desired re-weighting, we use the observation covariance matrix

$$C_{zz}^{(\nu)} = diag(\boldsymbol{w}) C_{zz}^{(0)} \qquad (28)$$

instead of the initially given covariance matrix $C_{zz}^{(0)}$ in each iteration.

Following the experimental validation of TING et al. (2007), we also demonstrate the robustification on the one dimensional estimation of a cosinus curve containing some outliers. In Fig. 1 the noisy observations with 5% of outliers are shown. Fig. 2 shows the non robust and the robust version of the estimated curve parameters. The robustification yields a much smoother estimate not being perturbed by the outliers.

2.3 *The final Algorithm*

We now summarize the recursive estimation algorithm. From a previous estimation or prediction step of the filter, a current state vector $\boldsymbol{p}_1$ together with its covariance C_{11} is known. We gather additional observations $\boldsymbol{z}_2$ together with their covariance matrix C_{22} in a subsequent measurement step. The following algorithm may then be applied to update the state vector accordingly

1. set $\Delta\boldsymbol{p} = \boldsymbol{0}$
2. set $\boldsymbol{p} = \bar{\boldsymbol{p}}$
3. set $\boldsymbol{v}_1 = \boldsymbol{0}$
4. set $\boldsymbol{v}_2 = \boldsymbol{0}$, hence $\boldsymbol{z}^0 = \boldsymbol{z}$
5. Iterate until $\Delta\boldsymbol{p}^\nu$ is sufficiently small
   (a) compute Jacobians A_2 and B_2 at $\boldsymbol{p}^\nu$ and $\boldsymbol{z}^\nu$
   (b) compute the gain matrix F according to (20)
   (c) compute $\boldsymbol{c}_{g_2}$ according to (23)
   (d) compute $\Delta\boldsymbol{c}_{g_1}$ according to (15)
   (e) compute $\Delta\boldsymbol{p}^\nu$ according to (19)
   (f) update $\boldsymbol{p}^\nu$ with $\Delta\boldsymbol{p}^\nu$
   (g) compute $\boldsymbol{v}_1$ according to (21)
   (h) compute $\boldsymbol{v}_2$ according to (22)
   (i) update $\boldsymbol{z}^\nu = \boldsymbol{z} + \boldsymbol{v}_2$
   (j) compute normalized test values according to (26)
   (k) compute variance factor for all observations with (27)
   (l) compute reweighted observation covariance matrix for the next iteration
6. compute C_{pp} according to (24) .

After the algorithm is converged, we finally obtain the updated state vector $\boldsymbol{p}$ together with its covariance matrix C_{pp}. The only problem specific part is the computation of the Jacobians in step 5a, which has to be adapted. This completes the measurement update using the implicit constraint and a subsequent time update may be performed. Also note that for implicit measurement equations obtained directly from explicit equations by subtraction, the presented algorithm yields the same results as the classical iterated extended Kalman filter.

3 Exemplary Applications

In this section we present some examples for the usefulness of the proposed algorithm. Given that theoretically both our proposed framework and the traditional Kalman filter are equivalent, we do not provide a numerical example. More precisely, the results of the proposed algorithm is equivalent to the results of the traditional Kalman filters when using the explicit constraints as well when using the pseudo parameterization. Our motivation is to demonstrate the straightforwardness using implicit measurement equations for Kalman filter based estimation. Note, the observation in a Kalman filter must be minimal represented. For instance, using homogeneous observations lead to a singular covariance matrix of the observations and the Kalman filter update will fail.

3.1 *Line and Plane Estimation*

The recursive estimation of basic primitives from an uncertain point cloud is useful in a lot of applications, e.g. in the task of urban scene reconstruction (POLLEFEYS et al. 2008). In this section we demonstrate solutions for a recursive 2d line and a recursive 3d plane estimation. A 2d line can be represented given a set of two parameters $\boldsymbol{l} = \{a, b\}$. For every Euclidean point $\boldsymbol{x}_i = \{x_i, y_i\}$ the incidence with the line $\boldsymbol{l}$ is given by

$$y_i = ax_i + b. \tag{29}$$

Note, we are not able to reformulate this equation for both observations x_i and y_i explitely, but we can easily rewrite (29) in the implicit form

$$0 = g(\boldsymbol{p}, \boldsymbol{z}) = ax_i + b - y_i. \tag{30}$$

Using our proposed method to estimate the line parameter recursively with (30) as measurement update equation, the state vector contains $\boldsymbol{p} = [a, b]^\mathsf{T}$ and the observation vector the observed 2d points $\boldsymbol{z} = [x_i, y_i]^\mathsf{T}$. To handle vertical lines, (30) should be replaced by $0 = \mathbf{l}^\mathsf{T}\mathbf{x}$ with $\mathbf{l}$ as the homogeneous line with the unknown parameter d (distance from origin) and ϕ (angle of the line) and $\mathbf{x}$ as the homogeneous 2d point.

In a similar way, we are able to recursively estimate a plane which approximates a given point cloud. The incidence of an homogeneous 3d point $\mathbf{X}_i = [X_i, Y_i, Z_i, 1]^\mathsf{T}$ and a plane $\mathbf{A} = [\boldsymbol{n}, -d]^\mathsf{T}$ can be expressed as a simple bilinear constraint (Fig. 3).

$$0 = \mathbf{A}^\mathsf{T}\mathbf{X}_i. \tag{31}$$

Again, this constraint can not be easily reformulated to an explicit observation measurement function.

3.2 *Pose Estimation with parameterizable Shapes*

Pose estimation from a single camera observation is another interesting application. In this section, we give an example for the pose estimation of a simple sphere. In the subsequent section, we extend this approach to a more general formulation.

Assume that we are able to observe the silhouette of a projected sphere with a known radius (Fig. 4). For every point $\mathbf{x}_i$ on the silhouette we can formulate an implicit measurement equation in the following way:

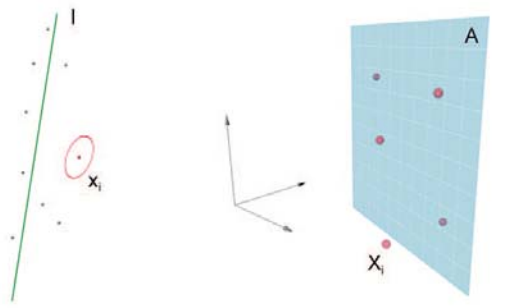

Fig. 3: Left: Recursive 2d line estimation from uncertain 2d points. Right: Recursive 3d Plane estimation from an uncertain 3d point cloud.

The projection ray $\mathbf{L}_i = [\mathbf{L}^h, \mathbf{L}^0]^\mathsf{T}$ in Pluecker representation can be obtained using a known camera orientation by

$$\mathbf{L}_i = \bar{\mathsf{P}}_L^\mathsf{T}\mathbf{x}_i = \begin{bmatrix} \mathbf{L}^h \\ \mathbf{L}^0 \end{bmatrix} = \begin{bmatrix} \mathsf{C} \\ \mathsf{D} \end{bmatrix} \mathbf{x}_i \tag{32}$$

The inverse projection matrix $\bar{\mathsf{P}}_L$ can be computed as a function of the camera orientation and its calibration (HEUEL 2004). The distance between the projection ray of the silhouette points $\boldsymbol{X}_i$ and the center of the sphere $\boldsymbol{X}_c$ has to be the known radius R. We can express this constraint implicitly using $S(\bullet)$ as the skew matrix of a vector by

$$\begin{aligned} 0 &= \frac{\|S(\mathbf{L}_i^h)\boldsymbol{X}_c + \mathbf{L}_i^0\|}{\|\mathbf{L}_i^h\|} - R \\ &= \frac{S(C\mathbf{x}_i)\boldsymbol{X}_c + D\mathbf{x}_i}{\|C\mathbf{x}_i\|} - R. \end{aligned} \tag{33}$$

Estimating the center of a moving sphere observing the projected silhouette and assuming a linear motion model, the state vector contains the spheres center and its velocity. The observation vector contains the observed Euclidean silhouette image coordinates. The parameters C, D and R are assumed to be known and constant.

3.3 *Pose Estimation with meshed Shapes*

Pose estimation from a single camera often deals with known shapes given as 3d meshes,

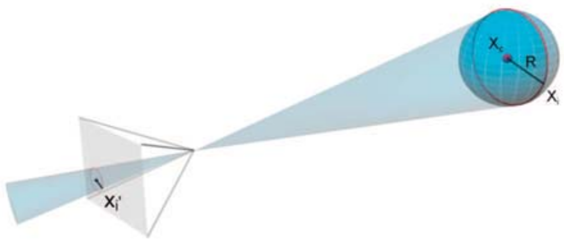

Fig. 4: Recursive 3d pose estimation of a sphere with known radius using observed silhouette points in an image.

(Rosenhahn et al. 2004). Extracting the silhouette in the projected image of the observed object, we can use an edge (from the mesh) to point (from the silhouette) incidence as the measurement equation, similar to the well known ICP algorithm.

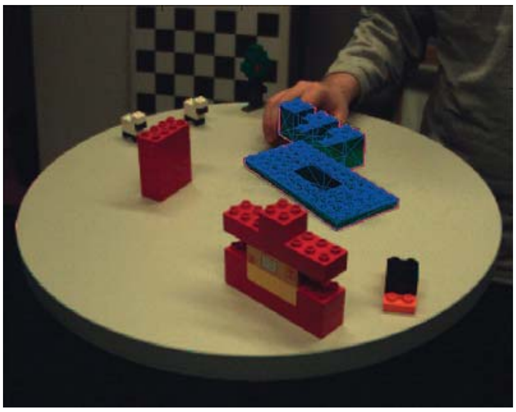

Fig. 5: Recursive 3d pose estimation with a single camera of a known object. The object is represented as a meshed shape with observed silhouette points in an image (Image and silhouette data provided by B. Rosenhahn, MPI).

The pose of an arbitrary 3d object can be represented by its Euclidean position $\boldsymbol{r}$ and its orientation $\mathbf{q}$ represented as quaternion. Position and orientation are combined in the motion matrix

$$\mathsf{M} = \mathsf{M}(\boldsymbol{r}, \mathbf{q}) = \begin{bmatrix} R(\mathbf{q}) & \boldsymbol{r} \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix}. \tag{34}$$

Applying the motion to the homogeneous 3d mesh point coordinates $\mathbf{X}_i$ given in a local object system and projecting them into the image space, we get the image coordinates of the projected mesh points with

$$\mathbf{x}_i = \mathsf{PM}\mathbf{X}_i. \tag{35}$$

Mesh edges are defined by two connected points $\{a, b\}$. The congruent line in the image space can be obtained by

$$\mathbf{l}_{ab} = S(\mathbf{x}_a)\mathbf{x}_b. \tag{36}$$

The incidence of the observed silhouette point $\mathbf{x}'$ and this projected edge is given by a bilinear contraint

$$0 = \mathbf{l}_{ab}^\mathsf{T}\mathbf{x}'. \tag{37}$$

Substitute (35) and (36) into (37) we get the final contraint

$$0 = \left(S(\mathsf{PM}\mathbf{X}_a)\mathsf{PM}\mathbf{X}_b\right)^\mathsf{T}\mathbf{x}'. \tag{38}$$

Using our recursive update model, the state vector contains the position and orientation of the 3d object and its velocities assuming a linear prediction model given by

$$\boldsymbol{p} = [\boldsymbol{r} \quad \boldsymbol{q} \quad \dot{\boldsymbol{r}} \quad \dot{\boldsymbol{q}}]^\mathsf{T}. \tag{39}$$

The measurement vector contains the observed Euclidean 2d silhouette points.

3.4 *Artificial Horizon Estimation*

In the field of robotics and navigation, an artificial horizon identification can be useful to stabilize the system over a long period of time as shown in Neto et al. (2011). In case of real measurements, the horizon line measurement is uncertain. This leads to undesirable effects in subsequent processes. In this case, the first choice is a Kalman filter based smoothing.

Using an implicit measurement update equation leads to an astonishling simple solution. The reference horizon in a normalized camera coordinate system can be expressed as a line in homogeneous representation by

$$\mathbf{l}^o = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}.$$

We denote the desired pitch by ω and roll by κ. In homogeneous representation, we use the simple measurement equation by the incidence

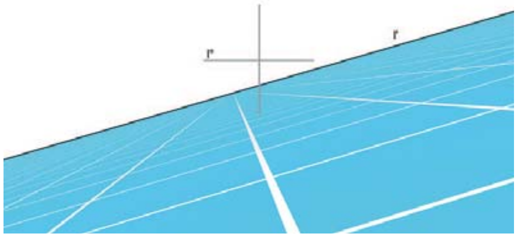

Fig. 6: Artificial horizon estimation from a single horizon line.

formulation

$$\mathbf{0} = S(\mathbf{l}) R_\kappa R_\omega \mathbf{l}^o. \quad (40)$$

Again, using our recursive update model the state vector contains the roll and pitch angles and their velocities by $\boldsymbol{p} = [\kappa \quad \omega \quad \dot\kappa \quad \dot\omega]^\mathsf{T}$. The measurement vector contains a line representation, for instance by angle and distance from origin (*hessian normal form*) or by *gradient and intercept* $\{a, b\}$, (29).

3.5 *Structure from Motion with Points and Lines*

Following the approach of DAVISON (2003), the motion of a single camera can be described by the following state vector

$$\boldsymbol{p} = [\boldsymbol{r} \quad \boldsymbol{q} \quad \dot{\boldsymbol{r}} \quad \dot{\boldsymbol{q}} \quad \boldsymbol{X}_1 \dots \boldsymbol{X}_i \quad \mathbf{L}_1 \dots \mathbf{L}_j]^\mathsf{T} \quad (41)$$

comprising the camera state followed by a set of feature parameters. The camera trajectory is represented by its actual position $\boldsymbol{r}$, its orientation quaternion $\mathbf{q}$, its velocity vector $\dot{\boldsymbol{r}}$ and its angular velocity vector $\dot{\boldsymbol{q}}$. The 3d point coordinates are represented by their Euclidean points $\boldsymbol{X}_i$. Additionally we introduce 3d lines represented by its Pluecker coordinates $\mathbf{L}_i$. The interior camera parameters are assumed to be known.

We assume a linear time update model. In the approach of DAVISON (2003) the measurement model for object points is based on the co-linearity equations, which can be written as homogeneous equations

$$\mathbf{x}_i = \mathsf{P} X_i \quad \text{with} \quad \mathsf{P} = \mathsf{K} R(\mathbf{q}) \left[I_{3\times3} | -\boldsymbol{r}\right]. \quad (42)$$

As our approach is able to cope with implicit functions, we formulate the co-linearity constraint using the cross-product such that the co-linearity equations can be stated as an implicit equation

$$S(\mathbf{x}_i)\mathsf{P}\mathbf{X}_i = -S(\mathsf{P}\mathbf{X}_i)\mathbf{x}_i = \mathbf{0}. \quad (43)$$

Obviously, those implicit constraints are equivalent to the explicit constraints used in DAVISON (2003). Also observe that they are non-linear in the camera pose parameters. Using only the point observations in our filter, the results are equal to the results achieved using a classical iterative extended Kalman filter.

Now let us incorporate the lines in the same way. The co-linearity equations for lines can be represented in homogeneous coordinates by

$$\mathbf{l}_j = \mathsf{P}_L \mathbf{L}_j \quad \text{with} \quad \underset{3\times6}{\mathsf{P}_L} = \mathsf{P}_L(\mathsf{P}), \quad (44)$$

where P_L can be computed as a function of the camera orientation and its calibration (HEUEL 2004). This constraint can not be easily expressed as non homogeneous observations in an explicit formulation. Using implicit constraints we can reformulate (44) in the same easy way to

$$S(\mathbf{l}_j)\mathsf{P}_L\mathbf{L}_j = -S(\mathsf{P}_L\mathbf{L}_j)\mathbf{l}_j = \mathbf{0}. \quad (45)$$

Using two constrains for points in (43) and two constrains for lines in (45) for every image point and line as an observation, we can solve the Kalman filter based update combining points and lines in one update step.

4 Conclusion

We presented a novel derivation of a recursive estimation framework in a Kalman filter approach which allows us to use implicit measurement constraint equations rather than being restricted to explicit ones. By using implicit constraints, the task of modeling recursive estimation schemes is eased significantly. Furthermore, we presented an improvement to the framework in order to deal with outliers in the observations. Instead of the elimination of observations, we used a re-weighting method.

We demonstrated the usefulness of this new algorithm on exemplary classical computer vision tasks.

The presented method is applicable to a broad range of time driven estimation problems, especially including all those resulting in homogeneous equation systems.

References

Davison, A.J., 2003: Real-time simultaneous localisation and mapping with a single camera. – 9th International Conference on Computer Vision: 674–679.

Förstner, W. & Wrobel, B., 2004: Mathematical concepts in photogrammetry. – McGlone, J.C., Mikhail, E.M. & Bethel, J. (eds.): Manual of Photogrammetry, ASPRS: 15–180.

Hartley, R. & Zisserman, A., 2000: Multiple View Geometry in Computer Vision. – Cambridge University Press.

Heuel, S., 2001: Points, lines and planes and their optimal estimation. – Pattern Recognition, 23rd DAGM Symposium, LNCS **2191**: 92–99, Springer.

Heuel, S., 2004: Uncertain Projective Geometry - Statistical Reasoning for Polyhedral Object Reconstruction. – Springer.

Huber, P., 1981: Robust Statistics. – Wiley, New York, USA.

Julier, S. & Uhlmann, J., 1997: A new extension of the Kalman filter to nonlinear systems. – International Symposium Aerospace/Defense Sensing, Simulations and Controls, Orlando, FL, USA.

Kalman, R.E., 1960: A new approach to linear filtering and prediction problems. – Journal of Basic Engineering: 35–45.

Koch, K.-R., 1999: Parameter Estimation and Hypothesis Testing in Linear Models. – 2nd edn., Springer.

Mikhail, E.M. & Ackermann, F., 1976: Observations and Least Squares. – University Press of America.

Neto, A.D.M., Victorino, A.C., Fantoni, I. & Zampieri, D.E., 2011: Robust horizon finding algorithm for real-time autonomous navigation based on monocular vision. – 14th International IEEE Conference on Intelligent Transportation Systems, ITSC 2011.

Perwass, C., Gebken, C. & Sommer, G., 2005: Estimation of geometric entities and operators from uncertain data. – 27. Symposium für Mustererkennung, DAGM 2005, Wien, LNCS: 459–467 Springer-Verlag, Berlin, Heidelberg.

Pollefeys, M., Nistér, D., Frahm, J.-M., Akbarzadeh, A., Mordohai, P., Clipp, B., Engels, C., Gallup, D., Kim, S.J., Merrell, P., Salmi, C., Sinha, S.N., Talton, B., Wang, L., Yang, Q., Stewénius, H., Yang, R., Welch, G. & Towles, H., 2008: Detailed real-time urban 3d reconstruction from video. – International Journal of Computer Vision **78** (2-3): 143–167.

Rosenhahn, B., Perwass, C. & Sommer, G., 2004: Pose estimation of 3d free-form contours: **62**: 267–289.

Soatto, S., Frezza, R. & Perona, P., 1994: Motion estimation on the essential manifold. – ECCV '94: Third European Conference-Volume II on Computer Vision: 61–72, Springer-Verlag, London, UK.

Steffen, R., 2009: Visual SLAM from image sequences acquired by unmanned aerial vehicles. – PhD thesis, Institute of Photogrammetry, University of Bonn.

Steffen, R. & Beder, C., 2007: Recursive estimation with implicit constraints. – Hamprecht, F., Schnör, C. & Jähne, B. (eds.): DAGM 2007, LNCS **4713**: 194–203, Springer.

Thrun, S., Burgard, W. & Fox, D., 2001: – Probabilistic Robotics (Intelligent Robotics and Autonomous Agents).

Ting, J., Theodorou, E. & Schaal, S., 2007: A kalman filter for robust outlier detection. – IEEE International Conference on Intelligent Robotics Systems.

Welch, G. & Bishop, G., 1995: An introduction to the Kalman filter. – Technical report. University of North Carolina at Chapel Hill, Chapel Hill, NC, USA.

Woodbury, M., 1950: Inverting modified matrices. – Memorandum 42, Statistics Research Group Princeton, New Jersey, USA.

Address of the Author:

Dr.-Ing. Richard Steffen, Löhberg 78, D-45468 Mülheim a.d. Ruhr, Tel: +49-208-4500025, e-mail: rsteffen@messbild.de

Manuskript eingereicht: Februar 2013
Angenommen: Mai 2013