ISSN: 1803-3814

EFFECTS OF ENVIRONMENT MODEL PARAMETRIZATION ON PHOTOGRAMMETRY RECONSTRUCTION

Jan Klecka, Ondrej Bostik

Brno University of Technology
Department of Control and Instrumentation
Technicka 12, Brno, 61600
Czech republic
klecka@feec.vutbr.cz

Abstract: *This paper is aimed at a description of effects which have assumptions of specific environment structure on quality of recurrently conducted photogrammetry reconstruction. The theoretical part covers the description of three different assumptions of environment structure and mathematical derivation of two suitable recurrent estimators: one based on Extended Kalman filter and the second one based on Maximum likelihood method. The experimental part is introducing simple virtual environment which is observed by linear camera model and then reconstructed using predefined algorithms and assumptions.*

Keywords: *Photogrammetry, 3D reconstruction, Multiple view reconstruction, recurrent estimation, maximum likelihood, Extended Kalman filter*

1 Introduction

Camera sensors and generally the real-time visual information are nowadays widely utilized way to implement feedback to control systems which would be, by processing other types of sensor data, controllable with serious difficulties or even completely uncontrollable. For example [4,5].

However, some areas of image processing are used as feedback rarely. Specifically, in the scope of this paper, we are dealing with unguided photogrammetry reconstruction. So processing image sequences captured from the camera (or even multiple cameras) which has been moved through the unknown environment in order to estimate both camera poses and geometrical structure of the observed environment. This is process still almost exclusively used as an instrument of offline processing of previously captured sequences into results like 3D visualizations, initial inputs for 3D modelling etc. and it is very rarely exploited as a real-time feedback Although there are fields whose applications can certainly benefit from such feedback as for example automatic optic inspection [5] or autonomous robotic navigation [10].

However, implementation of common photogrammetry methods into feedback struggles with a fact that photogrammetry tasks are usually solved by non-recurrent global optimization algorithm which is in this context called either Bundle Adjustment (BA) [1,2] or graph-based approach to simultaneous localization and mapping problem (SLAM) [3,11]. Such solutions have three major issues preventing them from becoming good feedback. Firstly, it will probably have not negligible transport delay on account that BA leads from its essence to high dimensional optimization problem and moreover, the main processing can start only after capturing the whole image sequence. Secondly, it is prone to significant estimation errors caused by easy-to-occur but uneasy-to-detect proximity to the singularity. And lastly, the resulting environment model would not be well suited for being a reliable feedback, because standardly used techniques usually leads to point cloud model [8] and processing it into understandable information will increase the above-mentioned transport delay.

To eliminate these problems, we working on the development of a recurrent estimator capable of working with photogrammetry tasks. The recurrence should suppress the transport delay by processing large proportion of observations immediately after their capture and moreover, it will bring some partial results during the capturing process which can be necessary to some feedbacked systems. However, if such recurrent algorithm should make sense and be distinct from the simple sequential application of BA it has to marginalize out some unimportant pieces of information (in our realization the old camera poses) to keep problem dimensionality low enough. From graph optimization terminology, some nodes have to be removed and their edges have to be recomputed into edges between remaining nodes. From this marginalization process is coming out the main realization challenge because if it should be handled properly the marginalized node edges have to fulfill some uneasily definable conditions (shortly parametrization of error function used by remaining edges have to be able properly approximate the marginalized linkages). And this is the first goal of this paper, we are trying to experimentally prove that with increasing number edges to environment nodes the camera pose nodes are coming closer to meet these conditions.

The second thing we want to explore is what effect will have different environment parametrization on these conditions. Because as we can see for example in [6,7] the environment structure does not have to always be represented by a point set and by exploiting assumption that the environment can be represented by some lower dimensional parametrization (for example that is composed strictly of planar elements [9]) we can reduce problem dimensionality for the price of adding additional nonlinearity to the edges which are connecting the environment and pose nodes. This can have actually two results either the additional nonlinearity makes the marginalization more difficult or the lower dimensionality will cause central limit theorem to be basically valid sooner and on the contrary makes the marginalization easier.

1.1 Used symbols and notation

For maximal clarity of following descriptions, we shortly specify used notation. The bold lower case symbols represent vectors e.g. $\mathbf{m}$ which are always column (row vectors are given with transposition), the bold upper case represents matrices e.g. $\mathbf{X}$. Estimations are identified by 'a hat' over the estimated variable symbol e.g. $\hat{\mathbf{m}}$ is an estimate of $\mathbf{m}$ Lower index containing i is attached to variables which vary during the mapping process e.g. $\mathbf{x}_i$.

2 Problem Formulation

From a probabilistic point of view, a solution to estimation problem is a definition of probability to every possible value of estimated variables vector to be its 'true' value given the processed observations. This is, in general, a problem of defining conditional probability function which can be simply expressed using Bayes formula. Applying this formula to non-recurrent photogrammetric reconstruction problem (bundle adjustment) results, after taking assumptions that observations are mutually independent and so as the camera poses, results in the following equation:

$$p(\mathbf{x}_{0:i}, \mathbf{m} \mid \mathbf{z}_{0:i}) = \eta p(\mathbf{x}_0) \prod_k p(\mathbf{z}_k \mid \mathbf{x}_k, \mathbf{m}) \tag{1}$$

Where $\mathbf{x}_{0:N}$ is a representation of camera position and orientation in a time of capturing observations $\mathbf{z}_{0:N}$, $\mathbf{m}$ is a mathematical representation of the environment and η is normalization constant ensuring $\int p(\mathbf{x}_{0:i}, \mathbf{m} \mid \mathbf{z}_{0:i}) d\mathbf{x}_{0:i}, \mathbf{m} = 1$.

Let us notice that dimensionality of estimated variables raises linearly with observations count. On the contrary, the recurrent processing is concerned only with estimation of actual position and the environment structure, from the previous estimate and a current observation. So with the same notation can be defined as:

$$p(\mathbf{x}_i, \mathbf{m} \mid \mathbf{z}_{0:i}) = \eta p(\mathbf{z}_i \mid \mathbf{x}_i, \mathbf{m}) \int p(\mathbf{x}_{i-1}, \mathbf{m} \mid \mathbf{z}_{0:i-1}) d\mathbf{x}_{i-1} \tag{2}$$

Where the integral part of the equation is a marginalization step which removes any previous camera poses from the estimated variable set and ensure that estimation problem will have dimensionality constant in terms of observation count.

However, while definitions and results in the form of the probability distribution are generally valid they can be not only extremely time-consuming to quantify but also challenging to work within real-world situations. In following subsections, we introduce assumptions which allow us to traverse from this mathematical abstraction to practically applicable algorithms.

2.1 Mathematical camera model

The basic assumption which makes generally every estimation task solvable is that the estimator has some knowledge about the link between estimated variables and processable observations. This knowledge is in preceding definition (1) (2) represented by term $p(\mathbf{z}_i \mid \mathbf{x}_i, \mathbf{m})$ and in the scope of this paper we will assume that it is a camera model defined as:

$$\mathbf{z}_i = \mathrm{h}(\mathbf{x}_i, \mathbf{m}) + v \tag{3}$$

Where v is a stochastic variable which has multivariate normal distribution with zero mean and known covariance matrix $\mathbf{R}$ and $\mathrm{h}(\cdot)$ is function composed by sequential application of the following function on every environment point:

$$h_k(\mathbf{x}, \mathbf{m}_k) = \begin{pmatrix} f & 0 & c_x & 0 \\ 0 & f & c_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} R(\mathbf{x}) & t(\mathbf{x}) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} m_{k,x} \\ m_{k,y} \\ m_{k,z} \\ 1 \end{pmatrix} \tag{4}$$

Where f, c_x, c_y are internal camera parameters and $R(\mathbf{x})$, $t(\mathbf{x})$ is rotational matrix and translation vector dependent on camera position and orientation.

By the end of the camera model definition, we want to quickly underline that this model is highly idealized. It mainly reflects way how 3D points are projected into an image plane of the freely positioned camera but do not take into account many real image processing problems as feature detection, correspondence search, limited angular range etc. However, for purposes of our simulations, we find it sufficient.

2.2 Solvability and unambiguity of multiple views reconstruction

Reconstruction based on series of camera views is an extensive and complex topic and it is in detail described in [1]. Camera projection is not surjective function so simple back projection leads to set of possible solutions. Combining information of multiple views may lead to a unique solution, however, itself does imply that generally. The exact rules to determine a level of reachable unambiguity are out of the scope of this paper. We just want to simply state that the way we defined camera model, camera trajectory, environment structure and information which estimator gets initially leads to image sequence processable into unambiguous reconstruction. Moreover, we want to briefly summer up assumptions that lead to this statement: Model we defined is model of monocular camera so every camera position and orientation is considered to be independent of any other, camera moves between each observation to unique position, no observed point is collinear with any two consequence camera positions, internal camera parameters are known to the estimator and environment points are not all coplanar to each other. With such defined parameters, the reconstruction can be uniquely computed invariantly up to 3D similarity transformation (no link to the global reference frame and metrical units). To get rid of this remaining of ambiguity we give estimator perfect information about camera position and orientation in a time of capturing first observation $\mathbf{x}_0$ (which fix rotation and translation ambiguity) and the distance between first and second camera position $\|t(\mathbf{x}_0)-t(\mathbf{x}_1)\|$ (that makes the scale factor unambiguous).

2.3 Environment and its parametrization

The defined camera model is point oriented so the environment consists of a set of 3D points to be compatible with it. Mathematical representation is a vector which contains a collection of coordinates of all points:

$$\mathbf{m} = \left(\mathbf{m}_1^T \mid \mathbf{m}_2^T \cdots \mathbf{m}_M^T\right)^T \tag{5}$$

As mentioned in advance the aim of this paper is on effects of environment parametrization so let us consider that the 3D points are not from estimators scope all independent of each other. Let us consider that estimator has been given with knowledge of some model that these point bounds:

$$\mathbf{m} = \mathrm{g}(\mathbf{p}) \tag{6}$$

Where $\mathbf{p}$ is a vector which represents set of parameters of environment model.

We had proposed three parametrizations $\mathrm{g}_{1:3}$ to examine their effects on reconstruction process. The first is just a reference so g_1 is an identity function, the second parametrization is weak a reduce dimensionality slightly by considering that clusters of $\mathbf{m}$ lies on same in advance unknown plane (each point lies on exactly one plane).

$$\mathbf{p} = \left(\mathbf{p}_1^T \mid \mathbf{p}_2^T \cdots \mathbf{p}_K^T\right)^T \tag{7}$$

$$\mathbf{p}_j = \left(\mathbf{s}_{j,1}^T \mid \mathbf{s}_{j,2}^T \mid d_j \mid \mathbf{r}_{j,1}^T \mid \mathbf{r}_{j,2}^T \cdots \mathbf{r}_{j,L}^T\right) \tag{8}$$

Where $\mathbf{s}_1, \mathbf{s}_2$ are direction vectors orthogonal to each other with unit length and their cross product forms plane normal vector $\mathbf{n}_j = \mathbf{s}_{j,1} \times \mathbf{s}_{j,2}$ (they are basis vector which forms the plane coordinate system), d_j is offset parameter from general form plane equation and $\mathbf{r}_{j,l}$ is 2D coordinates vector in plane coordinates system.

A piece of function g_2 that present projection of single point looks like this:

$$\mathbf{m}_i = \left(\mathbf{s}_{j,1} \mid \mathbf{s}_{j,2}\right)\mathbf{r}_{j,i_j} + d\left(\mathbf{s}_{j,1} \times \mathbf{s}_{j,2}\right) \tag{9}$$

Where i_j is an index of the projected point in the plane i_j indexing system.

The last parametrization we used reduced dimensionality significantly by considering that the points are in uniform grid mapped on the surface created by two unequal planes. Parameter vector is then composed in the following manner:

$$\mathbf{p} = \left(\mathbf{s}_1^T \mid \mathbf{s}_2^T \mid \mathbf{s}_3^T \mid \mathbf{c}^T \mid \mathbf{a}^T\right) \tag{10}$$

Where $\mathbf{x}$ is unit length 2D vector representing the angular difference between planes forming surface parametrization and $\mathbf{a}$ is a vertical and horizontal grid density parameter.

A piece of function g_3 that present projection of single point looks like this:

$$\mathbf{m}_i = \left(\mathbf{s}_x \mid \mathbf{s}_3\right)\mathrm{diag}(\mathbf{a})\begin{pmatrix} i_x \\ i_y \end{pmatrix} + \mathbf{c} \tag{11}$$

Where i_x, i_y are grid indexes assigned to each point a priory and $\mathbf{s}_x$ is either $\mathbf{s}_1$ or $\mathbf{s}_2$ based on point position.

2.4 Non-recurrent estimator

In scope of this paper we deal with effectivity of recurrent estimators, however, every recurrent estimator process observation and some previous estimate and because it has to start somehow the first processed previous estimate (initial estimate) have to come up eider from a prior knowledge or some non-recurrent estimator, which processed a small set of observations. And moreover, as is shown later, the initial estimate quality can have a large impact on recurrent algorithm performance.

During our experiments, we utilize for purposes of non-recurrent estimation the Maximum likelihood method. This method is based on maximization of the so-called likelihood function $\mathcal{L}$, however, from several practical reasons is for nonlinear estimation used logarithmical likelihood function $\ell = \log(\mathcal{L})$ it does not change the position of maximum only its value. So applied on our problem can be the method defined as:

$$\begin{pmatrix} \hat{\mathbf{x}}_{ML,1:N} \\ \hat{\mathbf{p}}_{ML,i} \end{pmatrix} = \underset{\mathbf{x}_{1:N},\mathbf{p}\in\Omega}{\operatorname{argmax}}\left(\ell\left(\mathbf{x}_{1:N}, \mathbf{p} \mid \mathbf{z}_{0:i}, \mathbf{x}_0\right)\right) \tag{12}$$

Where the combined log-likelihood function is:

$$\ell\left(\mathbf{x}_{1:i}, \mathbf{p} \mid \mathbf{z}_{0:i}, \mathbf{x}_0\right) = \ell\left(\mathbf{p} \mid \mathbf{z}_0, \mathbf{x}_0\right) + \sum_{k=1}^{i} \ell\left(\mathbf{x}_k, \mathbf{p} \mid \mathbf{z}_k\right) \tag{13}$$

Where the partial log-likelihood function is because of assumption of normally distributed noise in the quadratic form:

$$\ell\left(\mathbf{x}_i, \mathbf{p} \mid \mathbf{z}_i\right) = -\frac{1}{2}\mathbf{e}_i^T \mathbf{R}^{-1} \mathbf{e}_i \tag{14}$$

Where the error:

$$\mathbf{e}_i = \left(\mathbf{z}_i - h\left(\mathbf{x}_i, \mathrm{g}_l\left(\mathbf{p}\right)\right)\right) \tag{15}$$

Because of quadratic form nature of partial log-likelihood functions, the (12) can be solved numerically using Gauss-Newton method, however, we use Levenberg–Marquardt algorithm for its superior reliability.

By the end of non-recurrent estimator definition, we want to mention computation of estimation error covariance matrix. Although directly for purposes of non-recurrent estimation it has no meaning, it plays a significant role during the transition from this non-recurrent algorithm, which served as a source of the initial estimate, to some recurrent algorithm. To make this formula we utilize linearization of the error function using first-order Taylor expansion. It has common pros and cons of this approach.

$$\mathbf{P}_{(x_{1:i}, m), i} = \left(\sum_{k=0}^{i} \mathbf{E}_k^T \mathbf{R}^{-1} \mathbf{E}_k\right)^{-1} \tag{16}$$

Where $\mathbf{E}_k$ is Jacobi matrix of the error function $\mathbf{e}_i$.

$$\mathbf{E}_k = \left.\frac{\partial \mathbf{e}_k}{\partial\left(\mathbf{x}_{1:i}^T \mid \mathbf{p}^T\right)^T}\right|_{\begin{pmatrix} \hat{\mathbf{x}}_{ML,k} \\ \hat{\mathbf{p}}_{ML,k} \end{pmatrix}} \tag{17}$$

2.5 Recurrent estimation

Let us proceed to main tools of our experiments – recurrent estimator. For our experiments, we used two algorithms: EKF and recurrently applied maximum likelihood method.

Let us firstly look at common properties of both algorithms we utilized. They both operate with an estimate in a form composed of two statistics - estimate vector and covariance matrix. Algorithms inputs are observation $\mathbf{z}_i$ and the estimate representing $p\left(\mathbf{p} \mid \mathbf{z}_{0:i-1}\right)$. The output is an estimate representing $p\left(\mathbf{x}_N, \mathbf{p} \mid \mathbf{z}_{0:i}\right)$. However, one purpose of output is to became input in time of new observation so due to this have to be on every output estimate conducted before mentioned $\mathbf{x}_{i-1}$ outmarginalization. Because estimate form is this step straightforwardly realized by selection subvector and submatrix corresponding exclusively to $\mathbf{p}$.

EKF is so popular algorithm that in its basic form probably needs no introduction, however, our application slightly differs from the standard everywhere-to-find definition, because we have no link between $\mathbf{x}_i$ and $\mathbf{x}_{i-1}$ to utilize. Nevertheless, basic assumptions are the same we are looking for a linear unbiased solution which gives minimal variance estimates. To begin we have look for a way to obtaining unbiased estimate vector form estimator in the following form:

$$\begin{pmatrix} \hat{\mathbf{x}}_{EKF,i} \\ \hat{\mathbf{p}}_{EKF,i} \end{pmatrix} = \mathbf{a} + \mathbf{K}_i \mathbf{z}_i \tag{18}$$

Taking into consideration the approximation:

$$\mathrm{h}(\mathbf{x}_i, \mathrm{g}_l(\mathbf{p})) \approx \mathrm{h}(\hat{\mathbf{x}}_{i|i-1}, \mathrm{g}_l(\hat{\mathbf{p}}_{i-1})) + \mathbf{H}_x(\mathbf{x}_i - \hat{\mathbf{x}}_{i|i-1}) + \mathbf{H}_p(\mathbf{p} - \hat{\mathbf{p}}_{i-1}) \tag{19}$$

Where: $\mathbf{H}_x = \left.\frac{\partial \mathrm{h}}{\partial \mathbf{x}}\right|_{\begin{pmatrix} \hat{\mathbf{x}}_i \\ \hat{\mathbf{p}}_i \end{pmatrix}}$, $\mathbf{H}_p = \left.\frac{\partial \mathrm{h}}{\partial \mathbf{p}}\right|_{\begin{pmatrix} \hat{\mathbf{x}}_i \\ \hat{\mathbf{p}}_i \end{pmatrix}}$ are Jacobi matrices.

The unbiased solutions are:

$$\begin{pmatrix} \hat{\mathbf{x}}_{EKF,i} \\ \hat{\mathbf{p}}_{EKF,i} \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{x}}_{i|i-1} \\ \hat{\mathbf{p}}_{i-1} \end{pmatrix} + \mathbf{K}_i\left(\mathbf{z}_i - \mathrm{h}(\hat{\mathbf{x}}_{i|i-1}, \mathrm{g}_l(\hat{\mathbf{p}}_{i-1}))\right) \tag{20}$$

Where, however, the Kalman gain matrix is not completely free. Splitting it into two submatrices $\mathbf{K}_i = (\mathbf{K}_{x,i}^T \mid \mathbf{K}_{p,i}^T)^T$ the unbiased conditions can be defined by this two matrix equations: $\mathbf{K}_{x,i}\mathbf{H}_x = \mathbf{I}$, $\mathbf{K}_{p,i}\mathbf{H}_x = \mathbf{0}$.

Using this conditions the covariance can be expressed in following way

$$\mathbf{P}_{(x,p),i} = \begin{pmatrix} -\mathbf{K}_{x,i}\mathbf{H}_p \\ \mathbf{I} - \mathbf{K}_{p,i}\mathbf{H}_p \end{pmatrix} \mathbf{P}_{p,i-1} \begin{pmatrix} -\mathbf{K}_{x,i}\mathbf{H}_m \\ \mathbf{I} - \mathbf{K}_{p,i}\mathbf{H}_m \end{pmatrix}^T + \mathbf{K}_i \mathbf{R} \mathbf{K}_i^T \tag{21}$$

And because unbiased conditions do not fully define the Kalman gain the remaining variability is used to minimize estimate variance e.g. the trace of the covariance matrix $\mathrm{tr}(\mathbf{P}_{(x,m),i})$. This leads to Kalman gain:

$$\mathbf{K}_i = \begin{pmatrix} \mathbf{H}_x^+ - \mathbf{H}_x^+(\mathbf{R} + \mathbf{H}_p \mathbf{P}_{p,i-1} \mathbf{H}_p^T)\mathbf{S}_{inv} \\ \mathbf{P}_{p,i-1}\mathbf{H}_p^T \mathbf{S}_{inv} \end{pmatrix} \tag{22}$$

Where $\mathbf{H}_x^+$ is any solution of $\mathbf{K}_x\mathbf{H}_x = \mathbf{I}$ obtained for example with Moore–Penrose pseudoinverse and

$$\mathbf{S}_{inv} = \mathbf{N}_x{}^T\left(\mathbf{N}_x(\mathbf{R} + \mathbf{H}_m \mathbf{P}_{N|N-1} \mathbf{H}_m{}^T)\mathbf{N}_x{}^T\right)^{-1}\mathbf{N}_x \tag{23}$$

Where $\mathbf{N}_x$ is left nullspace of $\mathbf{H}_x$.

Final thing left to define is how to get position and orientation linearization point $\hat{\mathbf{x}}_{i|i-1}$. Standard EKF would utilize motion model $\hat{\mathbf{x}}_{i|i-1} = \mathrm{f}(\hat{\mathbf{x}}_{i-1})$, however, we previously specified that camera positions are independent of each other and we have no motion model to exploit. We instead use ML estimation $\hat{\mathbf{x}}_{i|i-1} = \underset{\mathbf{x}_i \in \Omega}{\mathrm{argmax}}\left(\ell(\mathbf{x}_i \mid \mathbf{z}_i, \hat{\mathbf{m}}_{i-1})\right)$ which is from a formal point of view consistent with our assumptions.

Recurrently applied maximum likelihood method is conceptually similar to the non-recurrent application. It is also based on maximization of the likelihood function

$$\begin{pmatrix} \hat{\mathbf{x}}_{ML,i} \\ \hat{\mathbf{p}}_{ML,i} \end{pmatrix} = \underset{\mathbf{x}_i, \mathbf{p} \in \Omega}{\mathrm{argmax}}\left(\ell(\mathbf{x}_i, \mathbf{p} \mid \mathbf{z}_i, \hat{\mathbf{p}}_{i-1})\right) \tag{24}$$

Where the likelihood function is a combination of observation likelihood and recurrent likelihood:

$$\ell(\mathbf{x}_i, \mathbf{p} \mid \mathbf{z}_i, \hat{\mathbf{p}}_{i-1}) = \ell(\mathbf{p} \mid \hat{\mathbf{p}}_{i-1}) + \ell(\mathbf{x}_i, \mathbf{p} \mid \mathbf{z}_i) \tag{25}$$

Where the recurrent term should be the best possible approximation of theoretical collected likelihood function based on all processed observations

$$\ell(\mathbf{p} \mid \hat{\mathbf{p}}_{i-1}) \approx \ell(\mathbf{p} \mid \mathbf{z}_{0:i-1}, \mathbf{x}_0) \tag{26}$$

We carry out this approximation in the quadratic form:

$$\ell(\mathbf{p} \mid \hat{\mathbf{p}}_{i-1}) = (\mathbf{p} - \hat{\mathbf{p}}_{i-1})^T \mathbf{P}_{p,i-1}^{-1}(\mathbf{p} - \hat{\mathbf{p}}_{i-1}) \tag{27}$$

Where the quadratic term is obtained via linearization of the observation function as:

$$\mathbf{P}_{(x_i,p),i}^{-1} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_{p,i}^{-1} \end{pmatrix} + \mathbf{H}_{(x_i,p),i}^{T} \mathbf{R}^{-1} \mathbf{H}_{(x_i,p),i} \tag{28}$$

3 Simulation

For evaluation of effects of different parametrizations on recurrent estimator quality, we decided to realize a simulation. We created a virtual environment composed of 32 3D points. These points are uniformly spaced and lie on two planes (as demanded by g_3). Then we define parameters for camera model $f=1$, $c_x=0$, $c_y=0$ and its trajectory. The trajectory is a spiral around environment points and the camera is always oriented in a way that its optical axis perpendicularly intersects the horizontal axis of environment frame.

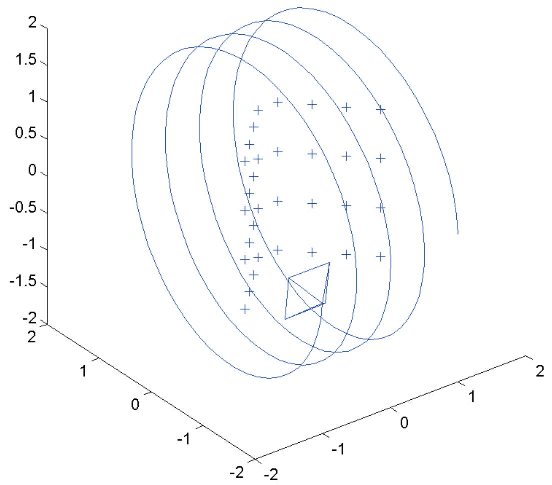

Figure 1: Environment with camera trajectory

On this trajectory are then conducted 200 observations which are then degraded by normally distributed noise with zero mean and standard deviation $\sigma = 0.1$. This bundle of 200 observations is subjected to processing by a predefined set of recurrent estimators using a predefined set of parametrizations. Namely, we use three estimators: EKF and recurrently applied maximum likelihood method. And we also look into dependency on quality of initial estimate by having initial estimate calculated by the non-recurrent processing of 2, 4, 8, 16, 32 and 64 observations.

During the implementation of estimators, we utilized concepts mentioned in [3] which deal with the non-singular and minimal representation of parameter vector. So every derivatives and increments are calculated in minimal form and then are added to the non-singular form using nonlinear operator. For example parametrization of g_2 contains two base 3D vectors although during optimization we compute only two angular increments and apply them via rotational transform. Base vectors are highly constrained because they have to have unit norm and be orthogonal to each other, however, they are much convenient to work with then the angular representation which would be much harder to be conveniently referenced to environment frame and properly limited. In the scope of this concept, we want to clearly state dimensionality of compared environment representations in following Table 1.

Table 1: Dimensionality of different parametrizations

Parametrization	Dimensionality of the non-singular representation	Dimensionality of the minimal representation
g_1 - reference	96	96
g_2 - weak	78	70
g_3 - strong	14	8

To evaluate statistical properties of different combinations against each other we conducted each estimation 100 times with different observation noise realizations. and evaluate two statistics: Estimation error using Euclidian norm to evaluate the precision of the reconstruction based on estimator parameters

$$Err_{Euclid} = \sum_{k=1}^{32} \left\| \mathbf{m}_k - g_l\left(\hat{\mathbf{p}}_{200}\right)_k \right\| \tag{29}$$

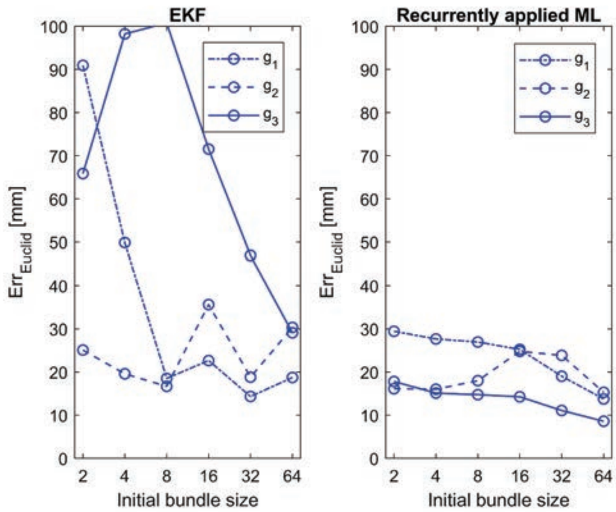

Figure 2: Median of euclidian estimation error

And secondly, we error the error using Mahalanobis distance to evaluate how is the estimate consistent with reality.

$$Err_{Mahanobis} = \left(\mathbf{p} - \hat{\mathbf{p}}_{200}\right)^T \mathbf{P}_{m,200}^{-1} \left(\mathbf{p} - \hat{\mathbf{p}}_{200}\right) / \dim(\Delta \mathbf{p}) \tag{30}$$

It is actually normalized squared Mahalanobis distance because the square of Mahalanobis distance computed on normally distributed data should be chi-squared distributed and this distribution has a mean value equal to degrees of freedom so to make this statistic visually comparable between estimates with different dimensionalities we normalize it to theoretically has unit mean value.

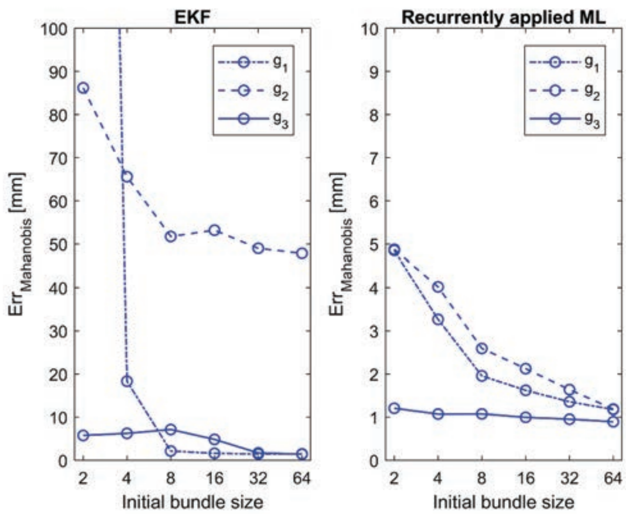

Figure 3: Median of mahanobis estimation error

Please note that for clarity of graphs we have to use the different scale on y-axis – EKF graph has 10 times larger step then recurrently applied ML.

4 Discussion

Let us discuss results we received: We used relatively simple simulation to get them instead of processing real camera images. Due to this, we avoided several serious problems which would have to be dealt, for example, features detection, their association and environment model classification. However, this gives us large freedom in choice of environment model and also in error evaluation. We tried to partially compensate the skipped problems by adding noise with unrealistically high variance proportionally to signal scale.

Looking on graphs in Figure 2 and 3 it is clear that both errors go lower with increasing size of the initial bundle. This is a phenomenon which we expected because processing larger set of observations will benefit more from central limit theorem and thus leads to estimate which better approximates the reality.

Specifically, on Euclidian error graphs, we can observe not only that recurrently applied ML performs almost always superiorly but also differences in the structure of performance based on parametrizations – EKF performs badly on strongest parametrization but the same parametrization used by recurrently applied ML leads to best results. And while observing the course of Mahalanobis error graphs we can see either that EKF is much more prone to lead to inconsistent estimate and also that with some exceptions both estimators produce most consistent estimates using strongest parametrization and least consistent using weak parametrization.

We used a statically set number of observations to form the initial bundle, however, it is clear that no generally optimal size of observation bundle can be found. Because it highly depends on facts whether chosen estimates distribution parametrization is able to appropriately approximate non-recurrent likelihood. Generalization of this problem actually leads to a testing statistical hypothesis about two multivariate distributions equality which considering hundreds of dimensions is a problematic task even without any time constraints.

5 Conclusion

To conclude our findings, our interpretation of the results of the experiment is that exploiting some a priory given information about a structure of reconstructed environment is not universally positive think. Because in this process occurs two contradictory factors: on the one hand it leads to reduction of estimation problem dimensionality which is the positive factor, however, on the other combined nonlinearities of camera model and environment model increasing demands on the estimator algorithm and makes the linearization using Taylor expansion less appropriate way for random variable propagation.

In near future, we want to in the same context explore effectivity of unscented transformation for propagation statistical moments via a nonlinear function either in form of Unscented Kalman filter as a tool for obtaining statistical moments from the course of likelihood function around ML estimate. And in long term, we aim at developing a quick and efficient way for processing camera observations which would appropriately switch between non-recurrent and recurrent independently on chosen environment model.

Acknowledgment: The completion of this paper was made possible by the grant No. FEKT-S-17-4234 - „Industry 4.0 in automation and cybernetics" financially supported by the Internal science fund of Brno University of Technology.

References

[1] Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge University Press, Cambridge (2003).

[2] Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle Adjustment — A Modern Synthesis. In: Vision Algorithms: Theory and Practice. pp. 298-372. Springer Berlin Heidelberg, Berlin, Heidelberg (2000). DOI: 10.1007/3-540-44480-7_21.

[3] Kummerle, R., Grisetti, G., Strasdat, H., Konolige, K., Burgard, W.: G2o: A general framework for graph optimization. In: IEEE International Conference on Robotics and Automation (ICRA). pp. 3607-3613. IEEE Publishing (2011). DOI: 10.1109/ICRA.2011.5979949.

[4] Horak, K., Richter, M., Kalova, I.: Human eyes localization for driver inattention monitoring system. In: R. Matousek (ed.), In MENDEL, (MENDEL 2009), vol. 15, No. 1, pp. 283-288, Brno University of Technology, Brno (June 2009), ISSN 1803-3814

[5] Horak, K., Richter, M., Kalova, I.: Automated Flaws Detection on Bottles in Food Industry. Annals of DAAAM & Proceedings. 20, 133-133 (2009).

[6] Jelinek, A., Zalud, L., Jilek, T.: Fast total least squares vectorization. Journal of Real-Time Image Processing. 1 - 17 (2016). DOI: 10.1007/s11554-016-0562-6.

[7] Prochazkova, J., Kratochvil, J.: Direct point cloud visualization using T-spline with edge detection. In: Advances in Intelligent Systems and Computing. 240-251. Springer Verlag (2017). DOI: 10.1007/978-3-319-58088-3_23

[8] Engel, J., Schöps, T., Cremers, D.: LSD-SLAM: Large-Scale Direct Monocular SLAM. Computer Vision – ECCV 2014. 834 (2014). DOI: 10.1007/978-3-319-10605-2_54.

[9] Xiao, J., Zhang, J., Adler, B., Zhang, H., Zhang, J.: Three-dimensional point cloud plane segmentation in both structured and unstructured environments. Robotics and Autonomous Systems. **61**(12), 1641-1652 (2013) DOI: 10.1016/j.robot.2013.07.001.

[10] Durrant-White, H., Bailey T.: Simultaneous localization and mapping (SLAM): Part I. IEEE Robotics and Automation Magazine **13**(2), 99-108. DOI: 10.1109/MRA.2006.1638022.

[11] Bailey, T., Durrant-White, H.: Simultaneous localization and mapping (SLAM): Part II. IEEE Robotics and Automation Magazine **13**(3), 108-117. DOI: 10.1109/MRA.2006.1678144.