

Sequential Covariance-weighted Quasiconvex solution to Mapping in Visual SLAM

A.H. Abdul Hafez *

* Dept. of Computer Engineering, Faculty of Engineering,
Hasan Kalyoncu University, Sahinbey, 27410 - Gaziantep, Turkey
(e-mail: abdul.hafez@hku.edu.tr).

Abstract: This paper presents a new sequential real time algorithm that solves the mapping problem in Visual SLAM. The considered problem is a particular example from the triangulation problem, that has direct applications to robotic vision domain. In other words, the problem is handled as 3D estimate problem. The estimation process is formulated as a minimization problem of quasiconvex objective function. The minimization process is realized using the well-known bisection algorithm. The bisection algorithm runs sequentially solving one convex feasibility problem in each iteration, trying to reduce the bound on the 3D estimate. New image measurements arrive after every new iteration, new convex visibility problem is solved, and the bounds on the 3D estimates are updated. These steps are repeated till convergence. We conducted a set of experiments to show the applicability to the general reconstruction (triangulation) problem as well as the application to mapping Visual SLAM.

Keywords: Convex Optimization; Visual SLAM; Mapping; Robotic vision.

1. INTRODUCTION

Convex optimization is now widely recognized as a powerful and efficient tool for providing solutions to many engineering and technical problems Quoc et al. (2011); Tran et al. (2013); Abdul Hafez et al. (2007); Hartley and Kahl (2007). It has been extensively explored while searching for solutions to a family of multiple view geometric problems in computer vision Cremers and Kolev (2011); Klodt et al. (2013). A wide range of geometric computer vision problems are reformulated as convex (quasi-convex) optimization problems with a few algebraic manipulations Wu et al. (2012); Chesi (2009); Salzmann et al. (2007); Hartley and Kahl (2007); Kahl (2005); Ke and Kanade (2005, 2007).

One of the important advantages of Convex optimization is that it does not have the local minima concerns, which assures that the global solution can be reached. The optimization is done here by replacing the L_2 norm objective function, *i.e.* the sum of squared reprojection error, by the norm L_∞ of the error, Wu et al. (2012); Kahl (2005), *i.e.* by definition the maximum distance of the image point-wise re-projection error, given a set of image points. This was done because it is now apparent to the vision community that minimizing the L_2 norm is a non-trivial problem. The L_2 norm does not have convex form due to the perspective projection effects, that appear when we use a pin-hole camera. In addition, The L_2 norm suffers multiple local minima. In other words, using the L_∞ norm as an objective function with a quasi-convex optimization problem results in a global and deterministic solution to many multiple view vision problems. Of course, this global solution does not add any limitations regarding to camera model or the number of images Ke and Kanade (2007).

It was claimed in Hartley and Schaffalitzky (2004) that most multiple views vision problems can reach a global solution by minimizing the L_∞ norm of the reprojection error. Later on,

Kahl in Kahl (2005), and Ke *et al.* in Ke and Kanade (2005) proved that this L_∞ error is a quasi-convex function. Consequently, this error can be efficiently minimized as a sequence of visibility problems, *i.e.* second order cone programs (SOCP), using the famous bisection algorithm. Since the L_∞ is sensitive to noise and outliers, Ke and Kanade Ke and Kanade (2005, 2007) suggested using the m^{th} smallest error. *i.e.* the L_m norm, instead of L_∞ . This is justified since the later one is robust to outliers and noise. However, the solution in Ke and Kanade (2005, 2007) becomes a local solution by considering the L_m norm error as an objective function. Later, another robustness technique is proposed to minimize the L_∞ norm of error while continuing to provide a global solution Sim and Hartley (2006); Abdul Hafez et al. (2007). Ke and Kanade have shown in Ke and Kanade (2007) that minimizing the L_∞ norm is meaningful only when the image measurement noise is considered.

Considering the uncertainty in the image measurements helps determine the effect of its noise properties on the minimization process. As is shown in Ke and Kanade (2007), minimizing the L_∞ requires that the image noise is isotropic (it is uniformly distributed in all direction), and *i.d.d.* (it is distributed independently and identically). Unfortunately, it is not the case for most 2D features Shi and Tomasi (1994). It is shown in Shi and Tomasi (1994) that a good feature to track is the one with non-directional or directional but uncorrelated noise. This is true since the image noise depends on the dissimilarity in the intensity of the image window around the feature point. However, most image features have a large amount of directionality which need to be considered when minimizing L_∞ norm.

Several multiple view geometric problems in computer vision are efficiently solved by minimizing the L_∞ norm using convex or quasi-convex optimization framework. These problems may include planar homography estimation, triangulation / construction, camera re-sectioning, and estimation of camera motion with a given rotation Ke and Kanade (2007). Triangulation

problem is the problem of estimating the 3D points M given their images m_i and the camera pose matrices \mathcal{P}_i . As another example, plan to plan homography estimation appears when the 3D points belong to the same plane. A homography can be estimated between 3D points and its images $m = HM$, or between two images of the 3D points $m_1 = H_{12}m_2$. It is commonly known now Ke and Kanade (2007); Kahl (2005); Ke and Kanade (2005); Hartley and Schaffalitzky (2004) that all of these problems can be solved by formulating a quasiconvex reprojection error. We particularly consider the reconstruction / triangulation problem. The current framework, *i.e.* geometric reconstruction, is suitable for problems like building a 3D model using a sequence of images of the environment. For example, given a video sequence (set of images) with a calibrated camera, the 3D/CAD model of the environment can be efficiently reconstructed. This is, in fact, known in the computer vision community as the batch processing techniques. However, several computer vision geometric problems need to be handled in a sequential manner to satisfy the real time requirements of the system.

As an application to real time systems, Hafez et al. proposed in Abdul Hafez et al. (2008) a sequential convex optimization algorithm that efficiently solves the mapping problem in VS-LAM. The most important step was the initialization of 3D features, and the insertion of them into the map. Since the camera pose is estimated using other previously available features, the pose is used to formulate the feature initialization as sequential triangulation problem. Another application of sequential convex optimization algorithm is presented in Abdul Hafez and Jawahar (2008) to estimate the depth of the visual features, assuming that the camera pose is estimated using a particle filter. The work presented in this paper is the continuation of these two works. Here, we adopt the minimization of the uncertainty-weighted L_∞ norm instead of the classical one.

The contribution of this paper is twofold. On one hand, it presents the adaptation of the geometric reconstruction problem to the sequential situations, where the real time constraints are satisfied. On the other hand, it considers the uncertainty in the image data. To adapt to the real time constraints, the solution is formulated as a solution of quasi-convex problem. The bisection algorithm is used as an on-line recursive method to solve a set of visibility convex optimization problems. The uncertainty is considered by transforming the classical L_∞ norm into a covariance weighted space where is the noise in 2D features becomes isotropic and independently identically distributed *i.i.d.*. The real time solution has applications in many domains like visual servoing, visual SLAM, virtual reality, structure from motion, etc. However, the proposed real time solution is suitable to a wide variety of geometric problems solved by minimizing the L_∞ norm using bisection algorithm via quasi-convex optimization framework. The only sequential solution, presented in the literature, to the minimization of L_∞ error was proposed in Seo and Hartley (2007). Similar algorithms that estimate 3D map in real time, by drawing a set of particles but not using convex optimization, is the one presented in Strasdat et al. (2012); Davison et al. (2007) and reviewed in Subsection 5. Its performance is compared to ours in Subsection 6.2.

The remaining parts of this paper are organized as follows: The next section presents a background about imaging process and multi-view geometry problems, about convex optimization with solutions to L_∞ norm problems, and finally more focus is given to the triangulation problem. Section 2 presents our

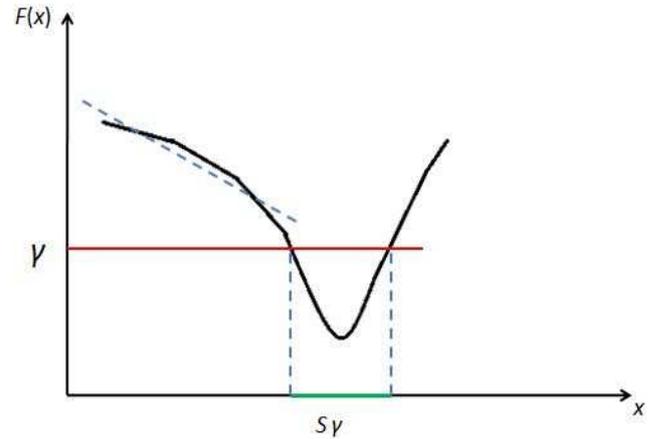


Fig. 1. The function $f(x)$ is example of quasiconvex function defined on \mathbb{R} .

sequential real time solution to the L_∞ norm problems. Section 4 presents our proposed algorithm, while Section 5 presents the application of the proposed algorithms to mapping Visual SLAM. Discussion about experiments that are carried out using the proposed algorithm is presented in Section 6.

2. BACKGROUND

We summarize the background material related to L_∞ norm and convex optimization, but more rigorous details can be seen in Boyd and Vandenberghe (2004).

2.1 Convex and Quasi-convex Optimization

By definition, we have convex optimization problem if the objective function $f_0(x)$ is convex, and it is minimized under a set of constraints. These constraints are convex functions $f_i(x)$; where $i = 1, \dots, N$. In more formal words, the convex optimization problem is the one of the form

$$\min_x f_0(x) \quad (1)$$

$$\text{s.t. ,} \\ f_i(x) \leq b_i, \quad i = 0, \dots, N.$$

However, the problem is not convex problem any more if the objective function $f_0(x)$ is quasi-convex function. It is indeed quasi-convex problem, see Figure 1 for definition of quasiconvex function. All γ -sublevel sets S_γ of the function $f(x)$ are convex range. In general, quasi-convex functions are not necessarily convex. In this example function, one may note that a segment of the dashed line, between the intersection points, lies below the portion of the function between the two intersection points. Usually the line segment is above the convex functions.

Let us have the parameter $\gamma \in \mathbb{R}$ and optimal solution $f_0^*(x) \leq \gamma$ to problem (1). To find such unknown solution, the following feasibility problem is feasible:

$$\text{find } x \quad (2)$$

$$\text{s.t. ,} \\ f_0(x) \leq \gamma \\ f_i(x) \leq b_i, \quad i = 0, \dots, N.$$

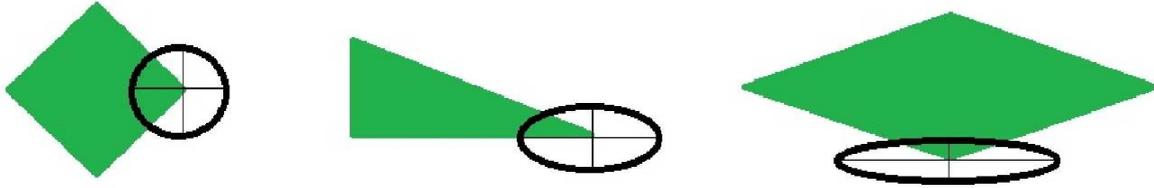


Fig. 2. The directionality types in image measurement uncertainty. Left: the uncertainty is isotropic and uncorrelated. Middle: the uncertainty is not isotropic, directional but uncorrelated. Right: the uncertainty is highly directional and uncorrelated.

If this problem is not feasible, then the solution f_0^* of problem (1) satisfies that $f_0^*(x) > \gamma$.

The bisection algorithm repeats the same steps above to find the optimal solution to the quasi-convex problem. As it is shown in Algorithm 1, the algorithm starts from initial given lower bound and upper bound $[\gamma^l, \gamma^h]$ of the optimal value f_0^* of the objective function. It initially solves the feasibility problem given in (2) for the lower half of the range $[\gamma^l, \gamma^h]$, that is after updating the higher bound feasibility range as $\gamma^h = (\gamma^l + \gamma^h)/2$. If the mentioned problem is not feasible, this means that the optimal value of the objective function holds as $f_0^*(x) > \gamma$. Indeed, the lower bound of the range is updated as $\gamma^l = (\gamma^l + \gamma^h)/2$. After a few number of iterations in which a set of feasibility problems are solved indeed the range is partitioned, the range of the feasibility becomes $\gamma^h - \gamma^l \leq \epsilon$, hence the produced solution is optimal.

2.2 Uncertainty in 2D/3D features

Since, non-directional features are rarely available, it was proposed in Ke and Kanade (2007) to minimize a weighted form of the L_∞ norm. The norm is weighted based on the directionality of the uncertainty in the image measurement. In other words, the Mahalanobis distance between the image measurement and its corresponding reprojection is considered instead of the Euclidean distance. For example, the Euclidean distance was used in Kahl (2005); Ke and Kanade (2005); Hartley and Schaffalitzky (2004). Since the directionality is well-represented using the covariance matrix of the 2D feature measurement, they proposed to use covariance-weighted L_∞ norm. It was shown that it is a quasi-convex function indeed a global minimum can be obtained by minimizing it. Using such error function, the contribution of each feature to the minimization process is inversely proportional to the amount of directionality in the image uncertainty. In other words, less directionality imposes stronger constraints on the unknowns under optimization.

Following, we demonstrate the uncertainty in image measurement and how it can be projected into the 3D space to represent the uncertainty of the estimated 3D point. Consider the projection $m = (x, y)$ of the 3D point $M = [X, Y, Z]^T$ to the image space of a given camera. Additive noise and different kinds of errors can affect the measurement \hat{m} of this image point. The zero mean Gaussian probability function is a suitable model for considering noise and errors. Consequently, the image measurement is represented as a random variable that has the distribution (Flandin and Chaumette (2001); Abdul Hafez and Jawahar (2006)) given by

$$p(m | M) = \frac{1}{(2\pi|\Sigma_m|^{1/2})} \exp\left[-\frac{1}{2}(m - KM)^T \Sigma_m^{-1} (m - KM)\right]. \quad (3)$$

The matrix Σ_m is the covariance matrix. It depends on the image measurements.

The inverse of the covariance matrix is computed using the following formula:

$$\Sigma_m^{-1} = \sum_{(x,y) \in w} \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix} \quad (4)$$

Here, w is a small patch considered around the feature position $m = (x, y)$ in image I of the corresponding camera. The vectors I_x and I_y are image gradient vectors along x and y directions respectively.

Based on the type of the uncertainty, shown in Figure 2, the matrix Σ_m shows three mathematical characteristics:

- (1) The matrix has two identical variance values in both x and y directions, *i.e.* $\Sigma_m = \text{diag}(\lambda, \lambda)$. This expresses scalar, isotropic, and uncorrelated uncertainty in both directions. This case is demonstrated in Figure 2(left).
- (2) The matrix has two different variance values in both x and y directions, *i.e.* $\Sigma_m = \text{diag}(\lambda_1, \lambda_2)$. This expresses directional but uncorrelated uncertainty in both directions. This case is demonstrated in Figure 2(middle).
- (3) The matrix Σ_m is full 2×2 matrix. This expresses correlated uncertainty with high directionality in both directions x and y . This case is demonstrated in Figure 2(right).

Let us have a depth map as the distribution $p(Z) = \mathcal{N}(Z; \bar{Z}, \sigma_Z)$. The mean of this distribution is \bar{Z} and the variance is σ_Z . The uncertainty in the image measurements can be reprojected to the Cartesian space using the function F^{-1} , the inverse of the reprojection function of the camera. A 3D distribution of 3D point M that corresponds to the image point measurement m is obtained. This distribution is $p(M | m) = \mathcal{N}(M; \bar{M}, \Sigma_M)$ and the parameters \bar{M} and Σ_M are computed as follows Flandin and Chaumette (2001)

$$\bar{M} = [\bar{Z}\bar{x}, \bar{Z}\bar{y}, \bar{Z}]^T, \quad \Sigma_M^{-1} = J_F^T \begin{pmatrix} \Sigma_m^{-1} & 0 \\ 0 & \sigma_Z^{-1} \end{pmatrix} J_F. \quad (5)$$

Here, the matrix J_F is the Jacobian of the inverse of the reprojection function Flandin and Chaumette (2001) and defined as

$$J_F = \frac{\partial F^{-1}}{\partial M} \Big|_{\bar{M}} = \begin{pmatrix} 1/\bar{Z} & 0 & -\bar{x}/\bar{Z} \\ 0 & 1/\bar{Z} & -\bar{y}/\bar{Z} \\ 0 & 0 & 1 \end{pmatrix}. \quad (6)$$

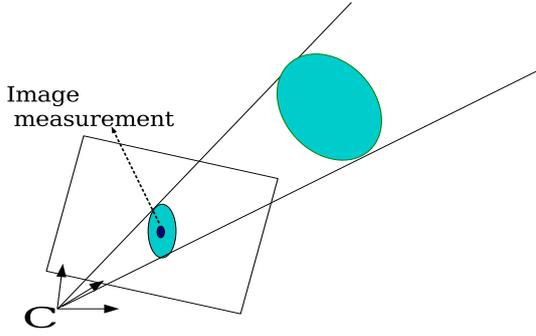


Fig. 3. Uncertainty in image measurement and its reprojection to the Cartesian space.

Figure 3 depicts the resulting 3D distribution of the estimated 3D point coordinates. This distribution is used to perform the visual servo control task and the localization process.

3. TRIANGULATION USING QUASICONVEX OPTIMIZATION WITH UNCERTAINTY

This section discusses the triangulation problem using quasi-convex optimization framework, and its extension to the real time situation. However, the presented method is general and can be applied to other problems like planar homography estimation and camera resectioning.

3.1 The image Re-projection error and its norms

It is commonly known that minimizing the image re-projection error is geometrically more meaningful than other algebraic errors Ke and Kanade (2007). Usually, the image re-projection error is defined as the difference between a given measurement \hat{m} of the image point and the projection of its 3D estimate M .

Assume that this projection $m = \mathcal{P}M$ is the image of the 3D estimate M using the camera matrix \mathcal{P} . Given an image point \hat{m} , we can write the n -norm L_n of the re-projection error as

$$L_n = \|\hat{m} - m\|_n = \|\hat{m} - \mathcal{P}M\|_n. \quad (7)$$

When $n = 2$, it means that the Euclidean distance is used. It is denoted as L_2 norm and given as $L_2 = \left[\frac{1}{N} \sum_{i=1}^N d_i^2\right]^{1/2}$ where N is the number of images (or cameras), and the distance d is given for a certain camera as

$$d^2 = \left(\hat{x} - \frac{P_1^T M}{P_3^T M}\right)^2 + \left(\hat{y} - \frac{P_2^T M}{P_3^T M}\right)^2. \quad (8)$$

In other words, when we write using the L_2 norm notation $\|\cdot\|$ we get

$$d = \left\| \frac{1}{P_3^T M} \left[\begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} P_3^T - \begin{pmatrix} P_1^T \\ P_2^T \end{pmatrix} \right] M \right\| \quad (9)$$

Here, P_j^T is the j^{th} row of the camera projection matrix \mathcal{P} . The error L_∞ norm can be similarly written as

$$L_\infty = \max_i \{d_i\}_{i=1}^N \quad (10)$$

where d is the distance function defined in (8).

The L_2 norm of the error is difficult to minimize due to multiple potential local minima problems Hartley and Schaffalitzky

(2004). In contrast, L_∞ norm has only one minimum. Hence, minimizing this L_∞ norm, obviously results in a global solution. Unfortunately, the L_∞ norm can be easily affected by noise and outliers, indeed more care needs to be taken regarding robustness issues. Additional robust processing stage is needed.

3.2 The L_∞ norm weighted using uncertainty

In this section we are going to see how the directionality of the image measurement uncertainty is considered while minimizing the L_∞ reprojection error. Assume that $M = (X, Y, Z, 1)$ are the 3D homogenous coordinates and $\hat{m} = (x, y)$ are the 2D homogenous coordinates of the image measurement. The uncertainty in the location of the image features is characterized using the inverse covariance matrix Σ_m^{-1} .

The method which was proposed in Ke and Kanade (2007) to handle the directional uncertainty is adopted here. The essence of the method is to project the reprojection error function onto a new data space. In this new space the uncertainty becomes isotropic and uncorrelated. This is done by weighting the error function using the covariance matrix of the uncertainty. This can be thanks to the symmetric positive semi-definite form of the matrix Σ_m^{-1} . This form allows it to be decomposed into V and Λ matrices like

$$\Sigma_m = V \Lambda V^T \quad (11)$$

where the matrix $\Lambda = \text{diag}(\lambda_1, \lambda_2)$, and V is a 2×2 orthogonal matrix. Similarly to Eq.(11), the inverse covariance matrix can be written as

$$\Sigma_m^{-1} = V \Lambda^{-1} V^T \quad (12)$$

One can note that the transformation

$$T_w = \Lambda^{-1/2} V^T \quad (13)$$

is affine transformation that can transform image data, hence the reprojection error, into uncertainty-weighted space. In this space the noise is isotropic and uncorrelated, that is independently and identically distributed.

Using the transformation T_w given in Eq.(13) we can write the image measurement \hat{m} as \hat{m}' and the projection m as m'

$$\hat{m}' = T_w \hat{m} = T_w (x, y)^T, \quad (14)$$

$$m' = T_w m = T_w \mathcal{P}_i M. \quad (15)$$

To write the error d in the new space, we substitute in Eq.(9) to get the weighted reprojection error

$$d^w = \left\| \frac{T_w}{P_3^T M} \left[\begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} P_3^T - \begin{pmatrix} P_1^T \\ P_2^T \end{pmatrix} \right] M \right\|, \quad (16)$$

which can be simplified as

$$d^w = \left\| \frac{\Phi M}{P_3^T M} \right\| \quad (17)$$

by denoting $\Phi = T_w \left[\begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} P_3^T - \begin{pmatrix} P_1^T \\ P_2^T \end{pmatrix} \right]$. Our final Uncertainty-weighted L_∞ norm is defined as

$$L_\infty^w = \max_i \{d_i^w\}_{i=1}^N. \quad (18)$$

3.3 Triangulation using quasiconvex optimization given N images

The triangulation problem can be formulated as quasi-convex optimization problem with respect to the 3D coordinates of the features (X, Y, Z) . The model considers that N images of

a 3D point M collected with \mathcal{P}_i cameras are available. Here $i = 1, \dots, N$ and \mathcal{P}_i is the i^{th} camera matrix. Figure 4 depicts the 3D configuration of the triangulation problem.

Let us have a set of camera matrices \mathcal{P}_i , which are estimated and independently available. Our problem is to find an estimate M of the world points using the N camera matrices and the corresponding N images. This problem can be formulated as quasi-convex optimization problem Kahl (2005).

The triangulation problem can be reformulated to consider the directionality in the image uncertainty. The 3D points M_i can be estimated by minimizing the weighted reprojection error L_∞^w shown in Eq. (18) as follows

$$\min_M \max_i \{d_i^w\}_{i=1}^N \quad (19)$$

subject to $P_{3i}^T M > 0$.

Since the objective function is $d^w = \|\frac{\Phi_i M}{P_{3i}^T M}\|$, the constraints in the second line from Eq.(19) are justified, i.e. the uncertainty-weighted distance function. The term $P_{3i}^T M$ represents the depth of the 3D point coordinate frame of camera \mathcal{P}_i . Bisection algorithm can efficiently solve this problem. Every iteration of the bisection algorithm solves a feasibility problems of the form

$$\begin{aligned} & \text{find } M \\ & \text{subject to} \quad \|\Phi_i M\| \leq \gamma P_{3i}^T M \quad (20) \\ & \quad P_{3i}^T M > 0 \quad \text{for } i = 1, \dots, N. \end{aligned}$$

It is clear that this feasibility problem is convex. It tries to find out whether the optimal solution L_∞^w is less or more than a given value γ . Thus, the quasi-convex function given in Eq. (19) can be solved through a sequence of feasibility problems like the one in Eq.(20) within the bisection algorithm whose steps are stated in Algorithm 1.

This method is used to efficiently solve reconstruction problems Kahl (2005); Ke and Kanade (2007) where a set of N images of the 3D point M is available. The bisection algorithm runs k times over the measurement values till reaching a value of the objective function that is $L_\infty^w \leq \epsilon$. In fact, it needs 5 to 10 iterations. This usually does not satisfy the real time requirements, i.e. working at the video rate. In the next section, We present an online bisection algorithm that is able to satisfy the real-time requirements of the considered problem.

Algorithm 1 Solution to the triangulation problem as quasiconvex optimization problem via off-line bisection algorithm.

- 1: **Input:** Given N images, the upper and lower bound $[\gamma^l, \gamma^h]$ of the optimal value $L_\infty^w(M)$ range, and tolerance $\epsilon > 0$.
- 2: **Repeat**
 - (a) $\gamma = (\gamma^l + \gamma^h)/2$.
 - (b) Solve problem (20), i.e. convex feasibility problem.
 - (c) **If** feasible, $\gamma^h = \gamma$;
else, $\gamma^l = \gamma$.
- 3: **Until** $\gamma^h - \gamma^l \leq \epsilon$.

4. SEQUENTIAL SOLUTION TO THE L_∞ PROBLEMS

In this section, we present our sequential bisection algorithm. The proposed algorithm is able to produce real time solution. Let us start with the batch triangulation problem that produces an optimal estimate of the unknowns. After that, we discuss

the situation when new image is available. The measurements from the new image are used to update the optimal estimate. Finally, we will present our algorithm that works sequentially in real time without any initial guess or priori available about the optimum solution.

The original *batch* 3D feature estimate problem is defined as follows. Given a set of N image correspondences \hat{m}_i , estimate the 3D point M , given the camera(s) information, with minimal re-projection error over these image point correspondences. This problem is solved by Kahl in Kahl (2005) as batch convex optimization problem. If some initial optimal solution is already available from already observed N image correspondences and their concern camera matrices, we may have the following problem. Given an optimal estimate M_N^* of the corresponding 3D point, estimate the new optimal 3D point as M_{N+1}^* when a new image measurement \hat{m}_{N+1} is acquired. This solution is sequentially updated to be still an optimal when new observations are available. This problem is discussed by Seo *et al.* in Seo and Hartley (2007). The final case which is discussed in this paper is there is no initial optimal solution available.

We assume that we do not have an initial optimal solution. In addition, the image correspondences \hat{m}_i are presented sequentially starting from $i = 1$ initially, till $i = N$ at the current moment in time when the \hat{m}_N measurement is presented. The problem here is to sequentially estimate the optimal solution M_N^* as soon as two image measurements are available. Our task is to find a 3D estimate M_i , starting by $i = 2$ that converges to the optimal one M_i^* soon after a few image measurements, say k images, are available.

Let us assume that we have a moving camera attached to frame F^t and has the camera matrix \mathcal{P}_t at every time instance t . The camera frame F^0 is assumed to be the reference frame. Let us note that γ is set as an upper bound of the objective function in problem (19). Consequently, we can say that $d_i^w = \|\frac{\Phi_i M}{P_{3i}^T M}\| < \gamma$ for $i = 1, \dots, N$. Assume that the optimal value γ^* is bounded between a known lower threshold γ^l and a higher one γ^h . Finally, the sequential bisection algorithm works as described in Algorithm 2.

The algorithm starts as soon as two views $N = 2$ are available. At each time instance t , the algorithm solves single convex feasibility problem of the form in (20) as one iteration of the quasi-convex triangulation problem given in (19). Whenever new image measurements are available in the next time instance, the objective function $L_\infty^w = \max_i (\|\frac{\Phi_i M}{P_{3i}^T M}\|)$ and the bounding thresholds are updated based on the new measurements. Here γ^h is the higher bound of the objective function, i.e. the L_∞^w norm. This γ^h should be equal or greater than the re-projection error $d_i^w = \|\frac{\Phi_i M}{P_{3i}^T M}\|$ in all the considered images. Thus, for the N^{th} observed image, if the re-projection error d_N is greater than the higher bound γ^h then, set this higher bound equal to the re-projection error d_N . Next is solving the convex feasibility problem in (20) once. This feasibility problem results in an update of the higher bound γ^h and the lower one γ^l based on whether the problem is feasible or not; as well as an estimate of the 3D point M with a re-projection error within the range (γ^l, γ^h) . Then, new convex feasibility problem is solved.

This process is repeated whenever new image is acquired providing new measurements. The solution of this problem is ob-

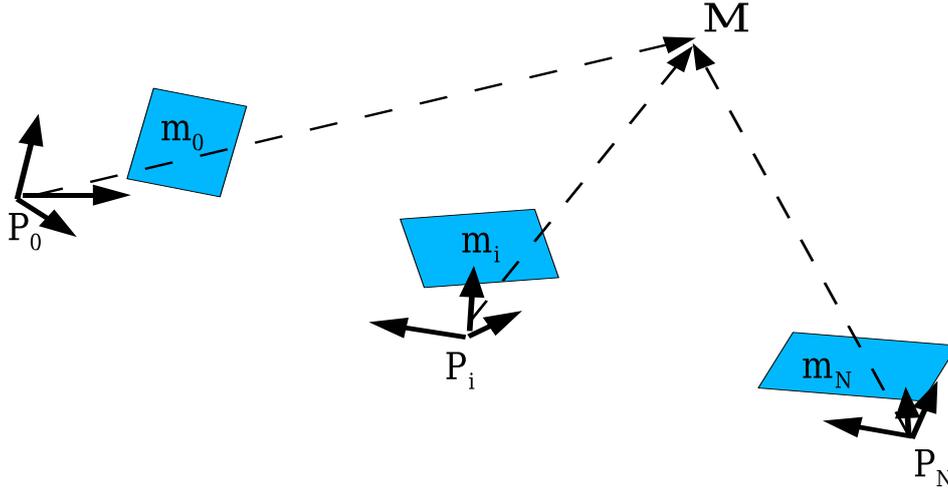


Fig. 4. The figure depicts the geometric reconstruction problem given N images along with corresponding N camera matrices \mathcal{P}_i . The objective is to compute the unknown 3D point coordinate M .

served to converge within 5-10 frames. Mapping process for another 3D feature begins to be solved within a few frames.

Algorithm 2 On-line Bisection algorithm to solve the Quasi-convex optimization Problem

- 1: **Input:** $N = 2$ image measurements, optimal value $f_0^*(M) \in [\gamma^l, \gamma^h]$ and tolerance ϵ . Initially, the time parameter t is set to N .
- 2: **Repeat**
 - (a) Collect the measurements from the N^{th} image.
 - (b) **If** $(d_N^w = \|\frac{\Phi_{N,M}}{P_{3N}^T M}\|) \geq \gamma^h$ **then**,

$$\gamma^h = \|\frac{\Phi_{N,M}}{P_{3N}^T M}\|.$$
 - (c) $\gamma = (\gamma^l + \gamma^h)/2$.
 - (d) Solve the convex feasibility problem as in (20).
 - (e) **If** feasible, $\gamma^h = d_N^w$,
else, $\gamma^l = \gamma$.
 - (f) $N = N + 1$
- 3: **Until** $\gamma^h - \gamma^l \leq \epsilon$

5. APPLICATIONS TO MAPPING IN VISUAL SLAM

The visual SLAM problem can be defined as the simultaneous real time estimation of the 3D pose of a moving camera and the structure of the environment. The pioneer work on visual SLAM has been presented by Davison Strasdat et al. (2012); Davison et al. (2007) using the Extended Kalman Filter (EKF). Their motion model using EKF utilizes the camera's state vector

$$\mathbf{x}_v = (\mathbf{r}, \mathbf{q}, \mathbf{v}, \mathbf{w})^T \quad (21)$$

This state vector consists of camera position \mathbf{r} , camera rotation \mathbf{q} , linear velocity \mathbf{v} , and rotational velocity \mathbf{w} . All are given with respect to a world frame. The state vector update is produced as

$$\mathbf{f}_v = (\mathbf{r}_{new}, \mathbf{q}_{new}, \mathbf{v}_{new}, \mathbf{w}_{new})^T = \mathbf{f}_v(\mathbf{x}_v, \dot{\mathbf{x}}_v, \mathbf{n}, \Delta t) \quad (22)$$

Here, Δt is the time interval and $\mathbf{n} = (\mathbf{V}, \mathbf{\Omega})$, these are the linear and rotational respectively. The uncertainty of the 3D estimates was used to constrain the search within an elliptical region in the image.

The feature initialization process in Davison et al. (2007); Strasdat et al. (2012) is based on a one dimensional particle filter to

represent the depth of a newly observed feature. The filter is a set of particles (hypotheses) that is uniformly distributed along the projection ray of the feature. The image measurements collected from the consecutive frames are used to iteratively update the distribution of the depth particles. The feature initialization process is completed by inserting the estimated 3D feature along with its uncertainty in the map. This insertion is done when the depth distribution has converged to Gaussian. One may note that the depth range is limited since the number of particles is determined in advance and cannot be chosen arbitrarily. Since the number of depth particles affect the performance in the real time, it is chosen in such a way to keep the computations at a minimum.

Since knowledge about the uncertainty of the depth is not available and no sufficient observations to estimate it, we propose a deterministic model. The 3D feature estimates are initialized using a robust and deterministic method built within convex optimization framework. In this case, an estimate of the depth can be sequentially available and consistent with the EKF framework due to the sequential triangulation algorithm proposed in this paper and given in Algorithm 2 presented in Sec. 4.

5.1 Searching for Features

Two methods of searching for features are adopted here to come up with the real-time active search method. The first one is used to search the image features corresponding to features currently being initialized only. The second method is used to search for image features that have been already initialized in the map.

The first method is based on the uncertain epipolar lines. The *a priori* motion distribution allows us to construct a distribution of the epipolar lines in the second image. This is subject to the availability of image measurements in the first frame. It was shown in Hartley and Zisserman (2003) that given a distribution of epipolar lines with mean \mathbf{l} and covariance $\Sigma_{\mathbf{l}}$, the region L that contains a certain portion of the epipolar lines can be analytically described. In other words, the features will be searched first in the initial two frames and then feature matching is performed. The search is limited to a region of epipolar lines given based on the motion model uncertainty.

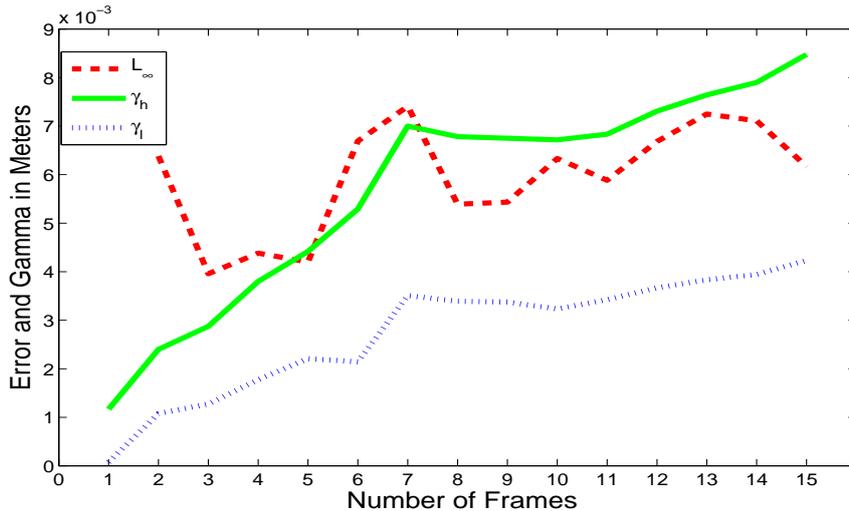


Fig. 5. The value of the initialization reprojection error over bisection iterations.

The second search method is used for already mapped features. In later time instances, the uncertain 3D features will be projected to the image to form a search region for expected feature measurement. The *a priori* knowledge about the 3D estimates of the features is projected to the image space using the proposal distribution of the motion posterior $P_{t+1/t}$. Each particle of the distribution $P_{t+1/t}$ will form measurement distribution in the image. The set of projections using the whole particles is approximated by a Gaussian distribution in the image. The region of search is determined by the ellipse with 3σ . This method is similar to the one presented in Davison et al. (2007)

6. EXPERIMENTAL RESULTS

In this section we show the results that demonstrate the efficiency of our proposed method. We show different kinds of results. First, we show preliminary results illustrating the on-line triangulation algorithm. Second, the application of the algorithm to mapping for VSLAM is shown.

6.1 Results from On-line Triangulation

We have carried out an experiment which is applied to data from real video sequence. The real video sequence was acquired with hand-held low cost Unibrain IEEE 1394 camera, with a 90deg field of view and 320x240 resolution monochrome at 30 fps. In fact this sequence is available on the web by the authors of Davison et al. (2007); Strasdat et al. (2012). In addition, we have used the Matlab code provided by them for both comparison of their mapping method with ours and for plugging our mapping algorithm to the VSLAM framework. In other words, many of our processing steps are same as in Davison et al. (2007); Strasdat et al. (2012) except the mapping stage. We compare with the work presented in Davison et al. (2007); Strasdat et al. (2012) since its source code of implementation is the only one available for use.

The on-line triangulation algorithm is applied to data collected from the mentioned video sequence. The camera matrices corresponding to the sequence images are collected from the localization stage output of the VSLAM framework. However, we present these camera matrices (camera pose with respect to

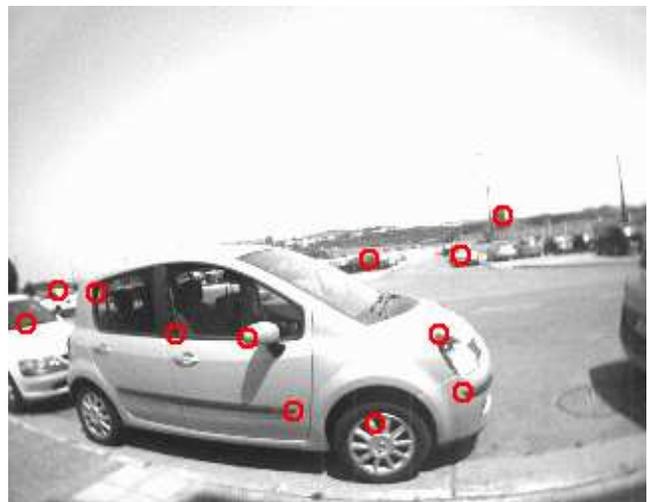


Fig. 6. A sample picture arbitrarily taken out from the considered video sequence.

the reference frame) to the triangulation algorithm assuming that they are correct estimates. The L_∞ error along with its boundaries γ^l and γ^h are illustrated in Fig. 5. The figure depicts the average value over all features. The error L^w is in dashed-red, the upper bound γ^h in dotted-blue, and the lower bound γ^l in (solid-green). It is clear that the L_∞ error converges to small values within the acceptable range of γ^l and γ^h . One may note that there are no notable variations of the estimated values after a few iterations.

6.2 Results from Mapping in VSLAM

To illustrate the applicability of our on-line triangulation method to VSLAM, we have applied it for mapping many features detected and extracted from the said video sequence. We have selected features that belong to two different classes, particularly far features that appear to be at infinity and near features that provide enough parallax. In addition, we compare the re-projection error for the two classes of features with the mapping method used in Montiel et al. (2006) which is the only available VSLAM implementation for use. Figure 6 shows the

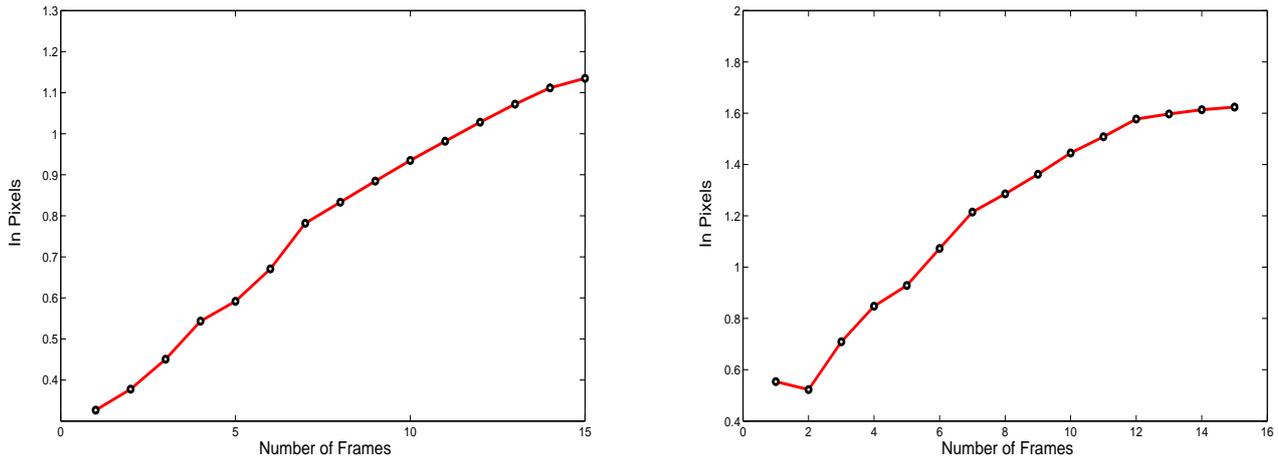


Fig. 7. The average of the re-projection Euclidean error for a selected set of near (far) features at the top (bottom) respectively. Errors that are shown above are in pixels.

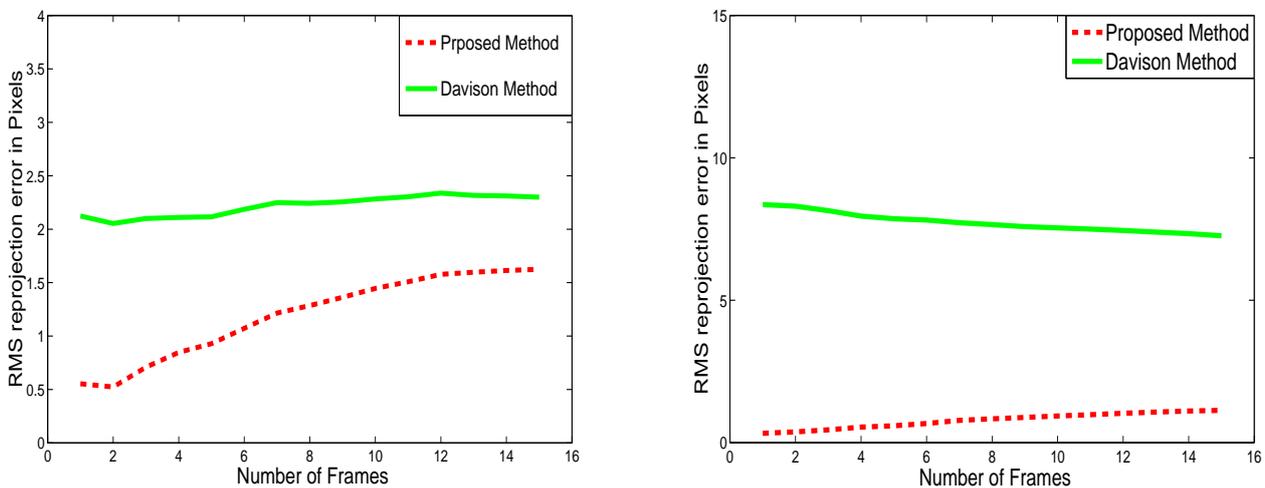


Fig. 8. The results of a comparison with Davison's work in Montiel et al. (2006). Ours in dotted-red color and the other in solid-green.

selected near and far features taken from the considered video sequence in Montiel et al. (2006), in which features are marked by a red circle.

The average re-projection error in pixels for near features is shown in Fig. 7(at the top). The average RMS error is less than one pixel and that is relatively small. Indeed, the mapping of the 3D features is precise and accurate. Similarly, the re-projection error in pixels for far features is shown in Fig. 7(at the bottom). The average is about three pixels. This means that the accuracy of the 3D estimates is much less for far features than the one for near features. This is due to the amount of parallax available from near features motion.

A comparison with the mapping results of the method presented in Montiel et al. (2006) are shown in Figure 8 for near and far features respectively. The average re-projection error for a selected set of far features, at the top, and selected set of near features, at the bottom during iterations of the bisection algorithm. In the case of near feature, our mapping method clearly outperforms the method in Montiel et al. (2006). They

show a notable improvement with smaller re-projection error in the image. Eventhough in far feature case where the accuracy of our estimate is less, it shows less re-projection error. The error of our mapping method is presented using a dashed line in red color while the error from the method presented in Montiel et al. (2006) is presented using a solid line in green color.

7. CONCLUSION

We have presented a sequential algorithm for solving a set of problems in multi-view geometry, the triangulation problem as a particular example with its application to the mapping problem in Visual SLAM . The algorithm provides an efficient solution to these problems in the real time. The algorithm begins with an unknown estimate and its output converges to the actual values within a few iterations. This algorithm is used to solve the mapping problem in VSLAM framework. It initializes features with the 3D information required before introducing the feature in the map. The experimental comparisons on mapping for VSLAM show that the proposed sequential algorithm

superpasses the most common mapping techniques like using particle filter.

ACKNOWLEDGEMENTS

The author would like to acknowledge Ruthanne Sikora and Borbala Toth for their help in revising the language of the paper.

REFERENCES

- Abdul Hafez, A.H., Bhuvanagiri, S., Krishna, K.M., and Jawahar, C.V. (2008). On-line convex optimization based solution for mapping in vslam. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, September 22-26, 2008 IROS'08*, 4072–4077. Nice, France.
- Abdul Hafez, A.H. and Jawahar, C.V. (2006). Target model estimation using particle filters for visual servoing. In *IEEE Int. Conference on Pattern Recognition, ICPR'06*. Hong Kong.
- Abdul Hafez, A.H. and Jawahar, C.V. (2008). Real time l_∞ -based solution to multi-view problems with application to visual servoing. In *IEEE International Conference on Information and Communication Technologies: From Theory to Applications, ICTTA'08*. Damasucs, Syria.
- Abdul Hafez, A.H., Nelakanti, A., and Jawahar, C.V. (2007). Path planning approach to visual servoing with feature visibility constraints: A convex optimization based solution. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'07*. San Diego, CA.
- Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.
- Chesi, G. (2009). Visual servoing path planning via homogeneous forms and lmi optimizations. *Trans. Rob.*, 25(2), 281–291.
- Cremers, D. and Kolev, K. (2011). Multiview stereo and silhouette consistency via convex functionals over convex domains. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(6), 1161–1174.
- Davison, A.J., Reid, I.D., Molton, N.D., and Stasse, O. (2007). Monoslam: Real-time single camera slam. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(6), 1052–1067.
- Flandin, G. and Chaumette, F. (2001). Visual data fusion: Application to object localization and exploration. Technical Report INRIA/RR-4168-FR+ENG, INRIA, Renne, France.
- Hartley, R. and Kahl, F. (2007). Optimal algorithms in multiview geometry. In Y. Yagi, S.B. Kang, I.S. Kweon, and H. Zha (eds.), *Computer Vision – ACCV 2007*, volume 4843 of LNCS, 13–34. Springer.
- Hartley, R. and Schaffalitzky, F. (2004). L_∞ minimization in geometric reconstruction problems. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR'04*, volume 01, 504–509. IEEE Computer Society, Los Alamitos, CA, USA.
- Hartley, R. and Zisserman, A. (2003). *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition.
- Kahl, F. (2005). Multiple view geometry and the L_∞ norm. In *Proceedings of the Tenth IEEE International Conference on Computer Vision, ICCV'05*, 1002–1009. Beijing, China.
- Ke, Q. and Kanade, T. (2005). Quasiconvex optimization for robust geometric reconstruction. In *Proceedings of the Tenth IEEE International Conference on Computer Vision, ICCV '05*, 986–993. Beijing, China.
- Ke, Q. and Kanade, T. (2007). Quasiconvex optimization for robust geometric reconstruction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(10), 1834–1847.
- Klodt, M., Steinbruecker, F., and Cremers, D. (2013). Moment constraints in convex optimization for segmentation and tracking. In *Advanced Topics in Computer Vision*. Springer.
- Montiel, J.M.M., Civera, J., and Davison, A.J. (2006). Unified inverse depth parametrization for monocular slam. In *Robotics: Science and Systems*.
- Quoc, T.D., Savorgnan, C., and Diehl, M. (2011). Real-time sequential convex programming for nonlinear model predictive control and application to a hydro-power plant. In *CDC-ECE*, 5905–5910. IEEE.
- Salzmann, M., Hartley, R., and Fua, P. (2007). Convex optimization for deformable surface 3-d tracking. In *Proceedings of the Tenth IEEE International Conference on Computer Vision, ICCV'07*. Rio de Janeiro, Brasil.
- Seo, Y. and Hartley, R. (2007). Sequential l_∞ norm minimization for triangulation. In Y. Yagi, S.B. Kang, I.S. Kweon, and H. Zha (eds.), *Computer Vision – ACCV 2007*, volume 4844 of LNCS, 322–331. Springer.
- Shi, J. and Tomasi, C. (1994). Good features to track. In *IEEE International Conference on Computer Vision and Pattern Recognition, CVPR'94*, 593–600. Seattle, Washinton.
- Sim, K. and Hartley, R. (2006). Removing outliers using the l_∞ norm. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'06*, 485–494. IEEE Computer Society, Washington, DC, USA.
- Strasdat, H., Montiel, J.M.M., and Davison, A.J. (2012). Visual slam: Why filter? *Image Vision Comput.*, 30(2), 65–77.
- Tran, D.Q., Savorgnan, C., and Diehl, M. (2013). Combining lagrangian decomposition and excessive gap smoothing technique for solving large-scale separable convex optimization problems. *Comp. Opt. and Appl.*, 55(1), 75–111.
- Wu, Y., Shen, B., and Ling, H. (2012). Online robust image alignment via iterative convex optimization. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 0, 1808–1814.