Modeling, Identification and Control, Vol. 45, No. 4, 2024, pp. 137–146, ISSN 1890–1328

Robust Camera Pose Estimation from Line Correspondences Using GNC

Hans K.R. Holen [1] Torbjørn Smith [1] Olav Egeland [1]

[1] *Department of Mechanical and Industrial Engineering, Norwegian University of Science and Technology (NTNU), N-7491 Trondheim, Norway. E-mail:* {hans.k.r.holen, torbjorn.smith, olav.egeland}@ntnu.no

Abstract

A solution to the Perspective-n-Lines (PnL) problem is proposed where a large fraction of outliers can be handled. The approach estimates camera pose from 2D-3D line correspondences where outliers are in the form of line mismatches. The solution is based on graduated non-convexity (GNC) with truncated least squares with Dynamical Pose Estimation (DAMP) as a solver. The solution is simple to implement and does not require specialized optimization software. The method is compared to 11 state-of-the-art PnL methods using synthetic and real data and evaluated in terms of accuracy, running time, and sensitivity to noise and outliers. The results show that our proposed method scores among the top for accuracy and robustness.

Keywords: PnL, PnP, Camera Pose Estimation, Dynamical Pose Estimation, GNC

1 Introduction

Pose estimation for a calibrated camera is a crucial task in many applications of robot vision. These include Augmented Reality (Wang et al., 2020), Simultaneous Localization and Mapping (SLAM) (Cadena et al., 2016) and navigation (Abdellali et al., 2019). A common way of estimating pose, that is, position and orientation, is based on the use of a 2D camera where features are detected in the image and matched with 3D features with a known position. In the well-known Perspective-n-Point (PnP) problem (Li et al., 2012), (Lepetit et al., 2009), (Ferraz et al., 2014), points are used as features in the pose estimation problem. In the Perspective-n-Lines (PnL) problem (Xu et al., 2017) the pose estimation problem is solved using line features. Lines will often appear in man-made structures (Coughlan and Yuille, 2003), and lines will typically contain rich geometric information that is not found in points features. Moreover, texture-less objects and scenes like indoor environments often have limited distinctive points, making it difficult to detect 2D-3D correspondences (Yu et al., 2020). In addition, lines are less affected by partial occlusion compared to points, making them more robust for applications that may experience occlusion due to weather or lighting conditions. Still, PnL remains a challenging topic due to its higher computational complexity.

Robust camera pose estimation starts with the identification of feature correspondences. Then outliers are found and eliminated. Finally, the inliers are used to compute the pose estimate. The usual method for outlier rejection is the RANSAC algorithm (Fischler and Bolles, 1981). The RANSAC algorithm randomly selects minimal subsets of feature correspondences so that a tentative pose estimate can be calculated for each subset. These tentative pose estimates are then tested on the whole dataset, and the pose estimate with the largest number of inlier correspondences is selected. RANSAC is effective for eliminating outliers, but it is non-deterministic (Antonante et al., 2022), does not guarantee optimality (Yang et al., 2020), and may be computationally inefficient for high outlier rates (Parra Bustos and Chin, 2018). To avoid potential

doi:10.4173/mic.2024.4.3

problems with RANSAC, the Algebraic Outlier Rejection (AOR) method was proposed in (Ferraz et al., 2014). AOR designates correspondences with large algebraic errors as outliers. Another method is consensus maximization (Chin and Suter, 2017), where the optimal solution maximizes the number of inlier correspondences where the residuals are below an inlier threshold. Graduated Non-Convexity (GNC) (Blake and Zisserman, 1987), (Black and Rangarajan, 1996) is also used for outlier rejection, where a robust cost like a truncated least squares is solved in a continuation method. Zhou et al. (2016) used GNC for point cloud registration, and Yang et al. (2020) used GNC on a wide range of registration problems with very good results. Line correspondences, however, were not included in their work.

Camera pose was estimated from line correspondences in (Přibyl et al., 2017) where Direct Linear Transformation (DLT) was used in a Plücker formulation, rejecting outliers with AOR. The paper contained a detailed comparison to previous work on robust camera pose estimation from line correspondences. In (Liu et al., 2021) the same problem was solved with Branch-and-Bound (BnB) for the estimation of rotation and RANSAC for the estimation of translation, and compared with the results of (Přibyl et al., 2017).

In this paper, we propose to use GNC for outlier rejection in camera pose estimation from line correspondences. The estimation problem in the GNC formalism is given as a sequence of nonlinear least-squares problems, where each least-squares problem is solved with Dynamical Pose Estimation (DAMP) (Yang et al., 2021). The excellent performance of the method is validated in tests on well-establish image benchmarks where our method is compared with the state-of-the-art PnL methods tested in (Přibyl et al., 2017) and (Liu et al., 2021) with respect to accuracy, computational time, and sensitivity to noise and outliers.

The contribution of the paper is an outlier rejection method for the Perspective-n-Lines problem based on GNC and Dynamical Pose Estimation, and the validation of the proposed method on image benchmarks in a comparison to state-of-the-art methods for Perspective-n-Lines.

The paper is organized as follows: The prior work is reviewed in section 2. Then the proposed method is explained, where section 3 presents line geometry, section 4 presents the outlier rejection method GNC, and section 5 presents the pose estimation algorithm DAMP. Finally, experiments with synthetic and real-world data are presented in section 6, and the conclusion is presented in section 7.

2 Related work

This section will present related work on PnL algorithms: Locally iterative, Algebraic, and Linearized solvers.

Iterative PnL solutions

The PnL problem is formulated as a nonlinear least squares problem solved by an iterative minimization of an error function. Liu et al. (Liu et al., 1990) conducted some of the earliest work by developing the R_then_T method where the rotation and position are solved separately. Kumar et al. (Kumar and Hanson, 1994) developed the R_and_T algorithm that simultaneously estimates rotation and position. Zhang et al. (Zhang et al., 2016) modified the method by exploiting the uncertainty properties of line segment endpoints. A problem with these methods is that they often converge slowly and sometimes fail to produce an accurate pose estimate due to local minima. Because of outliers and noise, the objective in the optimization function will often be non-convex; hence careful initialization is needed. Therefore most of these approaches are more suitable for a final refinement or tracking applications than for PnL problems without prior knowledge. These approaches are often nested in a RANSAC framework for robustness. Methods such as David et al. (David et al., 2003) and Zhang et al. (Zhang et al., 2012) present an approach that simultaneously conducts correspondence filtering and pose estimation.

Algebraic solutions

The PnL problem is solved with a system of equations, usually polynomial, by minimizing an algebraic error. Initialization is optional for these approaches. However, the solutions are not necessarily geometrically optimal if some correspondences are outliers. These methods are therefore combined with a RANSAC scheme to obtain an accurate result. Ansar and Daniilidis (2003) developed the first method to handle more than four lines, limiting possible solutions to only one. This approach has a quadratic computational complexity and becomes unstable with increasing noise. Mirzaei and Roumeliotis (2011) proposed a method that is more computationally efficient ($O(n)$) which can handle a minimum of three lines. A disadvantage of this method is that it may produce multiple solutions. Zhang et al. (2013) proposed the Robust PnL (RPnL) algorithm, which is more robust and accurate than (Mirzaei and Roumeliotis, 2011), but less computationally efficient. Later the RPnL algorithm was developed into the more efficient Accurate Subset-based PnL (ASPnL) method by Xu et al. (2017), which

gives more accurate results on small sets of line correspondences, but is sensitive to outliers. Zhou et al. (2021) recently proposed a method that applies the Gram-Schmidt process to solve a quadratic system of equations with the same form as the minimal problem. It produces good results with a low number of line correspondences.

Recently, Liu et al. (2021) proposed a method to determine the globally optimal camera orientation given outlier contaminated data. The method first solves the camera rotation by applying the Branch-and-Bound algorithm (Morrison et al., 2016), then RANSAC to estimate the camera position.

Linearized PnL solutions

Linearized PnL solutions solve the PnL problem using a homogeneous system of linear equations, where the size is proportional to the number of line correspondences. The most significant advantage of linearized PnL methods is their computational efficiency, making them fast regardless of the number of line correspondences. The earliest and most straightforward method of linearized PnL solutions is the Direct Linear Transformation (DLT) algorithm (Hartley and Zisserman, 2003). Silva et al. (2012) developed the method into DLT-Lines, using 3D points on the 3D lines, and not the lines directly. More recently, Přibyl et al. (2016) developed a DLT method that acts on 3D lines directly by applying Plücker coordinates. Then the method was expanded and combined with the Direct Linear Transformation algorithm to DLT-Combined-Lines (Přibyl et al., 2017), where the 2D structures are represented as lines and the 3D structures by both lines and points. Xu et al. (2017) released a set of solutions to the PnL problem, similar to DLT-Lines, using 2D lines and 3D points. The methods differ by whether Cartesian or barycentric coordinates parameterize the 3D points and how the solution is obtained from the null space. Either the solution is obtained from a closed-form solution using a homogeneous linear least square or it is estimated the same way as with EPnP (Lepetit et al., 2009). More recently, Wang et al. (2019) proposed a direct least-squares solution for the PnL problem of a multi-camera system.

These methods can be combined with Algebraic Outlier Rejection (AOR) to handle outliers, as proposed by Ferraz et al. (2014) and applied in (Přibyl et al., 2017) and (Přibyl et al., 2016). AOR can be implemented directly into the pose estimation procedure, where weights are reweighted iteratively by solving a least squares problem. This method is efficient, but usually has a breaking point when the rate of outliers reaches a specific limit.

3 Line Geometry

Let the homogeneous transformation matrix from the camera frame c to the world frame w be given by $\boldsymbol{T}_c = (\boldsymbol{R}_c, \boldsymbol{t}_c) \in SE(3)$. Consider a line given by the points $\boldsymbol{x} = \boldsymbol{x}_0 + \alpha \boldsymbol{a}$ in the world frame w, and by the points $\boldsymbol{y} = \boldsymbol{y}_0 + \beta \boldsymbol{b}$ in the camera frame c. Here $\boldsymbol{a}$ and $\boldsymbol{b}$ are the direction vectors, and α and β are scalar line parameters. It is noted that the points $\boldsymbol{x}_0$ and $\boldsymbol{y}_0$ are not necessarily corresponding points. However, for any point $\boldsymbol{x}_j = \boldsymbol{x}_0 + \alpha_j \boldsymbol{a}$ on the line in the world frame, there is a corresponding point on the same line in the camera frame given by $\boldsymbol{y}_j = \boldsymbol{y}_0 + \beta_j \boldsymbol{b}$ so that

$$\boldsymbol{y}_j = \boldsymbol{R}_c \boldsymbol{x}_j + \boldsymbol{t}_c \tag{1}$$

The line in the camera frame can be written in Plücker coordinates (Pottmann and Wallner, 2001) as

$$\boldsymbol{L} = (\boldsymbol{b}, \boldsymbol{m}) \tag{2}$$

where $\boldsymbol{m} = \boldsymbol{y}_0 \times \boldsymbol{b}$ is the moment of the line. It follows that the plane through the origin of the camera frame and the line $\boldsymbol{L}$ is given in homogeneous Plücker coordinates as $(0, \boldsymbol{n})$, where $\boldsymbol{n} = \boldsymbol{m}/\|\boldsymbol{m}\|$ is the unit normal vector.

Let the image of the 3D line $\boldsymbol{L}$ be the homogeneous 2D line $\boldsymbol{\ell}$ in the normalized image plane. Then the vector $\boldsymbol{n}$ is found from $\boldsymbol{\ell} = \boldsymbol{n}$. This is shown by considering two points $\boldsymbol{y}_a$ and $\boldsymbol{y}_b$ on the line $\boldsymbol{L}$ which correspond to the homogeneous 2D points $\boldsymbol{s}_a = \boldsymbol{y}_a/\|\boldsymbol{y}_a\|$ and $\boldsymbol{s}_b = \boldsymbol{y}_b/\|\boldsymbol{y}_b\|$ in the normalized image plane. The homogeneous line in the normalized image plane is $\boldsymbol{\ell} = \boldsymbol{s}_a \times \boldsymbol{s}_b = \gamma_1 \boldsymbol{y}_a \times \boldsymbol{y}_b = \gamma_2 \boldsymbol{n}$, where γ_1 and γ_2 are scalar constants. The scaling can be selected freely, and it follows that $\boldsymbol{\ell} = \boldsymbol{n}$.

The distance from a point $\boldsymbol{z}$ to the plane is $\boldsymbol{n}\boldsymbol{n}^T\boldsymbol{z}$. Suppose that the estimate of the homogeneous transformation matrix is $\boldsymbol{T} = (\boldsymbol{R}, \boldsymbol{t}) \in SE(3)$. Then an estimate of the point $\boldsymbol{y}_j$ corresponding to a point $\boldsymbol{x}_j$ in the world frame will be

$$\boldsymbol{y}_j = \boldsymbol{R}\boldsymbol{x}_j + \boldsymbol{t} \tag{3}$$

which has the distance $\boldsymbol{n}\boldsymbol{n}^T(\boldsymbol{R}\boldsymbol{x}_j + \boldsymbol{t})$ from the plane of the line.
The residual for this estimate is then given by

$$\boldsymbol{r}(\boldsymbol{n}, \boldsymbol{x}_j, \boldsymbol{T})^2 = \|\boldsymbol{n}\boldsymbol{n}^T(\boldsymbol{R}\boldsymbol{x}_j + \boldsymbol{t})\|^2 \tag{4}$$

The pose can then be found from N line correspondences $(\boldsymbol{L}_j, \boldsymbol{\ell}_j)$ by solving the minimization problem

$$\min_{\boldsymbol{T}} \sum_{i=1}^{2N} \boldsymbol{r}^2(\boldsymbol{n}_i, \boldsymbol{x}_i, \boldsymbol{T}) \tag{5}$$

where the residual is

$$\boldsymbol{r}^2(\boldsymbol{n}_i, \boldsymbol{x}_i, \boldsymbol{T}) = \|\boldsymbol{n}_i \boldsymbol{n}_i^T (\boldsymbol{R}\boldsymbol{x}_i + \boldsymbol{t})\|^2 \tag{6}$$

and two points $\boldsymbol{x}_{2j-1}$ and $\boldsymbol{x}_{2j}$ are selected for each line j, $j = 1, \ldots, N$, and where $\boldsymbol{n}_{2j-1} = \boldsymbol{n}_{2j}$ is the normal vector of plane j.

4 Graduated Non-Convexity

In Graduated Non-Convexity (GNC) the least-squares cost $\boldsymbol{r}_i^2$ in eq. (5) is replaced with a robust cost $\rho(\boldsymbol{r}_i)$ like the truncated least-squares (TLS) cost $\rho_{\text{TLS}}(\boldsymbol{r}_i) = \min\left(\boldsymbol{r}_i^2, \epsilon^2\right)$, where ϵ is the truncation threshold. This gives a non-convex truncated least-squares problem. Graduated non-convexity can then be used in a GNC-TLS continuation process (Black and Rangarajan, 1996), (Yang et al., 2020) by minimization of the surrogate cost function

$$\rho_\mu(\boldsymbol{r}_i) = w_i \boldsymbol{r}_i^2 + \frac{\mu(1-w_i)}{\mu + w_i}\epsilon^2 \tag{7}$$

with weights given by

$$w_i = \begin{cases} 1, & \text{if } \boldsymbol{r}_i^2 \leq \epsilon^2 \alpha_1 \\ \frac{\epsilon}{|\boldsymbol{r}_i|}\sqrt{\mu(\mu+1)} - \mu, & \text{if } \epsilon^2\alpha_1 \leq \boldsymbol{r}_i^2 \leq \epsilon^2\alpha_2 \\ 0, & \text{if } \boldsymbol{r}_i^2 \geq \epsilon^2\alpha_2 \end{cases} \tag{8}$$

where $\alpha_1 = \mu/(\mu+1)$ and $\alpha_2 = (\mu+1)/\mu$. The surrogate cost function $\rho_\mu(\boldsymbol{r}_i)$ will tend to a convex function when μ tends to zero (Smith and Egeland, 2022). As noted in (Yang et al., 2020) the second term of the surrogate cost function in eq. (7) is independent of the pose, and it follows that the GNC-TLS problem defined by eq. (5) and eq. (7) is equivalent to the weighted least-squares problem

$$\min_{\boldsymbol{T}} \sum_{i=1}^{2N} w_i \boldsymbol{r}^2(\boldsymbol{n}_i, \boldsymbol{x}_i, \boldsymbol{T}) \tag{9}$$

with weights w_i given by eq. (8) as functions of μ.

The optimization problem can then be solved in a continuation process. The first iteration is run with $\{w_i = 1\}_{i=1}^{2N}$, and the initial value of μ is set to $\mu = \epsilon^2/(2\boldsymbol{r}_{\max}^2 - \epsilon^2)$ where $\boldsymbol{r}_{\max}$ is the largest residual after the first iteration (Yang et al., 2020). Then a sequence of problems eq. (9) are solved where μ is increased to $\sqrt{2}\mu$ for each new step of the sequence until the surrogate cost function is sufficiently close to the TLS cost. It is noted that the weights will converge to $w_i = 0$ for outliers, and $w_i = 1$ for inliers, so that only inliers will contribute to the cost eq. (9) in the final solution. Further details on the method are found in (Antonante et al., 2022).

5 Dynamical Pose Estimation

DAMP was proposed by Yang et al. (2021) for a range of registration problems, and the method was extended to handle outliers with GNC in (Smith and Egeland, 2022). It was shown that the method could be used for camera pose estimation. The advantage of DAMP is that it is simple to implement and requires no specialized optimization software. It was shown in(Yang et al., 2021) and (Smith and Egeland, 2022) that the method's performance is comparable to alternative methods.

In DAMP, all the geometric features in the world frame w are fixed to a rigid body $\mathcal{B}_x$. The transformation $\boldsymbol{T}_c = (\boldsymbol{R}_c, \boldsymbol{t}_c) \in SE(3)$ is defined as the transformation from the c frame to the w frame in the c frame. The rigid body $\mathcal{B}_x$ is placed in frame c with the same pose as in frame w. In DAMP the geometric 2D-3D correspondences are represented by virtual springs and dampers which move the rigid body to the pose in c which corresponds to the image data. The resulting displacement in frame c is the transformation $\boldsymbol{T}_c$. In the case of line correspondences, each line is assigned two mass points. Each mass point is connected with a virtual spring and damper to the closest point on the corresponding plane, as given by the image data. This is illustrated in Figure 1. The parameters of the virtual rigid body system are selected so that the system's potential energy is equal to the loss function eq. (9) of the GNC pose estimation problem. Due to the stability properties of the virtual mass-spring-damper system, the potential energy is minimized, and it follows that the loss function eq. (9) is minimized for the resulting displacement $\boldsymbol{T}$ of the rigid body $\mathcal{B}_x$. GNC scales the individual mass, spring, and damper con-

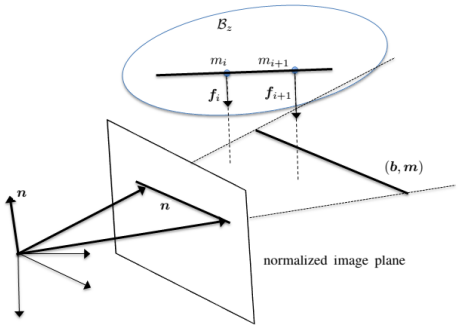


Figure 1: Illustration of the plane spanned by the camera center and the measured 2D line, and its corresponding 3D line and its two endpoints. The two mass points m_i and m_{i+1} on the 3D line in the rigid body $\mathcal{B}_z$ are connected to the plane with a spring and a damper, which give the forces $\boldsymbol{f}_i$ and $\boldsymbol{f}_{i+1}$.

stants with the weighting parameters $w_i \in [0, 1]$.

The DAMP method for the camera pose estimation problem with GNC is implemented as follows. Consider N line correspondences where N 3D lines L_i in the world frame correspond to N 2D lines $\boldsymbol{\ell}_i = \boldsymbol{n}_i$ in the normalized image plane, where $\boldsymbol{n}_i$ is a unit normal vector. The N lines L_i are assumed to be fixed in a rigid body $\mathcal{B}_x$. This rigid body consists of $2N$ mass points that are fixed in the body. Each mass point has position $\boldsymbol{x}_i$ and mass m_i, and the points are selected so that two mass points are attached to each line in the rigid body.
The center of mass of the rigid body is given in the spatial frame w as

$$\bar{\boldsymbol{x}} = \frac{1}{M} \sum_{i=1}^{2N} m_i \boldsymbol{x}_i \tag{10}$$

where $M = \sum_{i=1}^{2N} m_i$ is the total mass, and the position of mass point i is given by

$$\boldsymbol{x}_i = \bar{\boldsymbol{x}} + \boldsymbol{x}_{r_i} \tag{11}$$

A moving virtual rigid body $\mathcal{B}_z$ with a fixed body frame b is defined as the copy of $\mathcal{B}_x$ so that the two rigid bodies are initially aligned. The displacement from $\mathcal{B}_x$ to $\mathcal{B}_z$ is given by $\boldsymbol{R}_z, \boldsymbol{t}_z$. The position of a mass point i in $\mathcal{B}_z$ is given by

$$\boldsymbol{z}_i = \boldsymbol{R}_z \boldsymbol{x}_i + \boldsymbol{t}_z \tag{12}$$

The center of mass is given by

$$\boldsymbol{z}_c = \frac{1}{M} \sum_{i=1}^{2N} m_i \boldsymbol{z}_i \tag{13}$$

The force $\boldsymbol{f}_i$ acting on the point mass at $\boldsymbol{z}_i$ is given by

$$\boldsymbol{f}_i = m_i k_p \boldsymbol{n}_i \boldsymbol{n}_i^T \boldsymbol{z}_i - m_i k_d \dot{\boldsymbol{z}}_i \tag{14}$$

where the first term is the spring force with spring constant $m_i k_p$ and extension equal to the distance from $\boldsymbol{z}_i$ to the plane with normal vector $\boldsymbol{n}_i$. The second term is the damping force with damping constant k_d. The spring constant is set to $k_p = (2\pi)^2$, and the damping constant is $k_d = 2\sqrt{k_p}$.

The motion of the virtual rigid body $\mathcal{B}_z$ is described by the position $\boldsymbol{z}_c(t)$ and velocity $\boldsymbol{v}_c(t)$ of the center of mass, the rotation matrix $\boldsymbol{R}_z(t)$ and the angular velocity $\boldsymbol{\omega}(t)$. The initial conditions are $\boldsymbol{z_c}(0) = \bar{\boldsymbol{x}}$, $\boldsymbol{v}_c(0) = \boldsymbol{0}$, $\boldsymbol{R}_z(0) = \boldsymbol{I}$ and $\boldsymbol{\omega}(0) = \boldsymbol{0}$. The equations of motion for the virtual rigid body $\mathcal{B}_z$ are given by

$$\dot{\boldsymbol{z}}_c = \boldsymbol{v}_c \tag{15}$$

$$\dot{\boldsymbol{R}}_z = \boldsymbol{R}_z \boldsymbol{\omega}^\times \tag{16}$$

$$\dot{\boldsymbol{v}}_c = \boldsymbol{a}_c = \frac{1}{M} \boldsymbol{f} \tag{17}$$

$$\dot{\boldsymbol{\omega}} = \boldsymbol{\alpha} = \boldsymbol{J}^{-1} (\boldsymbol{\tau} - \boldsymbol{\omega}^\times \boldsymbol{J} \boldsymbol{\omega}) \tag{18}$$

where $\boldsymbol{f}$ is the force, $\boldsymbol{\tau}$ is the torque, and the operator $^\times$ is defined as the outer product.The torque $\boldsymbol{\tau}$ is given by

$$\boldsymbol{f} = \sum_{i=1}^{2N} \boldsymbol{f}_i \tag{19}$$

$$\boldsymbol{\tau} = \sum_{i=1}^{2N} m_i (\boldsymbol{x}_{r_i}^\times (\boldsymbol{R}_z^T \boldsymbol{f}_i)) \tag{20}$$

where the moment of inertia is

$$\boldsymbol{J} = -\sum_{i=1}^{2N} m_i (\boldsymbol{x}_{r_i})^\times (\boldsymbol{x}_{r_i})^\times \tag{21}$$

Stability of the virtual dynamical system can be established with Lyapunov analysis using energy arguments, which was was done in (Smith and Egeland, 2022) for point cloud registration. The analysis shows that the potential energy

$$V_p = \frac{1}{2} \sum_{i=1}^{2N} m_i k_p \left(\boldsymbol{n}_i \boldsymbol{n}_i^T (\boldsymbol{R}_z \boldsymbol{x}_i + \boldsymbol{t}_z) \right)^2 \tag{22}$$

will be minimized. It is seen that if the mass is selected as $m_i = w_i$, then the minimization of eq. (22) is equivalent to the minimization of the GNC cost function eq. (9) (Smith and Egeland, 2022). This shows that the DAMP can be used as a solver for this GNC problem.

5.1 Escape equilibrium

DAMP terminates when the stopping criterion $\|\dot{\boldsymbol{v}}_c, \dot{\boldsymbol{\omega}}_c\| \leq \gamma$ is met. To prevent a suboptimal solution, the DAMP framework allows a simple scheme (Yang et al., 2021) to escape unstable minima. As the stopping criterion is met, the solution is saved, and a perturbation is added to $\boldsymbol{v}_c$ and $\boldsymbol{\omega}$ before DAMP runs again. This is done a set number of times, and the solution with the lowest potential energy is returned as the final pose estimate.

6 Experiments

In this section, we have compared our proposed method with several state-of-the-art PnL methods, where each method is combined with an outlier rejection method. The methods are the same as the ones compared by Liu et al. (2021) and Přibyl et al. (2017). Synthetic and real image data was used, and the accuracy of the pose estimates, computational time, and robustness to noise and outliers were compared. The accuracy of the pose estimate is given in terms of the rotation error and the translation error of the camera. The rotation error

$e_r = \arccos(\frac{1}{2}(\text{trace}(\boldsymbol{R}_{GT}^T \boldsymbol{R}_e) - 1))$ is the rotation angle between the ground truth $\boldsymbol{R}_{GT}$ and the estimate $\boldsymbol{R}_e$. The translation error $e_t = \|\boldsymbol{t}_{GT} - \boldsymbol{t}_e\|$, is the length of the vector from the ground truth $\boldsymbol{t}_{GT}$ to the estimated $\boldsymbol{t}_e$. The success rate is the ratio of successful runs to the total number of runs. A run is considered successful if $e_r \leq 2^\circ$ and $e_t \leq 2$m.

All experiments were run on a computer with an Intel Core i7-1065G7 CPU and 32GB RAM, and all the methods[1] were implemented in Matlab2021a.

The state-of-the-art methods that are compared to our solution are:

Ansar MLESAC4 RPnL: A system of linear equations is constructed from the 2D-3D line correspondences, and the pose is obtained using a least-squares approach (Ansar and Daniilidis, 2003). RPnL computes the final solution (Zhang et al., 2013) and MLESAC detects outliers (Torr and Zisserman, 2000).

Mirzaei MLESAC: A set of nonlinear polynomial equations are solved by implementing the Grobner basis polynomial solver (Mirzaei and Roumeliotis, 2011). MLESAC (Torr and Zisserman, 2000) detects the outliers.

RPnL MLESAC4: Lines are divided into triplets to form a sixteenth-order cost function, then the optimum is retrieved from the roots of the derivative of the cost function by evaluating the orthogonal errors and the reprojected errors of the local minima (Zhang et al., 2013). The algorithm is fused with MLESAC (Torr and Zisserman, 2000) for outlier detection.

P3L RANSAC3: Three spatial lines are used to solve the PnL problem. The rotation matrix is found by exploiting the geometric constraints between the lines by solving nonlinear equations of 8th-order polynomials. Then translation is obtained by solving linear equations (Xu et al., 2017). P3L is used with RANSAC (Hartley and Zisserman, 2003) to detect outliers.

ASPnL RANSAC4: Lines are separated into triplets by selecting a rotation axis, then a 16th-order cost function is constructed from a set of P3L polynomials, and an optimum solution is retrieved from its local minima (Xu et al., 2017). Outliers are removed using RANSAC.

LPnL_Bar_LS AOR: Lines are parameterized with barycentric coordinates and solved by the least square solver as proposed by Xu et al. (2017) and AOR is applied for outlier rejection.

LPnL_Bar_ENull AOR: Lines are parameterized with barycentric coordinates and solved by the null space solver as proposed by Xu et al. (2017). AOR is applied for outlier rejection.

DLT-Lines AOR: Direct Linear Transformation is used to recover a combined projection matrix. This method is a linear formulation of the PnL problem, combined with Algebraic Outlier Rejection as proposed in (Přibyl et al., 2017).

DLT-Plücker-Lines AOR: Direct Linear Transformation is used to solve the PnL problem using lines in the form of Plücker coordinates (Přibyl et al., 2016) fused with AOR.

DLT-Combined-Lines AOR: DLT-Lines and DLT-Plücker-Lines are combined to solve the PnL problem (Přibyl et al., 2017) and fused with AOR.

RO_PnL: As proposed by Liu et al. (2021), rotation is first obtained by applying the Branch-and-Bound algorithm and then the translation is estimated using RANSAC.

DAMP_PnL GNC: The proposed PnL algorithm, which applies DAMP (Yang et al., 2021) to obtain pose and GNC (Yang et al., 2020) for outlier detection. The number of GNC iterations is set to 20, and the truncation threshold is $\epsilon = 0.3$. The maximum number of time steps for each DAMP iteration is 5 000 and the stopping criterion is $\gamma = 10^{-5}$. Escape Equilibrium is activated with a perturbation of $\boldsymbol{v}(3)_t = \boldsymbol{v}(3)_{t-1} + 4$, and is allowed to run four times before DAMP terminates.

The methods combined with RANSAC or MLESAC were limited to 10 000 trials. The inlier threshold was set to 1° in the RANSAC-based methods.

6.1 Synthetic Data Experiments

The synthetic data was generated with $N = 500$ line segments defined by $2 \times N$ randomly generated endpoints in a box of $10 \times 10 \times 10$ meters centered around the origin. A virtual perspective camera was used with an image size of 640×480 pixels and a focal length of 800 pixels. All lines were normalized and projected to the camera coordinate frame. The camera was pointed to the origin of the world frame at a distance of 25 meters from a random position. Gaussian noise with a standard deviation of 2 pixels was added to each endpoint in the simulated image. Outliers were made by adding Gaussian noise with a standard deviation of 100 pixels to the respective endpoints. Each method was tested with an outlier rate ranging from 10% to 70%, and ran 100 times for every outlier rate.

The results of the synthetic experiments are shown in Figure 2. The rotation error for our method, shown in Figure 2a, is the best for a 0.1 outlier rate, among the top two for four outlier rates, and in the top three for all outlier rates. The median rotation errors are all lower than 0.2 degrees, and range from 0.05 degrees for a 0.1 outlier rate to 0.2 for a 0.7 outlier rate. The translation error for our method is among the top four for all outlier rates. The method estimates a camera position with a median error of less than 0.14 m for all outlier

[1] http://www.fit.vutbr.cz/∼ipribyl/DLT-based-PnL/

Table 1: Example images from each sequence in the real dataset.

Sequence	Abbreviation	#images	lines
Model House	MH	10	30
Corridor	COR	11	69
Merton College I	MCI	3	295
Merton College II	MCII	3	302
Merton College III	MCIII	3	177
Wadham College	WDC	5	380

rates, as shown in Figure 2c. Figure 2b illustrates the median run time and shows little change depending on the outlier rate, and a median run time of 2.89 seconds for 70% outliers. Figure 2d shows that our method maintains robustness, showing a success rate of 97% or better for all outlier rates. DLT-Combined-Lines AOR (Přibyl et al., 2017) and LPnL_Bar_ENull AOR (Xu et al., 2017) produce comparably accurate results, but break at a 0.7 outlier rate and a 0.5 outlier rate, respectively. LPnL_Bar_LS AOR Xu et al. (2017), DLT-Lines AOR (Přibyl et al., 2017), and RO_PnL (Liu et al., 2021) maintain robustness, but are not as accurate as our method. Mirzaei MLESAC (Mirzaei and Roumeliotis, 2011) achieves comparable median accuracy in translation to our method, but achieves a lower success rate.

6.2 Real Data Experiments

A dataset was retrieved from the Visual Geometry Group at Oxford University[2]. The dataset contains sequences of 2D-3D line correspondences from pictures of various buildings. Table 1 summarizes the characteristics of images sequences in the dataset. All methods were run on the six different data sequences, and no additional outliers were added; hence the noise level in each image is unknown.

The results are shown in Figure 3. Our proposed method has the best median accuracy in three of the sequences and is among the top two for five of the sequences. As in the synthetic data case, the LPnL_Bar_ENull AOR (Xu et al., 2017) method shows excellent results for the real data; however, the most significant difference in error between this method and our method is 0.146° for the rotation and 0.0773m for the translation, which validates the high performance of our method.

[2]https://www.robots.ox.ac.uk/∼vgg/data/mview/

7 Conclusion

We have proposed a new method to solve the PnL problem with noisy and faulty line correspondence data. The proposed method is one of the best overall, as it is among the best methods for accuracy and the best methods for robustness, achieving a 97% success rate for the 0.7 outlier rate as shown in Figure 2d. The experiments using real data also show promising results, as the median errors are the most accurate in three out of six cases and the top two in five out of six. Figure 2 shows that DLT-Combined-Lines AOR (Přibyl et al., 2017) and LPnL_Bar_ENull AOR (Xu et al., 2017) are comparably accurate, but are not as robust as our method. LPnL_Bar_LS AOR (Xu et al., 2017), DLT-Lines AOR (Přibyl et al., 2017), and RO_PnL (Liu et al., 2021) maintain the same robustness as our method, but fail to produce the same level of accuracy, illustrated in both the synthetic results in Figure 2 and the real data results in Figure 3.

The running time is comparable to the method of Liu et al. (2021). As Figure 2b shows, the other methods are faster, but we still argue that our method compensates for this with high robustness and accuracy. With a good initial guess for rotation and position, the running time would be significantly improved and comparable to the faster methods. Hence, applications such as object tracking, visual navigation, or SLAM would make a great fit. The robustness of our method is also important, as it is vital in real-world applications.

Acknowledgments

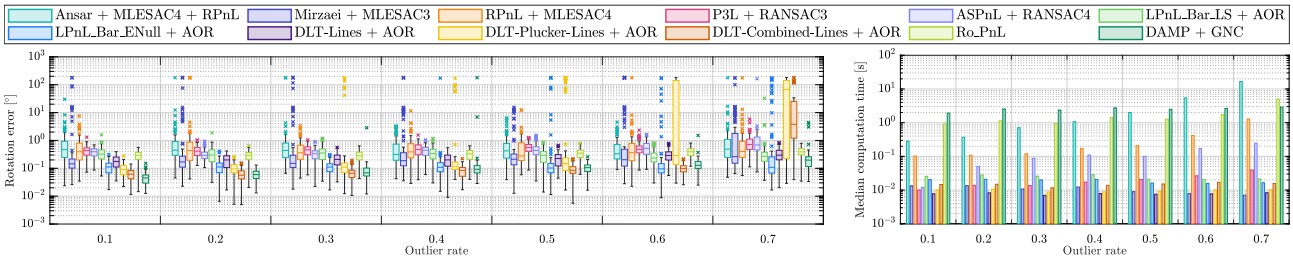
The research presented in this paper was funded by the Norwegian Research Council under Project Number 237896, SFI Offshore Mechatronics.

References

Abdellali, H., Frohlich, R., and Kato, Z. A direct least-squares solution to multi-view absolute and relative pose from 2D-3D perspective line pairs. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. pages 2119–2128, 2019. doi:10.1109/ICCVW.2019.00267.

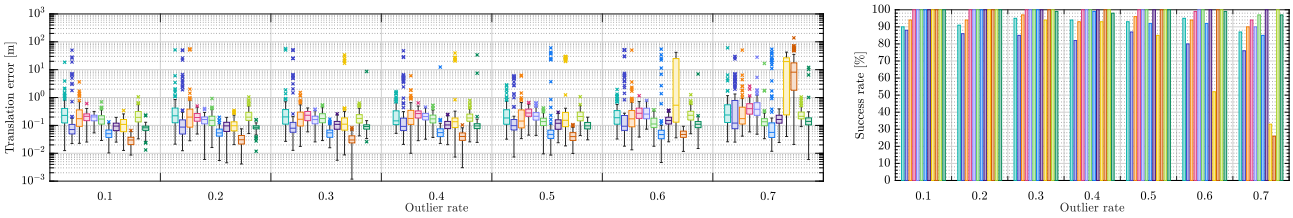
Ansar, A. and Daniilidis, K. Linear pose estimation from points or lines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003. 25(5):578–589. doi:10.1109/TPAMI.2003.1195992.

Antonante, P., Tzoumas, V., Yang, H., and Carlone, L. Outlier-robust estimation: Hardness, minimally



(a) Box charts showing rotation errors for the synthetic experiments. Each box represents 100 Monte Carlo runs for the given outlier rate.

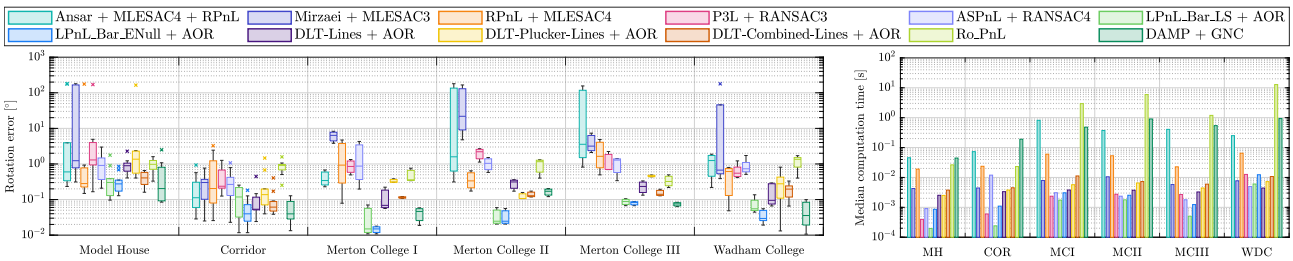
(b) Bar charts showing the median computation times for the synthetic experiments.



(c) Box charts showing translation errors for the synthetic experiments. Each box represents 100 Monte Carlo runs for the given outlier rate.

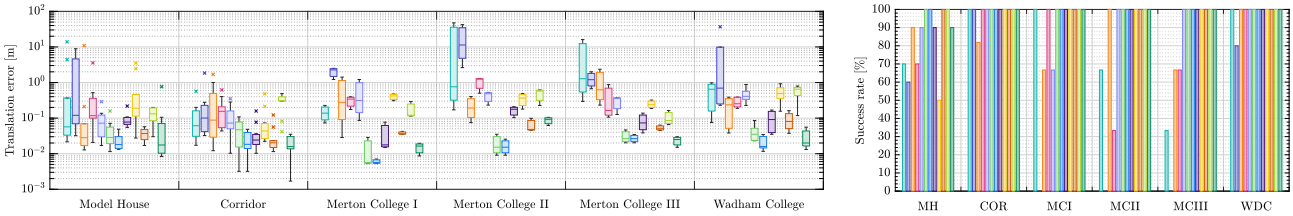
(d) Bar charts showing the success rates for the synthetic experiments.

Figure 2: Synthetic data experiment results generated across 100 Monte Carlo runs for each method at each outlier rate. The box charts show the sample median, upper and lower quartiles, nonoutlier range, and outliers more than 150% of the interquartile range removed from the upper or lower quartiles.



(a) Box charts showing rotation errors for the real data experiments.

(b) Bar charts showing the median computation times for the real data experiments.



(c) Box charts showing translation errors for the real data experiments.

(d) Bar charts showing the success rates for the real data experiments.

Figure 3: Real data experiment results. The box charts show the sample median, upper and lower quartiles, nonoutlier range, and outliers more than 150% of the interquartile range removed from the upper or lower quartiles.

tuned algorithms, and applications. *IEEE Transactions on Robotics (T-RO)*, 2022. 38(1):281–301. doi:10.1109/TRO.2021.3094984.

Black, M. J. and Rangarajan, A. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *Intl. J. of Computer Vision*, 1996. 19(1):57–91. doi:10.1007/BF00131148.

Blake, A. and Zisserman, A. *Visual Re-*

construction. The MIT Press, 1987. doi:10.7551/mitpress/7132.001.0001.

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., and Leonard, J. J. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on Robotics*, 2016. 32(6):1309–1332. doi:10.1109/TRO.2016.2624754.

Chin, T.-J. and Suter, D. The maximum consensus problem: Recent algorithmic advances. In *Synthesis Lectures on Computer Vision*. 2017. doi:10.2200/S00757ED1V01Y201702COV011.

Coughlan, J. M. and Yuille, A. L. Manhattan world: Orientation and outlier detection by bayesian inference. *Neural Computation*, 2003. 15(5):1063–1088. doi:10.1162/089976603765202668.

David, P., DeMenthon, D., Duraiswami, R., and Samet, H. Simultaneous pose and correspondence determination using line features. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 2. IEEE, pages II–II, 2003. doi:10.1109/CVPR.2003.1211499.

Ferraz, L., Binefa, X., and Moreno-Noguer, F. Very fast solution to the PnP problem with algebraic outlier rejection. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*. pages 501–508, 2014. doi:10.1109/CVPR.2014.71.

Fischler, M. A. and Bolles, R. C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981. 24(6):381–395. doi:10.1145/358669.358692.

Hartley, R. and Zisserman, A. *Multiple view geometry in computer vision*. Cambridge University Press, 2003. doi:10.1017/CBO9780511811685.

Kumar, R. and Hanson, A. Robust methods for estimating pose and a sensitivity analysis. *CVGIP: Image Understanding*, 1994. 60(3):313–342. URL https://www.sciencedirect.com/science/article/pii/S1049966084710606, doi:https://doi.org/10.1006/ciun.1994.1060.

Lepetit, V., Moreno-Noguer, F., and Fua, P. Epnp: An accurate $o(n)$ solution to the pnp problem. *International Journal of Computer Vision*, 2009. 81(2):155–166. doi:10.1007/s11263-008-0152-6.

Li, S., Xu, C., and Xie, M. A robust $O(n)$ solution to the perspective-n-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012. 34(7):1444–1450. doi:10.1109/TPAMI.2012.41.

Liu, Y., Chen, G., and Knoll, A. Globally optimal camera orientation estimation from line correspondences by BnB algorithm. *IEEE Robotics and Automation Letters*, 2021. 6(1):215–222. doi:10.1109/LRA.2020.3037843.

Liu, Y., Huang, T., and Faugeras, O. Determination of camera location from 2-D to 3-D line and point correspondences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1990. 12(1):28–37. doi:10.1109/34.41381.

Mirzaei, F. M. and Roumeliotis, S. I. Globally optimal pose estimation from line correspondences. In *2011 IEEE International Conference on Robotics and Automation*. pages 5581–5588, 2011. doi:10.1109/ICRA.2011.5980272.

Morrison, D. R., Jacobson, S. H., Sauppe, J. J., and Sewell, E. C. Branch-and-bound algorithms: A survey of recent advances in searching, branching, and pruning. *Discrete Optimization*, 2016. 19:79–102. doi:10.1016/j.disopt.2016.01.005.

Parra Bustos, Á. and Chin, T. J. Guaranteed outlier removal for point cloud registration with correspondences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. 40(512):2868–2882. doi:10.1109/TPAMI.2017.2773482.

Pottmann, H. and Wallner, J. *Computational Line Geometry*. Springer-Verlag, Berlin, 2001. doi:10.1007/978-3-642-04018-4.

Přibyl, B., Zemčík, P., and Čadík, M. Camera pose estimation from lines using Plücker coordinates. *arXiv preprint arXiv:1608.02824*, 2016. doi:10.5244/C.29.45.

Přibyl, B., Zemčík, P., and Čadík, M. Absolute pose estimation from line correspondences using direct linear transformation. *Computer Vision and Image Understanding*, 2017. 161:130–144. doi:10.1016/j.cviu.2017.05.002.

Silva, M., Ferreira, R., and Gaspar, J. Camera calibration using a color-depth camera: Points and lines based dlt including radial distortion. In *Workshop Color-Depth Camera Fusion in Robotics, held with IROS*. 2012. doi:10.1109/IROS.2012.6385509.

Smith, T. and Egeland, O. Dynamical Pose Estimation with Graduated Non-Convexity for Outlier Robustness. *Modeling, Identification and Control*, 2022. 43(2):79–89. doi:10.4173/mic.2022.2.3.

Torr, P. and Zisserman, A. Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 2000. 78(1):138–156. URL https://www.sciencedirect.com/science/article/pii/S1077314299908329, doi:https://doi.org/10.1006/cviu.1999.0832.

Wang, K., Liu, D., Liu, Z., Duan, G., Hu, L., and Tan, J. A fast object registration method for augmented reality assembly with simultaneous determination of multiple 2D-3D correspondences. *Robotics and Computer-Integrated Manufacturing*, 2020. 63:101890. URL https://www.sciencedirect.com/science/article/pii/S0736584519301644, doi:https://doi.org/10.1016/j.rcim.2019.101890.

Wang, P., Xu, G., Cheng, Y., and Yu, Q. Camera pose estimation from lines: a fast, robust and general method. *Machine Vision and Applications*, 2019. 30(4):603–614. doi:DOI: 10.1007/s00138-019-01012-0.

Xu, C., Zhang, L., Cheng, L., and Koch, R. Pose estimation from line correspondences: A complete analysis and a series of solutions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 39(6):1209–1222. doi:10.1109/TPAMI.2016.2582162.

Yang, H., Antonante, P., Tzoumas, V., and Carlone, L. Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection. *IEEE Robotics and Automation Letters*, 2020. 5(2):1127–1134. doi:10.1109/LRA.2020.2965893.

Yang, H., Doran, C., and Slotine, J.-J. Dynamical pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pages 5926–5935, 2021. doi:10.48550/arXiv.2103.06182.

Yu, H., Zhen, W., Yang, W., and Scherer, S. Line-based 2-D–3-D registration and camera localization in structured environments. *IEEE Transactions on Instrumentation and Measurement*, 2020. 69(11):8962–8972. doi:10.1109/TIM.2020.2999137.

Zhang, L., Xu, C., Lee, K.-M., and Koch, R. Robust and efficient pose estimation from line correspondences. In K. M. Lee, Y. Matsushita, J. M. Rehg, and Z. Hu, editors, *Computer Vision – ACCV 2012*. Springer Berlin Heidelberg, Berlin, Heidelberg, pages 217–230, 2013. doi:10.1007/978-3-642-37431-9_17.

Zhang, X., Zhang, Z., Li, Y., Zhu, X., Yu, Q., and Ou, J. Robust camera pose estimation from unknown or known line correspondences. *Applied optics*, 2012. 51(7):936–948. doi:10.1364/AO.51.000936.

Zhang, Y., Li, X., Liu, H., and Shang, Y. Probabilistic approach for maximum likelihood estimation of pose using lines. *IET Computer Vision*, 2016. 10(6):475–482. doi:10.1049/iet-cvi.2015.0099.

Zhou, L., Koppel, D., and Kaess, M. A complete, accurate and efficient solution for the perspective-n-line problem. *IEEE Robotics and Automation Letters*, 2021. 6(2):699–706. doi:10.1109/LRA.2020.3047797.

Zhou, Q., Park, J., and Koltun, V. Fast global registration. In *European Conference on Computer Vision (ECCV)*. Springer, pages 766–782, 2016. doi:10.1007/978-3-319-46475-6_47.