ROBUST ESTIMATION IN ROBOT VISION AND PHOTOGRAMMETRY: A NEW MODEL AND ITS APPLICATIONS

Jiayuan Li, Yongjun Zhang, Qingwu Hu *

School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China
(ljy_whu_2012, zhangyj, huqw)@whu.edu.cn

Commission I,WG I/IV

KEY WORDS: robust estimation, outlier, feature matching, image orientation, point cloud registration, simultaneous localization and mapping (SLAM)

ABSTRACT:

Robust estimation (RE) is a fundamental issue in robot vision and photogrammetry, which is the theoretical basis of geometric model estimation with outliers. However, M-estimations solved by iteratively reweighted least squares (IRLS) are only suitable for cases with low outlier rates ($< 50\%$); random sample consensus (RANSAC) can only obtain approximate solutions. In this paper, we propose an accurate and general RE model that unifies various robust costs into a common objective function by introducing a "robustness-control" parameter. It is a superset of typical least-squares, l_1-l_2, Cauchy, and Geman-McClure estimates. We introduce a parameter-decreasing strategy into the IRLS to optimize our model, called adaptive IRLS. The adaptive IRLS begins with a least-squares estimate for initialization. Then, the "robustness-control" parameter is decreased along with iterations so that the proposed model acts as different robust loss functions and has different degrees of robustness. We also apply the proposed model in several important tasks of robot vision and photogrammetry, such as line fitting, feature matching, image orientation, and point cloud registration (scan matching). Extensive simulated and real experiments show that the proposed model is robust to more than 80% outliers and preserves the advantages of M-estimations (fast and optimal). Our source code will be made publicly available in https://ljy-rs.github.io/web.

1. INTRODUCTION

Robust estimation (RE) is an important technique for model fitting, whose goal is to seek a true geometric model from contaminated observations. Almost all geometric problems related to gross errors (outliers) are inseparable from RE since it is the theoretical basis of outlier detection. RE has been widely applied in robot vision and photogrammetry, such as line fitting, feature matching, image orientation, image odometry, bundle adjustment, point cloud registration (scan matching), and simultaneous localization and mapping (SLAM), etc. Various robust estimation methods have been developed, among which random sample consensus (RANSAC) (Fischler, Bolles, 1981) family and M-estimations solved by iteratively reweighted least squares (IRLS) (Holland, Welsch, 1977) are the most widely used methods.

RANSAC is a hypothesize-and-verify technique that alternates between minimal subset sampling and estimated model verification until the stop criterion is reached. The model with maximum consensus set is accepted as the correct one. RANSAC has many variants, e.g., fixed locally optimized RANSAC (FLO-RANSAC) (Lebeda et al., 2012), universal RANSAC (USAC) (Raguram et al., 2012), and marginalizing sample consensus (MAGSAC++) (Barath et al., 2020). Although many efforts have been made, RANSAC-type methods suffer from several limitations caused by the basic framework of hypothesize-and-verify. First, RANSAC-type methods can not obtain optimal solutions. They use minimal subsets instead of all observations for model estimation, which is sensitive to noise for the lack of redundant observations. Second, the number of sampling trials increases exponentially with outlier rate. As

a result, RANSAC becomes slow at high outlier rates (Chin, Suter, 2017). Third, RANSAC-type methods are difficult to be adapted in adjustment systems such as bundle adjustment and pose graph optimization.

M-estimations originated from statistics have been applied in various fields. Huber, l_1-l_2, Fair, Cauchy, Geman-McClure, Welsch, and Tukey estimators are well-known M-estimators. In addtion, there are many variants, such as GM-estimators, S-estimators, and MM-estimators. More comprehensive surveys can be found in (Huber, 2004). The basic idea of M-like estimators is to use robust functions to penalize outliers and optimizes the cost via the IRLS method. In the optimization, outliers are given small weights (close to 0) while inliers are given large weights (close to 1). Thus, the effect of outliers towards to the total energy is largely discounted. M-estimations overcome some of the shortcomings of RANSAC. First, they can reach globally optimal solutions if good initializations are provided. Second, they are very efficient (1~2 orders of magnitude faster than RANSAC-type methods). More importantly, they can be easily adapted in an adjustment system for joint optimization. Therefore, M-estimations are the preferred methods in robot vision and photogrammetry, especially for gross error detection in adjustment. Nowadays, many observations are obtained automatically by algorithms, which contain a large fraction of outliers, such as the feature matching and point cloud registration tasks. However, M-estimations can not handle cases with more than 50% outliers (Chin, Suter, 2017).

Recently, several estimators try to break through the bottleneck problem of M-estimation. For example, generalized pbM-estimator (GpbM-estimator) (Mittal et al., 2012) proposes a robust errors-in-variables model. Q-norm estimator (Li et al.,
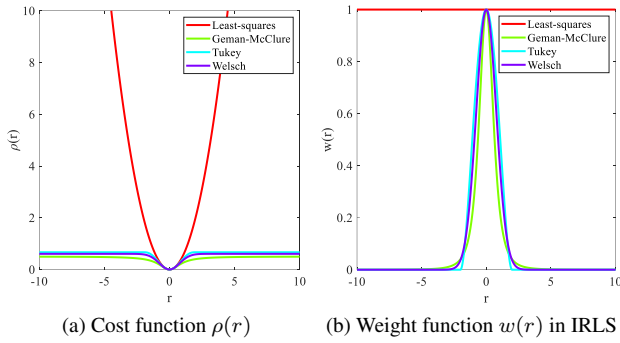
---

* Corresponding author

(a) Cost function $\rho(r)$ (b) Weight function $w(r)$ in IRLS

Figure 1. Comparisons between non-robust least-squares cost and robust M-estimation costs (r is the residual of an observation).

(a) Cost function $\rho(r)$ (b) Weight function $w(r)$ in IRLS

Figure 2. Visualization of the proposed cost function (a) and its weight function (b) with different "robustness-control" parameters α. ($\beta = 1$)

2016) and weighted q-norm estimator (Li et al., 2020b) further improve the robustness by introducing a l_q-norm ($0 < q < 1$) model. These methods are robust to more than 70% outliers. However, they involve non-convex optimization problems, which decrease their efficiency and make them almost impossible to be used in an adjustment system.

In this paper, we aim to develop an estimator with following properties: fast, optimal, robust to high outlier rates, and suitable for adjustment systems. We propose a new robust model that is a superset of typical least-squares, l_1-l_2, Cauchy, and Geman-McClure estimates. This model introduces a "robustness-control" parameter to unify various robust costs into a common framework. We propose a parameter-decreasing strategy in the IRLS (called adaptive IRLS), where the "robustness-control" parameter is decreased along with iterations. In adaptive IRLS optimization, the proposed model acts as different robust loss functions and has different degrees of robustness. We use several important tasks of robot vision and photogrammetry, including line fitting, feature matching, image orientation, and point cloud registration, to show the effectiveness and versatility of the proposed model. Extensive simulated and real experiments show its great potentials, i.e, it is robust to more than 80% outliers and inherits all advantages of M-estimations. Our contributions are as follows:

- We propose a robust cost function that unifies a set of loss functions with different degrees of robustness. Our cost function can be generalized to different dimensions.

- We propose an adaptive IRLS for optimization, so that the model acts as different robust loss functions.

- We provide several applications in photogrammetry for readers to understand and use the proposed model.

2. OUR ROBUST MODEL

2.1 Motivation

Least-squares cost is widely used in robot vision and photogrammetry because of its optimality for Gaussian noise. However, it is very sensitive to outliers. A single outlier can make the solution highly suboptimal, since the Gaussian noise assumption is violated by outliers. As shown in Figure 1, least-squares cost (the red curve) increases quadratically and gives same weights to all observations. It is not a robust cost. In
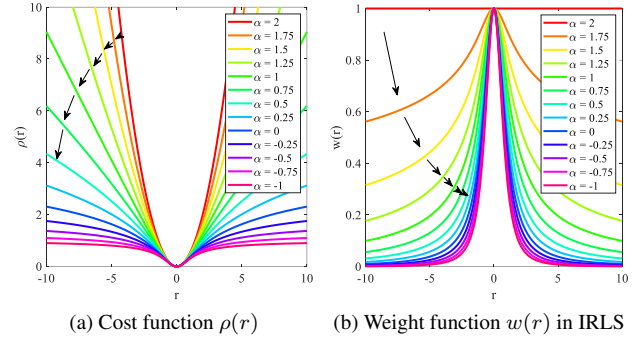
contrast, typical M-estimation costs (such as Geman-McClure, Tukey, and Welsch) are bounded functions. They give small weights to observations with large residuals while giving large weights to others in the IRLS, so that the observations with large residuals contribute small to the total energy. The problem of these costs is the sensitivity to high outlier rates. If outlier rate is higher than 50%, initial estimated model is largely biased from its ground truth. In this case, inliers may have large residuals and are given very small weights (Observations with residuals lager than 4 are given ≈ 0 weights in this figure). This is so strict that many inliers may not be participated in the optimization at the beginning.

So, if we can bridge the gap between least-squares cost and M-estimation costs, the above-mentioned problem will be largely alleviated. At the beginning, the cost should let all observations participate in optimization like the least-squares cost. Then, the cost gradually increases the robustness and becomes a robust function like the M-estimation costs (The change process is shown by arrows in Figure 2.). Although initial model may be not accurate, true inliers with large residuals still contribute to the total energy. Actually, in each change of the cost function, a small portion of observations with the largest residuals are excluded from optimization (assigned ≈ 0 weights). Generally, these observations are outliers. Therefore, as the cost function changes, many outliers are discarded and the true outlier rates decrease. When the cost becomes a M-estimation like cost, the true outlier rates can be smaller than 50%.

2.2 Cost Function

The basic cost function of our proposed robust estimation model is,

$$f(r, \alpha, \beta) = \frac{\beta^2}{\alpha} \left(\left(1 + \left(\frac{r}{\beta} \right)^2 \right)^{\frac{\alpha}{2}} - 1 \right) \qquad (1)$$

where r is the residual of an observation; α reflects the robustness of the function, which is called "robustness-control" parameter; and $\beta > 0$ is a scale parameter. Our cost yields the classic least-squares loss when $\alpha = 2$,

$$f(r, 2) = \frac{1}{2} r^2 \qquad (2)$$

Least-squares cost is very sensitive to gross errors. Even one outlier can lead to an arbitrary bad solution. When $\alpha = 1$, the

---

Algorithm 1: Robust estimation based on our robust model

Input: observations $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$
Output: parameters $\boldsymbol{\delta}$ of model $\mathcal{F}$

1 Initialization $\boldsymbol{w}^{(0)} = \mathbf{1}$, $\boldsymbol{\delta} = \boldsymbol{\delta}^{(0)}$, $\alpha^{(0)} = 2$, $\beta = 10$;
2 **while** $\| \Delta \boldsymbol{\delta} \| < \varepsilon$ **do**
3   **Variable update:** estimate $\boldsymbol{\delta}^{(t)}$ via WLS (Eq. (9));
4   **Weight update:** compute weights $\boldsymbol{w}^{(t)}$ via Eq. (7);
5   α **update:** decrease α by a step, $\alpha^{(t)} = \alpha^{(t-1)} - \tau$;
6     $\Delta \boldsymbol{\delta} = \boldsymbol{\delta}^{(t)} - \boldsymbol{\delta}^{(t-1)}$, t++;
7 Compute final $\boldsymbol{\delta}$ via WLS with the newest weights;
8 Output $\boldsymbol{\delta}$ and select inliers based on residuals.

---

proposed cost becomes the l_1-l_2 estimate,

$$f(r, 1, \beta) = \beta^2 \left(\sqrt{1 + \left(\frac{r}{\beta} \right)^2} - 1 \right) \tag{3}$$

In our cost, $\alpha = 0$ is a singularity, i.e., $f(r, 0, \beta)$ is undefined. Take the limit as α approaches to 0, function (1) turns into the Cauchy estimate,

$$\lim_{\alpha \to 0} f(r, \alpha, \beta) = \frac{\beta^2}{2} \ln \left(1 + \left(\frac{r}{\beta} \right)^2 \right) \tag{4}$$

If we set $\alpha = -2$, our cost is equivalent to the Geman-McClure estimate,

$$f(r, -2, \beta) = \frac{\beta^2 r^2}{2 \left(\beta^2 + r^2 \right)} \tag{5}$$

As known, the l_1-l_2, Cauchy, and Geman-McClure estimates are robust cost functions. Our proposed cost is a superset of these functions and the shape parameter α controls its robustness. Considering the singularity, our final general cost is,

$$\rho(r, \alpha, \beta) = \begin{cases} \frac{\beta^2}{2} \ln \left(1 + \left(\frac{r}{\beta} \right)^2 \right) & \alpha = 0 \\ \frac{\beta^2}{\alpha} \left(\left(1 + \left(\frac{r}{\beta} \right)^2 \right)^{\frac{\alpha}{2}} - 1 \right) & \alpha \neq 0 \end{cases} \tag{6}$$

Our cost can also be optimized by the IRLS like M-estimations. In the IRLS, Eq. (6) is reformulated as a weighted least-squares (WLS) problem, i.e., $\sum w(r) r^2$. $w(r) = \frac{\partial \rho}{\partial r} / r$ is a weight function, where $\frac{\partial \rho}{\partial r}$ is the derivative of ρ with respect to r. Hence, our weight function used in the IRLS is,

$$w(r, \alpha, \beta) = \begin{cases} \frac{\beta^2}{\beta^2 + r^2} & \alpha = 0 \\ \left(1 + \left(\frac{r}{\beta} \right)^2 \right)^{\frac{\alpha}{2} - 1} & \alpha \neq 0 \end{cases} \tag{7}$$

2.3 Main Algorithm

Suppose $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ is a set of outlier contaminated observations. The geometric relationship between $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ can be modeled by a function $\mathcal{F}$, i.e., $\boldsymbol{y}_i = \mathcal{F}(\boldsymbol{x}_i, \boldsymbol{\delta})$ (noise and outliers are not considered), where $\boldsymbol{\delta}$ is the parameters of $\mathcal{F}$. The goal is to estimate the parameters $\boldsymbol{\delta}$. This problem can be formulated as our robust model,

$$\min_{\boldsymbol{\delta}} \sum_{i=1}^n \rho \left(r_i, \alpha, \beta \right) \tag{8}$$

where $r_i = r(\boldsymbol{x}_i, \boldsymbol{y}_i, \boldsymbol{\delta}) = \| \boldsymbol{y}_i - \mathcal{F}(\boldsymbol{x}_i, \boldsymbol{\delta}) \|$. We then rewrite this problem as an IRLS problem,

$$\min_{\boldsymbol{\delta}} \sum_{i=1}^n w_i r_i^2 \tag{9}$$

where $w_i = w(r_i, \alpha, \beta)$. Problem (9) can be optimized by adaptive IRLS, whose details are summarized in **Algorithm** 1. At each inner iteration t, three main steps are performed:

1. **Variable update**: estimate $\boldsymbol{\delta}^{(t)}$ with fixed weights $w_i^{(t-1)}$ in Eq. (9),

$$\boldsymbol{\delta}^{(t)} = \min_{\boldsymbol{\delta}} \sum_{i=1}^n w_i^{(t-1)} r_i^2 \tag{10}$$

This is a simple WLS problem that can be easily solved.

2. **Weight update**: compute weights $\boldsymbol{w}^{(t)} = \{w_i^{(t)}\}_1^n$ based on Eq. (7) with fixed $\boldsymbol{\delta}^{(t)}$,

$$w_i^{(t)} = w \left(r_i^{(t)}, \alpha^{(t-1)}, \beta \right) \tag{11}$$

3. α **update**: decrease the "robustness-control" parameter α by a step-size τ to change the weight function, i.e., $\alpha^{(t)} = \alpha^{(t-1)} - \tau$.

Compared with traditional IRLS, our proposed adaptive IRLS has one more step of α update. Along with iterations, this step changes the cost function and increases robustness. As can be seen, our adaptive IRLS has the same time complexity with traditional IRLS. If the WLS optimization in **Variable update** costs $O(n)$ time. Then, the complexity of the proposed robust estimation method is $O(n)$.

3. APPLICATIONS

3.1 Line Fitting

The goal of this problem is to fit a 2D straight line from outlier-contaminated points $\{(x_i, y_i)\}_1^n$. The 2D line model $\mathcal{F}$ is,

$$y_i = \mathcal{F}(x_i, \boldsymbol{\delta}) = ax_i + b \tag{12}$$

where $\boldsymbol{\delta} = (a, b)$. Then, the residual $r_i = (y_i - ax_i - b)$ and the weight function of our model for line fitting is,

$$w(x_i, y_i, \alpha, \beta) = \begin{cases} \frac{\beta^2}{\beta^2 + (y_i - ax_i - b)^2} & \alpha = 0 \\ \left(1 + \left(\frac{y_i - ax_i - b}{\beta} \right)^2 \right)^{\frac{\alpha}{2} - 1} & \alpha \neq 0 \end{cases} \tag{13}$$

3.2 Image Feature Matching

Invoking the image feature matching problem, our goal is to eliminate mismatches from initial correspondences $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ that are extracted from an image pair (I_1, I_2), where $\boldsymbol{x}_i$, $\boldsymbol{y}_i \in \mathbb{R}^2$. We choose affine transformation to model the relationship between (I_1, I_2), i.e., estimating an affine model $\mathcal{F} : \mathbb{R}^2 \to \mathbb{R}^2$ that aligns I_1 and I_2,

$$\boldsymbol{y}_i = \mathcal{F}(\boldsymbol{x}_i, \boldsymbol{\delta}) = \mathbf{A}\boldsymbol{x}_i + \boldsymbol{t} \tag{14}$$

where $\boldsymbol{\delta} = (\mathbf{A}, \boldsymbol{t})$, $\mathbf{A}$ is a 2×2 affine matrix, and $\mathbf{t}$ is a 2×1 translation vector. In this problem, $r_i = \| \boldsymbol{y_i} - (\mathbf{A}\boldsymbol{x_i} + \boldsymbol{t}) \|$ and the weight function used in **Algorithm 1** becomes,

$$
w(\boldsymbol{x_i}, \boldsymbol{y_i}, \alpha, \beta) = \begin{cases} \frac{\beta^2}{\beta^2 + \|\boldsymbol{y_i} - (\mathbf{A}\boldsymbol{x_i} + \boldsymbol{t})\|^2} & \alpha = 0 \\ \left(1 + \left(\frac{\|\boldsymbol{y_i} - (\mathbf{A}\boldsymbol{x_i} + \boldsymbol{t})\|}{\beta}\right)^2\right)^{\frac{\alpha}{2} - 1} & \alpha \neq 0 \end{cases}
$$
(15)

3.3 Image Orientation

The goal of exterior orientation is to recover the image pose from n contaminated 3D-2D correspondences $\{(\boldsymbol{Q_i}, \boldsymbol{p_i})\}_1^n$, where $\{\boldsymbol{Q_i}\}_1^n$ are 3D object points and $\{\boldsymbol{p_i}\}_1^n$ are their 2D counterparts on the image. Assume camera internal parameters are provided by matrix $\mathbf{K}$, then, $\boldsymbol{Q_i}$ can be projected onto $\boldsymbol{p_i}$ by camera projection model,

$$
\boldsymbol{p_i} = \mathcal{F}(\boldsymbol{Q_i}, \boldsymbol{\delta}) = \frac{\mathbf{P}_{1:2}\begin{bmatrix} \boldsymbol{Q_i} & 1 \end{bmatrix}^T}{\mathbf{P}_3\begin{bmatrix} \boldsymbol{Q_i} & 1 \end{bmatrix}^T}
$$
(16)

where $\mathbf{P} = \mathbf{K}[\mathbf{R}, \boldsymbol{T}]$ is the camera projection matrix ($\mathbf{P}_j (j = 1, 2, 3)$ is the j-th row of $\mathbf{P}$), $\mathbf{R}$ is a 3×3 rotation matrix, $\boldsymbol{T}$ is a 3×1 translation, and $\boldsymbol{\delta} = (\mathbf{R}, \boldsymbol{T})$. The weight function for exterior orientation is,

$$
w(\boldsymbol{Q_i}, \boldsymbol{p_i}, \alpha, \beta) = \begin{cases} \frac{\beta^2}{\beta^2 + \|\boldsymbol{p_i} - F(\boldsymbol{Q_i}, \delta)\|^2} & \alpha = 0 \\ \left(1 + \left(\frac{\|\boldsymbol{p_i} - F(\boldsymbol{Q_i}, \delta)\|}{\beta}\right)^2\right)^{\frac{\alpha}{2} - 1} & \alpha \neq 0 \end{cases}
$$
(17)

3.4 Point Cloud Registration

In the point cloud registration problem, we aim to estimate a rigid transformation that merges a point cloud pair $(\mathcal{H}_1, \mathcal{H}_2)$ in the same coordinate system. Given a set of 3D initial correspondences $\{(\boldsymbol{X_i}, \boldsymbol{Y_i})\}_1^n$ that is extracted from $(\mathcal{H}_1, \mathcal{H}_2)$, the model $\mathcal{F}$ for registration problem is,

$$
\boldsymbol{Y_i} = \mathcal{F}(\boldsymbol{X_i}, \boldsymbol{\delta}) = \mathbf{R}\boldsymbol{X_i} + \boldsymbol{T}
$$
(18)

where $\boldsymbol{\delta} = (\mathbf{R}, \boldsymbol{T})$, $\mathbf{R}$ is a 3D rotation matrix, and $\boldsymbol{T}$ is a 3×1 translation vector. Then, the weight function for point set registration is,

$$
w(\boldsymbol{X_i}, \boldsymbol{Y_i}, \alpha, \beta) = \begin{cases} \frac{\beta^2}{\beta^2 + \|\boldsymbol{Y_i} - (\mathbf{R}\boldsymbol{X_i} + \boldsymbol{T})\|^2} & \alpha = 0 \\ \left(1 + \left(\frac{\|\boldsymbol{Y_i} - (\mathbf{R}\boldsymbol{X_i} + \boldsymbol{T})\|}{\beta}\right)^2\right)^{\frac{\alpha}{2} - 1} & \alpha \neq 0 \end{cases}
$$
(19)

Note that if we add a scale factor s in Eq. (18), the problem becomes an absolute orientation problem.

4. EXPERIMENTS AND EVALUATIONS

This section evaluates the proposed robust model on both simulated and real experiments. We compare our method with Cauchy M-estimator, Welsch M-estimator, RANSAC, FLO-RANSAC, Q-norm estimator, and MAGSAC++ on line fitting, feature matching, image orientation, and point set registration tasks. We use root-mean-square error (RMSE), success rate, and running time for quantitative evaluations. Table 1 summarizes parameter and implementation information of each method. All experiments are performed on a laptop with CPU Core i7-8550U @ 1.8GHz, and 8 GB of RAM.
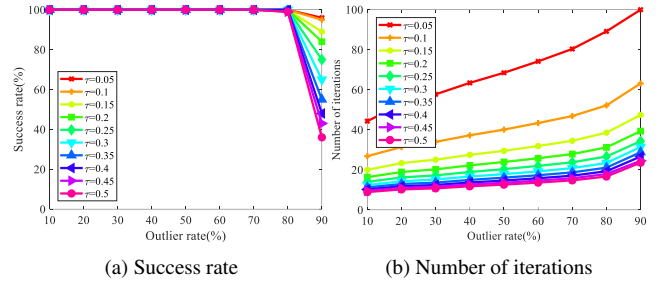

(a) Success rate   (b) Number of iterations

Figure 3. Feature matching results of the proposed model with different values of τ.

Method	Parameter	Implementation
Cauchy	Tuning constant: 2.385; MI: 100.	MATLAB http://rosa.unipr.it/fsda.html
Welsch	Tuning constant: 2.985; MI: 100.	MATLAB http://rosa.unipr.it/fsda.html
RANSAC	Subset size: 3; confidence: 0.99; MI: 10^4.	MATLAB https://www.peterkovesi.com
FLO-RANSAC	Subset size: 3; confidence: 0.99; MI: 10^4; local optimized (LO) size: 21.	MATLAB https://zhipengcai.github.io/
Q-norm	q: 0.5; step size: 1.65; initial penalty: 10^{-4}; MI: 100.	MATLAB https://github.com/LJY-RS
MAGSAC++	Subset size: 3; confidence: 0.99; MI: 10^4; maximum threshold: 10σ.	C++ https://github.com/danini/magsac
Our method	τ: 0.2; β: 10; MI: 100.	MATLAB https://ljy-rs.github.io/web

Table 1. Detailed settings of the compared methods (MI represents maximum number of iterations)

4.1 Parameter τ Study

Parameter τ is important for the proposed model. We conduct a simulated feature matching experiment to study the sensitivity to τ. The simulation details are as follows: Given the number of observations n and an outlier rate η, the size of inliers n_1 is computed by $n_1 = n * (1 - \eta)$. We first randomly generate n_1 2D feature points $\mathcal{X}_1 = \{\boldsymbol{x_i}\}_1^{n_1}$ by using a normal distribution $\mathcal{N}(0, 500^2)$. Then, we obtain their correspondences $\mathcal{Y}_1 = \{\boldsymbol{y_i}\}_1^{n_1}$ via transformation $\boldsymbol{y_i^t} = \mathbf{A}^t \boldsymbol{x_i} + \boldsymbol{t}^t$, where $\mathbf{A}^t = \begin{bmatrix} s_x \cos\theta & s_x \sin\theta \\ -s_y \sin\theta & s_y \cos\theta \end{bmatrix}$ and $\boldsymbol{t}^t = \begin{bmatrix} t_x \\ t_y \end{bmatrix}$. The rotation angle $\theta \in (-\frac{\pi}{2}, \frac{\pi}{2})$, anisotropic scales $s_x, s_y \in (0.5, 1.5)$, and translations $t_x, t_y \in (-500, 500)$ are all randomly generated. We also add Gaussian noise $\mathcal{N}(0, 2^2)$ to $\mathcal{Y}_1$ to make the simulation realistic, obtaining an inlier set $\mathcal{H}_1 = (\mathcal{X}_1, \mathcal{Y}_1)$. Finally, two sets of points with the same size $n_2 = n - n_1$ are generated based on a normal distribution $\mathcal{N}(0, 500^2)$ to obtain the outlier set $\mathcal{H}_2 = (\mathcal{X}_2, \mathcal{Y}_2)$. The inlier set and outlier set are merged to obtain the outlier-contaminated feature correspondences $\{(\boldsymbol{x_i}, \boldsymbol{y_i})\}_{i=1}^n$. In our experiment, we set $n_1 = 1000$ and $r = \{10\%, 20\%, 30\%, 40\%, 50\%, 60\%, 70\%, 80\%, 90\%\}$. For each (n_1, η) configuration, we perform 100 independent tests and report the average results. In a test, if the RMSE of a method is smaller than triple of the noise level ($\sigma = 2$ in this experiment), this estimation is successful. Thus, the definition of success rate is the ratio of successful estimation times in 100 independent tests.

Figure 3 shows the results of the proposed method with different τ. If the outlier rate is lower than 80%, our model is not sensitive to τ. It achieves impressive performance when τ is within [0.05 0.5], i.e., the success rate are always close to 100% (Figure 3(a)). However, larger τ results in worse performance
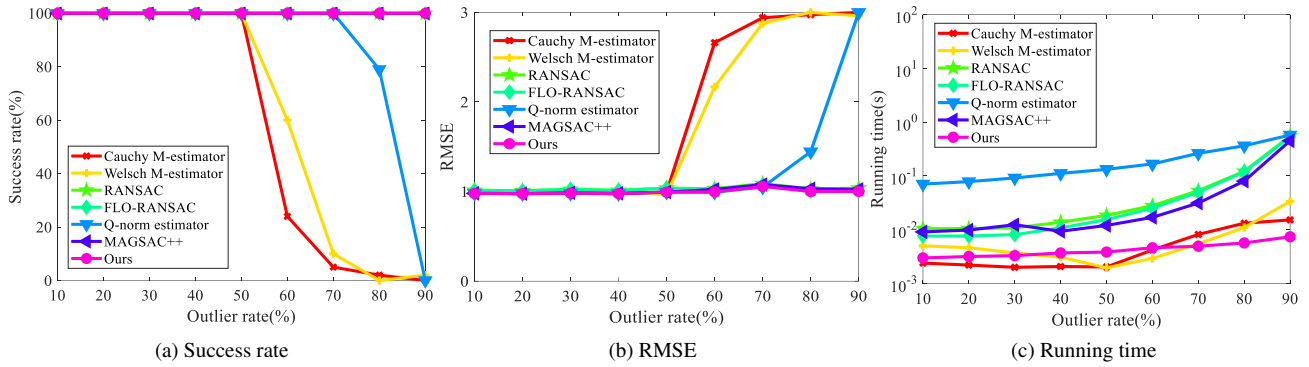
(a) Success rate   (b) RMSE   (c) Running time

Figure 4. Comparison of line fitting results on the simulated data.

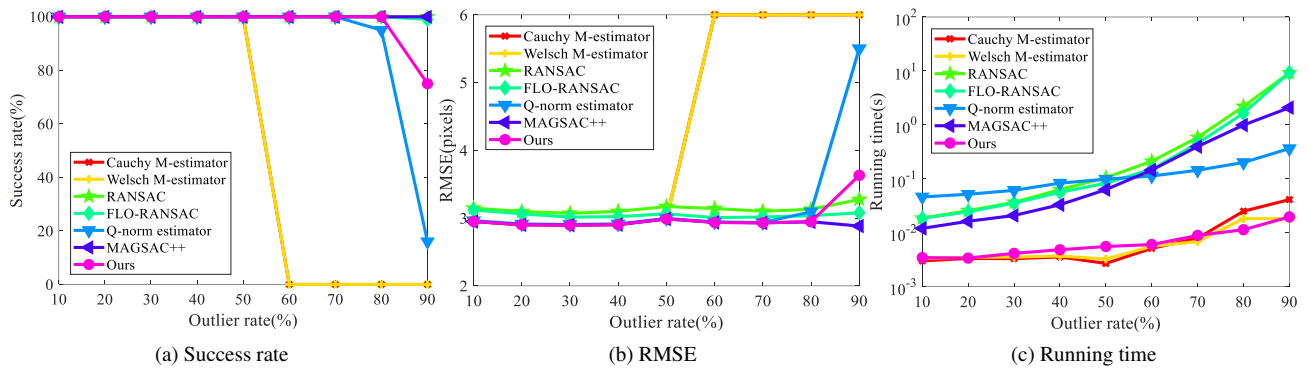(a) Success rate   (b) RMSE   (c) Running time

Figure 5. Comparison of feature matching results on the simulated data.

when $\tau > 80\%$. We also report the required number of iterations. As can be seen in Figure 3(b), smaller τ requires more iterations for convergence and costs more time. Thus, we make a trade-off and fix $\tau = 0.2$ for all following experiments.

4.2 Line Fitting Experiment

The simulation process is similar to the one in Section 4.1. The differences are that the observations are 2D points $\{(x_i, y_i)\}_1^n$ and the transformation is $y_i = a^t x_i + b^t$, where the slope angle is generated within $\arctan(a^t) \in (-\frac{\pi}{2}, \frac{\pi}{2})$ and the offset is within $b^t \in (-100, 100)$. The standard deviation of Gaussian noise is $\sigma = 1$. The results are summarized in Figure 4.

As shown, M-estimators (Cauchy and Welsch) work well at low outlier rates ($r \leq 50\%$). However, their performances dramatically drop if the outlier rate r is larger than 50%. Thus, their success rate curves show a "cliff-like drop". Q-norm estimator performs much better than M-estimators. It is robust against 70% outliers. Compared with abovementioned estimators, RANSAC-type methods (RANSAC, FLO-RANSAC, and MAGSAC++) and our method can deal with 90% outliers. Their success rates are always 100%. RANSAC and FLO-RANSAC have slightly lower estimation accuracies (larger RMSE) than MAGSAC++ and our method, since they only estimate an approximate solution. Although MAGSAC++ is also a RANSAC variant, it uses multiple strategies such as a M-estimation stage to refine the model obtained by the RANSAC-like stage. Our method is much faster than MAGSAC++, e.g., 60+ times faster at an outlier rate of 90%, even though MAGSAC++ is implemented in C++.

4.3 Feature Matching Experiment

Simulation: The simulation process is the same as the one in Section 4.1. Figure 5 reports the results. Again, M-estimators breakdown at an outlier rate of 50%. The reason is that M-estimators use median absolute deviations (MAD) for residual scale estimation. The median operator only considers half of the observations and gets inaccurate residual scale when the outlier rate is higher than 50%. In this task, our method is still very robust although the performance slightly decreases when outlier rate reaches 90%. Compared with RANSAC-type methods, our main advantages are two-fold: First, our model is much more efficient. For example, our method is about 90 times faster than MAGSAC++ at an outlier rate of 80%. Second, our method is suitable for residual adjustment, since adaptive IRLS is almost the same as traditional IRLS. As known, M-estimators solved by IRLS are used in bundle adjustment for outlier removal. Our method can also be extended for multi-image feature matching problem, which will be our future work.

Real data: Six real multimodal image pairs are used for evaluation. We apply radiation-variation insensitive feature transform (RIFT) (Li et al., 2020a) algorithm to extract initial feature correspondences. The lowest outlier rate η of these image pairs is 69.8% and the highest one is 96.5% due to severe nonlinear radiation differences. Figure 6 shows our matching results. The RMSEs are smaller than 2 pixels. The average running time is only 4 ms while MAGSAC++ costs 176 ms.

4.4 Image Orientation Experiment

Simulation: Given camera internal parameter matrix $\mathbf{K}$, we first randomly generate n 3D image space points $\{\boldsymbol{Q}_i^c\}_1^n$ inside

n =1420, η =96.5%, Time=0.005s, RMSE=1.78

(a) visible-visible

n =1588, η =77.2%, Time=0.006s, RMSE=1.59

(b) infrared-visible

n =1369, η =69.8%, Time=0.004s, RMSE=1.95

(c) SAR-visible

n =1129, η =92.0%, Time=0.003s, RMSE=1.93

(d) depth-visible

n =873, η =80.0%, Time=0.002s, RMSE=1.86

(e) map-visible

n =847, η =93.1%, Time=0.004s, RMSE=1.81
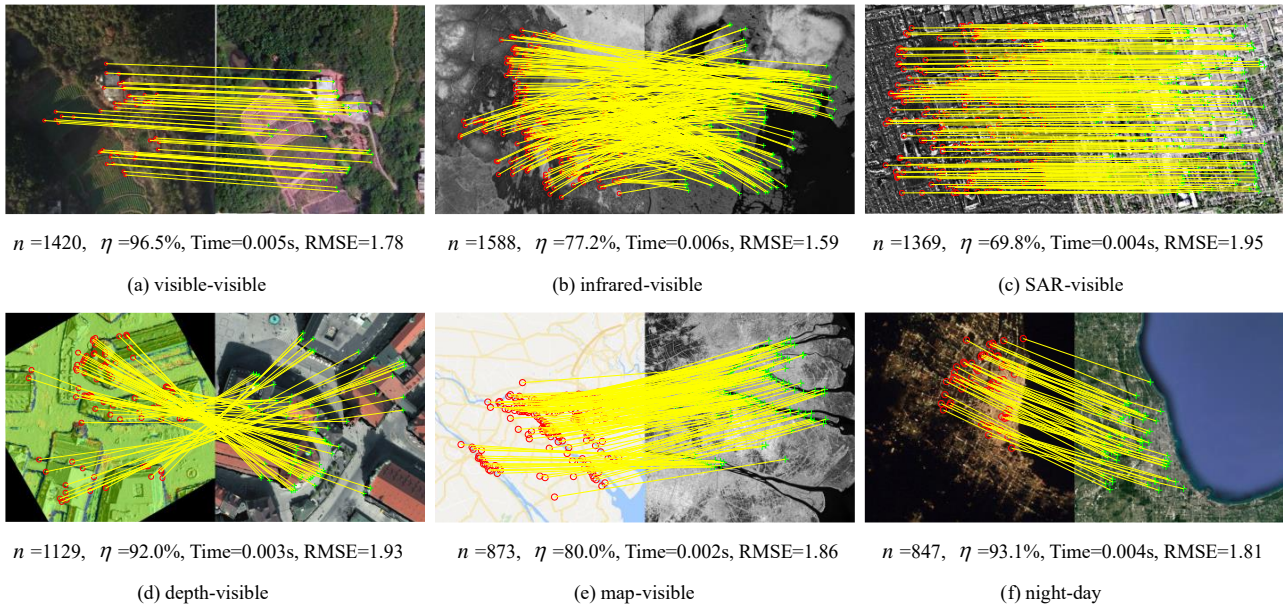
(f) night-day

Figure 6. Our matching results on six different types of multimodal image pairs. n is the size of initial correspondence set, η is the outlier rate of initial correspondences. Each line represents a feature correspondence.
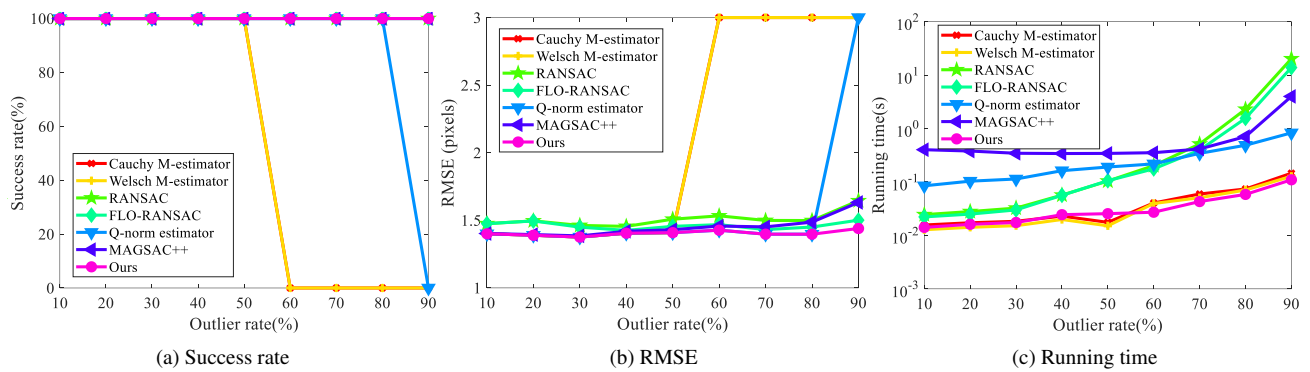

(a) Success rate

(b) RMSE

(c) Running time

Figure 7. Comparison of image orientation results on the simulated data.

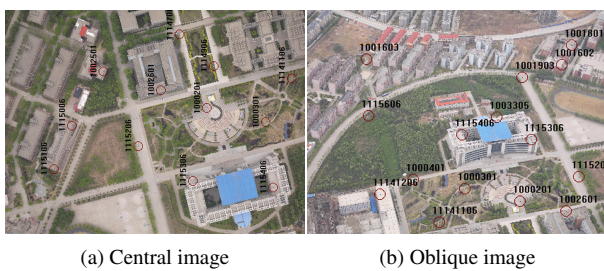(a) Central image   (b) Oblique image

Figure 8. Aerial images for real image orientation experiment. Red circles are correct correspondences, whose 3D control points are measured by GPS-RTK technique and 2D image points are manually picked.

a box $[-8, 8] \times [-8, 8] \times [8, 16]$. Their corresponding object space points are obtained by $\boldsymbol{Q}_i = (\mathbf{R}^t)^{-1}(\boldsymbol{Q}_i^c - \boldsymbol{T}^t)$, where the rotation matrix $\mathbf{R}^t$ and translation vector $\boldsymbol{T}^t$ are randomly generated. Then, the object space points are projected onto the image via $d_i \begin{bmatrix} \boldsymbol{p}_i^t & 1 \end{bmatrix}^T = \mathbf{K}\boldsymbol{Q}_i^c$, where d_i is the depth and $\boldsymbol{p}_i^t$ is the image correspondence of $\boldsymbol{Q}_i$. The Gaussian noise with 1-pixel deviation is added to $\{\boldsymbol{p}_i^t\}_1^n$. Finally, n_2 image

points are selected to add errors (outliers), obtaining observed image points $\{\boldsymbol{p}_i\}_1^n$. $\{(\boldsymbol{Q}_i, \boldsymbol{p}_i)\}_1^n$ are the contaminated 3D-2D correspondences. We use Gauss-Newton method in the WLS optimization to solve the nonlinear image orientation problem. It requires initialization for rotation and translation parameters. We obtain initial rotation angles by adding a random disturbance between [-10°, 10°] to the ground truth and generate initial translations between $[70\% \times \boldsymbol{T}, 130\% \times \boldsymbol{T}]$. For fairness, this initialization is adapted for all compared methods. The image orientation results are displayed in Figure 7. With appropriate initializations, our method achieves 100% success rate at an outlier rate of 90% and performs better than RANSAC-type methods in terms of RMSE.

Real data: We use two aerial images (see Figure 8) for evaluation, which are obtained over the Pingdingshan City, Henan, China. The left image is captured by the central camera of SWDC system with a focal length of 12102.1 pixels and a size of 5406×7160 pixels and the right one is acquired by an oblique camera with a focal length of 14671.5 pixels and a size of 7160×5406 pixels. We measured 24 (contains 12 inliers) and 75 (contains 15 inliers) 3D-2D correspondences for Figure 8(a) and Figure 8(b), respectively. The inliers are displayed as red circles in the figure. The initializations (three rotation angle

Method	Image ID	φ (°)	ω (°)	κ (°)	X_s (m)	Y_s (m)	Z_s (m)	RMSE (pixels)	TIME (s)
Initial guesses	1	0	0	0	0	0	500	–	–
	2	0	0	-30	0	0	500	–	–
PATB	1	-3.21	0.54	-19.80	16.74	32.82	650.60	1.05	–
	2	17.99	-44.69	-117.58	657.72	162.19	646.99	1.19	–
Our method	1	-3.22	0.58	-19.78	16.85	32.77	650.59	1.00	0.02
	2	17.96	-44.67	-117.56	657.84	162.09	647.89	1.05	0.02

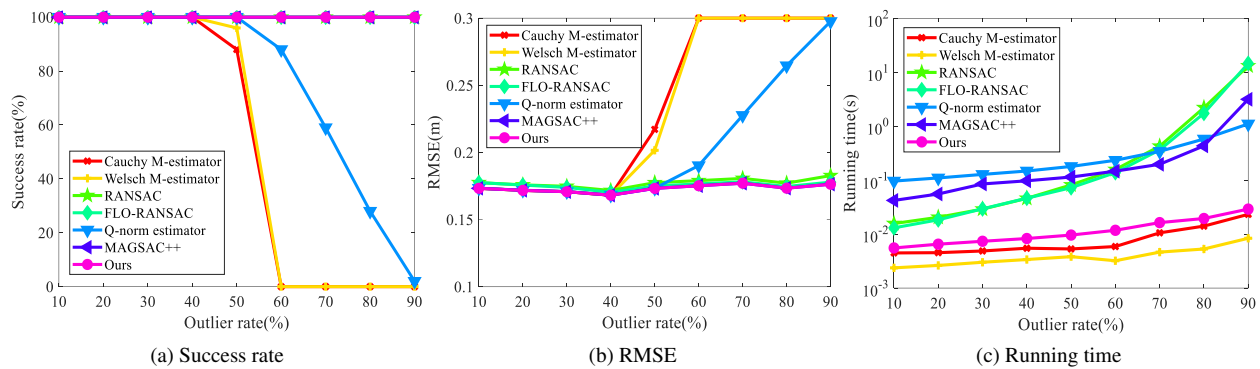Table 2. The image orientation results on two large-scale real images.

(a) Success rate   (b) RMSE   (c) Running time

Figure 9. Comparison of point set registration results on the simulated data.

n =9022, η =99.2%, RMSE=0.14 n =4985, η =97.2%, RMSE=0.15 n =15699, η =99.5%, RMSE=0.16 n =5201, η =96.4%, RMSE=0.11

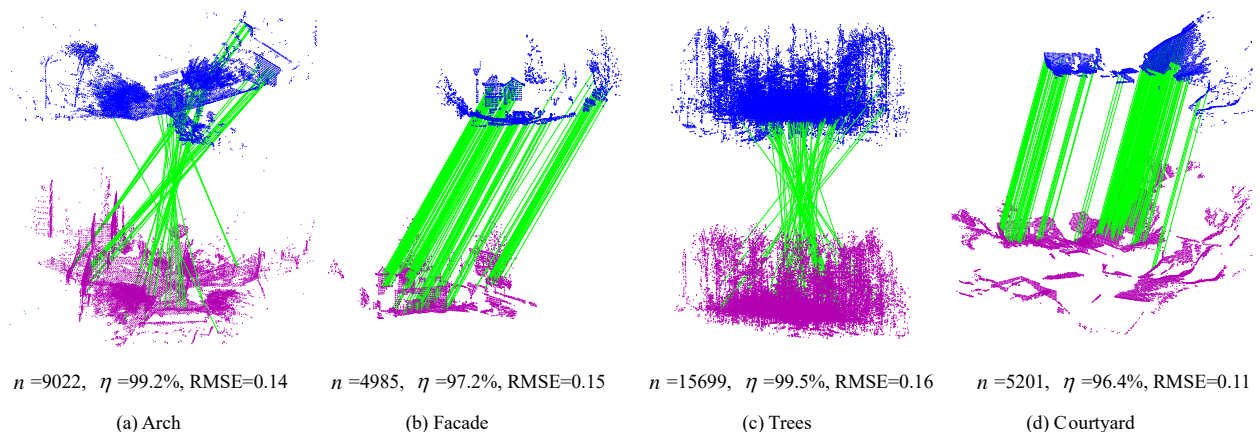(a) Arch   (b) Facade   (c) Trees   (d) Courtyard

Figure 10. Our 3D matching results on four LIDAR scan pairs. n is the number of initial 3D correspondences and η is the outlier rate of initial correspondences. Each line represents a 3D correspondence.

parameters $[\varphi, \omega, \kappa]$ and three location parameters $[X_s, Y_s, Z_s]$) and results are reported in Table 2. As shown, our results are very close to the ones obtained by the commercial software PATB with only inlier correspondences. Although the outlier rate of the second image is 80%, the proposed method still achieves an accuracy of 1.05 pixels.

4.5 Point Cloud Registration Experiment

Simulation: The simulation process is similar to the one in Section 4.1. The 3D points are generated by a normal distribution $\mathcal{N}(0, 100^2)$. The rotation angles are randomly generated within $\left(-\frac{\pi}{2}, -\frac{\pi}{2}\right)$ and translations are generated within $(-100, 100)$. The noise level is 0.1m. Figure 9 provides the comparison results. Again, our method and MAGSAC++ achieve the highest accuracies, including success rate and RMSE. Our method is much faster than MAGSAC++. We note that the efficiency of MAGSAC++ becomes slow when the estimated model is complicated, such as in this task and image orientation task.

Real data: Four LIDAR scan pairs are collected from the

ETH dataset[1] (ground truth transformation of each scan pair is provided) for evaluation. The resolution of these point clouds is downsampled to 0.1m. We use ISS (Zhong, 2009) feature detector and FPFH (Rusu et al., 2009) descriptor to establish initial 3D correspondences. 3D feature matching is much more difficult than its 2D counterpart. Therefore, the outlier rates of the initial 3D correspondences are very high ($> 96\%$ in these scan pairs). We first use an edge voting strategy (Li et al., 2020b) to filter some outliers before applying the proposed method for rigid model estimation. The 3D matching results are shown in Figure 10. The registration accuracy of our method is better than 0.2m (twice of the resolution).

We also show the potential of the proposed method in 3D laser SLAM task. We use three sequences of KITTI dataset[2] (with ground truth) for evaluation. Sequence 5 consists of 2761 laser

---

[1] https://prs.igp.ethz.ch/research/completed_projects/automatic_registration_of_point_clouds.html

[2] http://www.cvlibs.net/datasets/kitti/eval_odometry.php

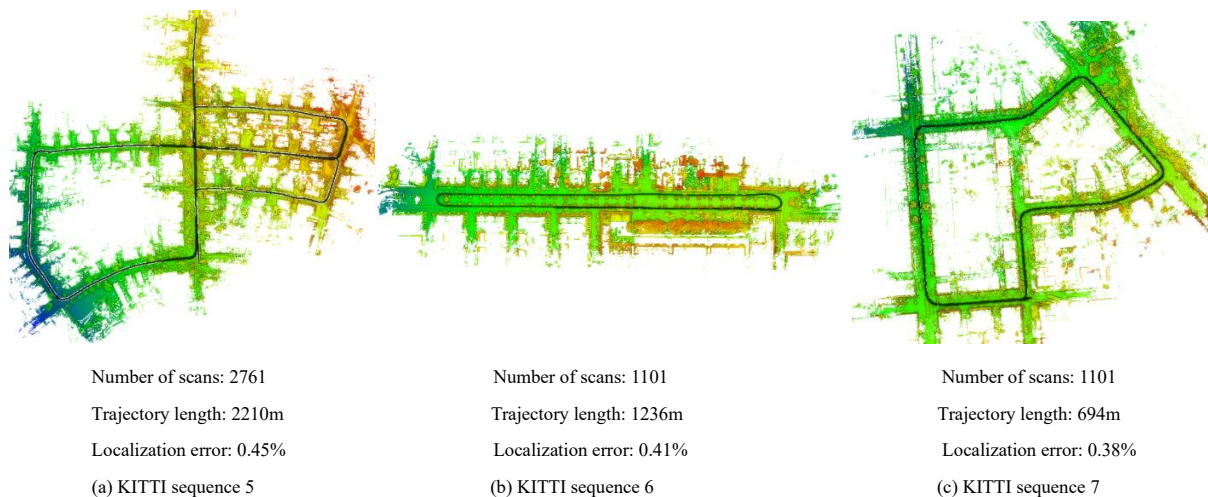| Number of scans: 2761 | Number of scans: 1101 | Number of scans: 1101 |
Trajectory length: 2210m	Trajectory length: 1236m	Trajectory length: 694m
Localization error: 0.45%	Localization error: 0.41%	Localization error: 0.38%
(a) KITTI sequence 5	(b) KITTI sequence 6	(c) KITTI sequence 7

Figure 11. Our laser odometry results on the KITTI dataset. The color of point clouds is rendered by height. The black line inside each subfigure is the estimated trajectory of the vehicle.

scans; sequence 6 and 7 contain 1101 laser scans. For each sequence, we regard every 20 consecutive scans as a group. First, each group is registered by our method to obtain a submap. Then, we register consecutive submaps based on our method to get the whole scene map. We use the translational error (localization error) of KITTI as the evaluation metric. The reconstructed 3D scene maps and the estimated trajectories of the vehicle are displayed in Figure 11. The localization accuracy of our laser odometry is better than 0.5% on these three laser sequences.

4.6 Limitations

Our method relies on an assumption that outliers are randomly distributed or approximately uniform distributed. If this assumption is violated, our method may fail. Thus, our method is not suitable for cases with multiple geometric models.

5. CONCLUSIONS

We proposes a general objective function by introducing a "robustness-control" parameter. It is a superset of typical least-squares, l_1-l_2, Cauchy, and Geman-McClure costs. We present an adaptive IRLS for optimization. It uses different weight functions in iterations and has different degrees of robustness. We also provide the applications to several important tasks of robot vision and photogrammetry. Extensive simulated and real experiments demonstrate that our model is robust to more than 80% outliers and is much faster than RANSAC-type methods. We will integrate our robust model in visual and laser SLAM systems in the future.

ACKNOWLEDGEMENTS

REFERENCES

Barath, D., Noskova, J., Ivashechkin, M., Matas, J., 2020. Magsac++, a fast, reliable and accurate robust estimator. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1304–1312.

Chin, T.-J., Suter, D., 2017. The maximum consensus problem: recent algorithmic advances. *Synthesis Lectures on Computer Vision*, 7(2), 1–194.

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

Holland, P. W., Welsch, R. E., 1977. Robust regression using iteratively reweighted least-squares. *Communications in Statistics - Theory and Methods*, 6(9), 813-827.

Huber, P. J., 2004. *Robust statistics*. 523, John Wiley & Sons.

Lebeda, K., Matas, J., Chum, O., 2012. Fixing the locally optimized ransac–full experimental evaluation. *British Machine Vision Conference*, Citeseer, 1–11.

Li, J., Hu, Q., Ai, M., 2016. Robust feature matching for remote sensing image registration based on $L_-\{q\}$-estimator. *IEEE Geoscience and Remote Sensing Letters*, 13(12), 1989–1993.

Li, J., Hu, Q., Ai, M., 2020a. RIFT: Multi-modal image matching based on radiation-variation insensitive feature transform. *IEEE Transactions on Image Processing*, 29, 3296–3310.

Li, J., Zhao, P., Hu, Q., Ai, M., 2020b. Robust point cloud registration based on topological graph and Cauchy weighted lq-norm. *ISPRS Journal of Photogrammetry and Remote Sensing*, 160, 244–259.

Mittal, S., Anand, S., Meer, P., 2012. Generalized projection-based M-estimator. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(12), 2351–2364.

Raguram, R., Chum, O., Pollefeys, M., Matas, J., Frahm, J.-M., 2012. USAC: a universal framework for random sample consensus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8), 2022–2038.

Rusu, R. B., Blodow, N., Beetz, M., 2009. Fast point feature histograms (fpfh) for 3d registration. *IEEE International Conference on Robotics and Automation*, 3212–3217.

Zhong, Y., 2009. Intrinsic shape signatures: A shape descriptor for 3d object recognition. *IEEE International Conference on Computer Vision Workshops*, 689–696.