2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Madrid, Spain, October 1-5, 2018

Robust Camera Pose Estimation via Consensus on Ray Bundle and Vector Field

Haoang Li, Ji Zhao, Jean-Charles Bazin, Lei Luo, Junlin Wu and Jian Yao

Abstract— **Estimating the camera pose requires point correspondences. However, in practice, correspondences are inevitably corrupted by outliers, which affects the pose estimation. We propose a general and accurate outlier removal strategy for robust camera pose estimation. The proposed strategy can detect outliers by leveraging the fact that only inliers comply with two effective consensuses, i.e.,** ***3D ray bundle consensus*** **and** ***2D vector field consensus*****. Our strategy has a nested structure. First, the outer module utilizes the 3D ray bundle consensus. We define the likelihood based on the probabilistic mixture model and maximize it by the expectation-maximization (EM) algorithm. The inlier probability of each correspondence and the camera pose are determined alternately. Second, the inner module exploits the 2D vector field consensus to refine the probabilities obtained by the outer module. The refinement based on the Bayesian rule facilitates the convergence of the outer module and improves the accuracy of the entire framework. Our strategy can be integrated into various existing camera pose estimation methods which are originally vulnerable to outliers. Experiments on both synthesized data and real images have shown that our approach outperforms state-of-the-art outlier rejection methods in terms of accuracy and robustness.**

I. INTRODUCTION

Camera pose estimation is a crucial technology for vision-based robot navigation. It has been widely applied in various fields including simultaneous localization and mapping (SLAM) [1], structure from motion (SfM) [2], and augmented reality [3]. Camera pose estimation can be roughly classified into two main categories with respect to the type of pose: absolute pose [4] and relative pose [5]. The absolute pose problem is to determine the rotation and translation which align the camera to the world frame; the relative pose problem is to estimate the relative orientation and position between two cameras (or two images). In this paper, we focus on the absolute pose problem.

To estimate the absolute camera pose, a set of 3D-to-2D point correspondences is required. They can be obtained by matching points of the 3D map in the world frame and points of the 2D image [1] [6]. However, in practice, the set of 3D-to-2D point correspondences may contain lots of mismatches (outliers), due to various reasons such as repetitive patterns, occlusions, visual ambiguities and viewpoint differences. The outliers dramatically affect the accuracy of the estimated camera pose, leading to significant drift error of location over time. Existing pose estimation methods robust to outliers either are designed for specific scenarios with strong assumptions (and thus lack generality) [7]–[10] or cannot guarantee accuracy and efficiency simultaneously [6], [11]–[13]. In contrast, we propose a general outlier removal strategy with high accuracy and high efficiency.

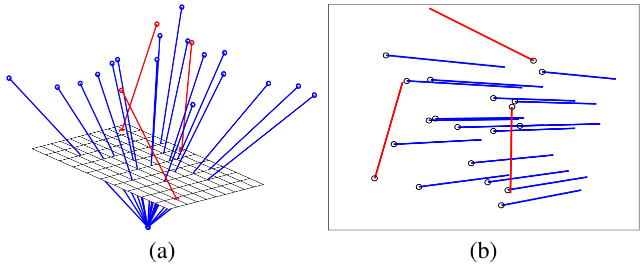


Fig. 1. Our two consensus constraints for robust camera pose estimation (color blue and red represent the inliers and outliers of observations, respectively): (a) 3D ray bundle consensus; (b) 2D vector field consensus.

The proposed outlier removal strategy for camera pose estimation has a nested structure composed of the outer and inner modules. The outer module alternately determines the inlier probability of each correspondence and estimates the camera pose; the inner module refines the probabilities obtained by the outer module. Even if the inner module is not used, the outer module can work independently and reliably in most cases, as we will show in the experiments. With the aid of the inner module, the outer module converges more robustly and leads to more accurate results. To differentiate the inliers from the outliers, both outer and inner modules leverage the fact that only inliers comply with certain consensuses, while outliers do not.

Specifically, the outer module is based on the *3D ray bundle consensus* (RBC) which is shown in Fig. 1(a). A bundle of rays passing through the inliers of 3D-to-2D point correspondences intersect at a common point, i.e., the optical center of the camera, while the rays formed by outliers have arbitrary directions. The inner module is based on the *2D vector field consensus* (VFC). We define a virtual camera and project 3D points to its image, so that original 3D-to-2D correspondences are mapped as 2D-to-2D correspondences. Then a set of 2D vectors is formed by these 2D-to-2D correspondences, and the result is shown in Fig. 1(b). The vector inliers share a regular orientation trend, while the outliers are disordered. From another perspective, a vector field fitted by inliers is much smoother than the field fitted by the combination of inliers and outliers [14].

Overall, we propose a nested outlier removal strategy for camera pose estimation, leveraging these two consensus constraints simultaneously. The main contributions of the

This work is partially supported by the National Key R&D Program of China (Project No. 2017YFB1302400) and the National Natural Science Foundation of China (Project No. 41571436).

H. Li, L. Luo, J. Wu and J. Yao are with CVRS Lab, Wuhan University, China. J. Zhao is with the ReadSense Ltd., Shanghai, China. J.C. Bazin is with Computational Media Lab, KAIST, South Korea.

978-1-5386-8093-3/18/$31.00 ©2018 IEEE

proposed strategy are summarized as follows:

- 3D RBC is utilized by the outer module. We define the likelihood based on the probabilistic mixture model and maximize it by the expectation-maximization (EM) algorithm. The inlier probability of each correspondence and the camera pose are determined alternately;
- 2D VFC is exploited by the inner module to refine the probabilities obtained by the outer module. The refinement based on the Bayesian rule facilitates the convergence of the outer module and improves the accuracy of the entire framework;
- The proposed outlier removal strategy is general. It can be easily integrated into various existing pose estimation methods which are originally vulnerable to outliers.

Experiments on both synthesized data and real images have shown that our approach outperforms state-of-the-art methods in terms of accuracy and robustness.

II. Related Work

Numerous previous work provide reliable camera poses in the outlier-free scenarios, like DLT [15], EPnP [4], DLS [16] and OPnP [17]. However, outliers are inevitable in practice, and they highly affect the accuracy and robustness of these methods. To solve this problem, some robust approaches which can handle outliers have been proposed. These methods can be roughly classified into two main categories according to whether the prior knowledge of partial camera pose is needed [8]–[10] or not [6], [7], [11]–[13].

On the one hand, some robust approaches do not rely on any prior pose information. Well-known RANSAC [11] has been successfully combined with existing camera pose estimation methods like EPnP [4] and OPnP [17] to detect outliers. However, RANSAC-alike methods hardly deal with high outlier ratios and are well-known to become inefficient when the number of correspondences is large. Ferraz *et al.* [7] proposed an outlier rejection strategy which was inherently integrated into EPnP [4]. This strategy detects outliers with negligible time, so it can keep high efficiency when handling a large number of correspondences. However, it is only designed for EPnP [4] and can hardly be extended for other methods. Moreno *et al.* [12] proposed a more general strategy. They used a robust kernel to reject outliers, but the actual performance is sensitive to the parameters chosen for the kernel. Recently, Speciale *et al.* [13] solved the consensus set maximization by the linear matrix inequality. While this approach can determine all inliers in theory, it is not suitable for the practical uses due to high computational complexity. Camposeco *et al.* [6] exploited the toroidal constraint to remove outliers. This method is relatively efficient, but its accuracy is limited because the estimated camera position is only an approximation. In the experiments, we will compare our approach to these methods.

On the other hand, some robust methods require the prior knowledge of partial camera pose, like full orientation [8] or vertical direction [9] [10]. Larson *et al.* [8] designed an outlier rejection strategy, assuming that the full camera orientation is known apriori. Svärm *et al.* [9] proposed an approach using the known vertical direction of the camera to filter out outliers. These two methods [8] [9] aim at determining the maximum inlier set of correspondences, but they are computationally expensive. In addition, Zeisl *et al.* [10] proposed a voting-based method to detect outliers, which also assumes the known vertical direction of the camera. While these approaches can provide accurate results for certain occasions, they are less general because the prior knowledge of partial camera pose cannot be given beforehand in many practical cases.

In contrast, the proposed outlier removal strategy does not require any prior information regarding the camera pose. In addition, it can be easily integrated into various existing camera pose estimation approaches which cannot handle outliers. Therefore, our strategy is general. Furthermore, by fully exploiting the 3D RBC and 2D VFC, our strategy leads to high accuracy and robustness, even for the challenging cases of high outlier ratios and large number of correspondences.

III. Problem Formulation

Throughout this paper, all the matrices, vectors and scalars are denoted as bold capital like "$\mathbf{R}$", bold lowercase like "$\mathbf{t}$", and plain letters like "i", respectively. Vectors are column-wise. We use "$\propto$" to represent equality regardless of scale, and use $\|\cdot\|$ to denote the L_2 norm of vector.

We assume that the intrinsic matrix $\mathbf{K}$ is known from camera calibration [15]. Given N 3D points $\{\mathbf{p}_i\}_{i=1}^N$ in the world frame and corresponding 2D points $\{\mathbf{q}_i\}_{i=1}^N$ in the image, we exploit the 3D-to-2D correspondence set $\mathcal{M} = \{(\mathbf{p}_i, \mathbf{q}_i)\}_{i=1}^N$ to estimate the camera pose consisting of rotation $\mathbf{R} \in SO3$ and translation $\mathbf{t} \in \mathbb{R}^3$. Our aim is to robustly compute accurate $\mathbf{R}$ and $\mathbf{t}$ when $\mathcal{M}$ is corrupted by some mis-matches (outliers). Our two consensus constraints are introduced as follows.

A. 3D Ray Bundle Consensus

Ideally, a bundle of 3D rays formed by *inliers* of the correspondence set $\mathcal{M}$ intersect at the optical center of the camera (shown in Fig. 1(a)). Any ray passing through a 3D point $\mathbf{p}_i = [p_i^x, p_i^y, p_i^z, 1]^\top$ and a 2D point $\mathbf{q}_i = [q_i^u, q_i^v, 1]^\top$ (in homogeneous coordinates) complies with this consensus, which can be expressed as

$$\mathbf{q}_i \;\propto\; \mathbf{K}\,[\mathbf{R}|\mathbf{t}]\,\mathbf{p}_i. \tag{1}$$

Most existing camera pose estimation methods [4], [15]–[17] for the outlier-free case are based on the transformation of Eq. (1). Our outlier removal strategy can be integrated into all these pose estimation methods, no matter how Eq. (1) is transformed. Due to limited space, we only take classical DLT [15] as example in this paper to show how Eq. (1) is transformed and integrated into the proposed strategy.

Let $\{s_i\}_{i=1}^{12}$ denote the entries of 3×4 projection matrix $\mathbf{P} = \mathbf{K}\,[\mathbf{R}|\mathbf{t}]$ in the row-major order. We analyze two cases: (i) if the correspondence $\mathcal{C}_i = (\mathbf{p}_i, \mathbf{q}_i) \in \mathcal{M}$ is inlier, in the noise-free case, DLT reformulates Eq. (1) as $\mathbf{A}_i\mathbf{s} = \mathbf{0}$, where $\mathbf{A}_i$ has the form:

$$\begin{bmatrix} p_i^x & p_i^y & p_i^z & 1 & 0 & 0 & 0 & 0 & \text{-}q_i^u p_i^x & \text{-}q_i^u p_i^y & \text{-}q_i^u p_i^z & \text{-}q_i^u \\ 0 & 0 & 0 & 0 & p_i^x & p_i^y & p_i^z & 1 & \text{-}q_i^v p_i^x & \text{-}q_i^v p_i^y & \text{-}q_i^v p_i^z & \text{-}q_i^v \end{bmatrix},$$

and $\mathbf{s} = [s_1, s_2, \cdots, s_{12}]^\top$. Under the presence of noise, $\mathbf{A}_i\mathbf{s} = \mathbf{0}$ cannot be strictly satisfied, and we define a 2D error which approximately follows the Gaussian distribution as $\boldsymbol{\epsilon}_i = \mathbf{A}_i\mathbf{s} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$, where σ denotes the standard derivation and $\mathbf{I}$ represents the 2×2 identity matrix; (ii) if the correspondence $\mathcal{C}_j = (\mathbf{p}_j, \mathbf{q}_j) \in \mathcal{M}$ is outlier, the output space of the error $\boldsymbol{\xi}_j = \mathbf{A}_j\mathbf{s}$ is a 2D bounded region whose area equals to a constant value a. We thus assume that $\boldsymbol{\xi}_j$ follows the uniform distribution $1/a$.

Therefore, the 3D ray bundle consensus (RBC) constraint can be summarized as follows. For the inlier of 3D-to-2D correspondences, the error $\boldsymbol{\epsilon}_i = \mathbf{A}_i\mathbf{s}$ follows the *Gaussian distribution*; for the outlier, the error $\boldsymbol{\xi}_j = \mathbf{A}_j\mathbf{s}$ follows the *uniform distribution*.

B. 2D Vector Field Consensus

We begin with introducing the concept of the vector field. Let a vector set $\mathcal{V}$ contain N vectors on a 2D plane (shown in Fig. 1(b)). Vector $\mathbf{v}_i \in \mathcal{V}$ is expressed by $\mathbf{v}_i = \mathbf{x}'_i - \mathbf{x}_i$, where $\mathbf{x}_i$ and $\mathbf{x}'_i$ denote the start and end points, respectively. Vector interpolation is to assign arbitrary position $\mathbf{x}$ on the plane with a vector $\mathbf{y}$. The vector $\mathbf{y}$ is determined by the vector-valued function $\mathbf{f}$ as $\mathbf{y} = \mathbf{f}(\mathbf{x})$. We call $\mathbf{f}$ the *vector field* and present how to fit $\mathbf{f}$ using sparse vector samples of $\mathcal{V}$.

Fitting a general function $\mathbf{f}$ by the samples of $\mathcal{V}$ as $\min_{\mathbf{f}} \left\{ \sum_{i=1}^{N} \|\mathbf{v}_i - \mathbf{f}(\mathbf{x}_i)\|^2 \right\}$ is an ill-posed problem because it depends on the type or properties of $\mathbf{f}$ (model, degree, smoothness, etc). In our context, inlier samples should share a certain pattern consensus (e.g. regular orientation trend). Therefore, we use the property that the vector field $\mathbf{f}$ should be smooth [14]. By enforcing the smoothness constraint of the field, we can solve the fitting problem based on the Tiknonov regularization [18]. The regularization is conducted in the reproducing kernel Hilbert space $\mathcal{H}$ [19] as

$$\arg\min_{\mathbf{f}} \left\{ \sum_{i=1}^{N} \|\mathbf{v}_i - \mathbf{f}(\mathbf{x}_i)\|^2 + \lambda \|\mathbf{f}\|_{\mathcal{H}}^2 \right\}, \tag{2}$$

where λ is the regularization coefficient to control the trade-off between the data term and the smoothness term; $\|\cdot\|_{\mathcal{H}}$ denotes the norm in the space $\mathcal{H}$. The smaller value of $\|\mathbf{f}\|_{\mathcal{H}}^2$ (cf. Section V-B for specific definition) is, the smoother $\mathbf{f}$ is.

Therefore, the 2D vector field consensus (VFC) constraint can be summarized as follows. The vector inliers share a regular orientation trend, while the outliers are disordered. The vector field fitted only by the inliers is smoother than the field fitted by the samples corrupted by some outliers.

C. Overview of the Proposed Outlier Removal Strategy

The proposed outlier removal strategy for camera pose estimation has a nested structure composed of the outer and inner modules.

a) Outer module: The outer module identifies inliers and outliers by exploiting the fact that only inliers follow the 3D RBC, while outliers do not. We define the likelihood based on the probabilistic mixture model of the Gaussian distribution for inliers and the uniform distribution for outliers. In the likelihood, each correspondence is associated with a latent variable indicating it is inlier or outlier. We maximize the likelihood by the expectation-maximization algorithm to alternately determine the inlier probability of each correspondence and estimate the camera pose.

b) Inner module: The inner module refines the probabilities obtained by the outer module based on the 2D VFC. We leverage the fact that varying the weights of the vector samples leads to the smoothness difference between the fitted vector fields. Each original probability from the outer module is treated as prior and updated by the Bayesian rule.

Next, we present details of the outer and inner modules in Section IV and Section V, respectively.

IV. OUTER MODULE BASED ON THE 3D RAY BUNDLE CONSENSUS

In this section, we formulate the camera pose estimation under the presence of outliers as maximizing the likelihood with latent variables. We first present the likelihood definition based on the 3D RBC, followed by introducing how to maximize this likelihood using the expectation-maximization (EM) algorithm.

A. Likelihood Definition

Let γ denote the unknown inlier ratio of the 3D-to-2D point correspondence set $\mathcal{M}$. We use two types of error following the Gaussian distribution $\mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$ and the uniform distribution $1/a$ (cf. Section III-A) to define the likelihood. Specifically, the likelihood which is the mixture model of these two types of distributions is defined as

$$L(\mathcal{M}|\Omega) = \prod_{i=1}^{N} \left(\frac{\gamma}{2\pi\sigma^2} \exp\{-\frac{\|\mathbf{A}_i\mathbf{s}\|^2}{2\sigma^2}\} + \frac{1-\gamma}{a} \right), \tag{3}$$

where $\Omega = \{\mathbf{s}, \sigma^2, \gamma\}$ is the set of unknown parameters. Traditional maximum likelihood estimation (MLE) [20] for determining the unknown variables in Ω can be formed as $\arg\max_{\Omega} \{\ln L(\mathcal{M}|\Omega)\}$. Since we cannot make sure each observation $\mathbf{A}_i\mathbf{s}$ follows the Gaussian or uniform distribution, maximizing Eq. (3) by traditional MLE is unrealistic.

To solve this problem, we assign each correspondence $\mathcal{C}_i$ with a latent variable $z_i \in \{0, 1\}$, where $z_i = 1$ and $z_i = 0$ indicates $\mathcal{C}_i$ is inlier or outlier, respectively. We reformulate $\ln L(\mathcal{M}|\Omega)$ following the complete-data likelihood [20] as

$$\sum_{i=1}^{N} \big(p_i \cdot \ln L(z_i = 1|\Omega)\big) + \sum_{i=1}^{N} \big((1-p_i) \cdot \ln L(z_i = 0|\Omega)\big),$$

where $p_i = P(z_i = 1|\Omega)$ is the unknown inlier probability of the correspondence $\mathcal{C}_i$. We omit some terms which are independent of Ω and obtain the negative log-likelihood as

$$\begin{aligned} F(\Omega) = & \frac{1}{2\sigma^2} \sum_{i=1}^{N} p_i \|\mathbf{A}_i\mathbf{s}\|^2 + (\ln\sigma^2 - \ln\gamma) \sum_{i=1}^{N} p_i \\ & - \ln(1-\gamma) \sum_{i=1}^{N} (1 - p_i). \end{aligned} \tag{4}$$

In the next section, we will determine the unknown parameters in Ω by minimizing Eq. (4).

B. Likelihood Maximization

The inlier probability p_i is unknown, so we cannot directly take partial derivatives of Eq. (4) with respect to the unknown variables in the parameter set Ω. To solve this problem, we exploit the iterative expectation-maximization (EM) algorithm [20]. The EM algorithm contains two main steps during each iteration, i.e., the expectation step (E-step) and the maximization step (M-step). The E-step updates the inlier probabilities based on the estimated parameters from the previous iteration, and the M-step exploits the newly-updated probabilities to obtain more precise parameters.

a) E-step: For each correspondence $\mathcal{C}_i = (\mathbf{p}_i, \mathbf{q}_i)$, its inlier probability p_i is updated as

$$p_i = \frac{\gamma e^{-\frac{\|\mathbf{A}_i\mathbf{s}\|^2}{2\sigma^2}}}{\gamma e^{-\frac{\|\mathbf{A}_i\mathbf{s}\|^2}{2\sigma^2}} + (1-\gamma)\frac{(\sqrt{2\pi}\sigma)^2}{a}} \qquad (5)$$

b) M-step: After the probability p_i is updated, we compute the unknown parameters $\Omega = \{\mathbf{s}, \sigma^2, \gamma\}$ by minimizing Eq. (4). We take partial derivatives of Eq. (4) with respect to the standard deviation σ^2 and the inlier ratio γ respectively, and then set the results as zero. After that, new σ^2 and γ can be obtained as follows:

$$\sigma^2 = \sum_{i=1}^{N} p_i \|\mathbf{A}_i\mathbf{s}\|^2 / \sum_{i=1}^{N} p_i \text{ and } \gamma = \sum_{i=1}^{N} p_i / n. \qquad (6)$$

To minimize Eq. (4) with respect to the pose parameters $\mathbf{s}$, we extract the term related to $\mathbf{s}$ from Eq. (4) as

$$\arg\min_{\mathbf{s}} \sum_{i=1}^{N} p_i \|\mathbf{A}_i\mathbf{s}\|^2 \Rightarrow \arg\min_{\mathbf{s}} \|\mathbf{D}^{1/2}\mathbf{A}\mathbf{s}\|, \qquad (7)$$

where the block matrix $\mathbf{A}$ is composed of $\{\mathbf{A}_i\}_{i=1}^N$ in a column, and the diagonal probability matrix $\mathbf{D}$ consists of $\{p_i\}_{i=1}^N$, i.e., $\mathbf{D} = \mathrm{diag}(p_1, p_2, \cdots, p_n)$. We use SVD [15] to solve $\mathbf{s}$, subjected to the constraint $\|\mathbf{s}\| = 1$. Then the rotation $\mathbf{R}$ and the translation $\mathbf{t}$ are recovered from $\mathbf{s}$ by applying the inverse of the known intrinsic matrix $\mathbf{K}$ and enforcing the orthogonality constraint $\mathbf{R}^\top\mathbf{R} = \mathbf{I}$ [15].

Along the iterations, the probability p_i has two opposite variation trends. (i) p_i of the outlier will decrease, so outliers have less weights when involving in the camera pose estimation; (ii) p_i of the inlier will increase and inliers exert more effects. Therefore, the EM framework can reduce the effects of outliers, which in turn provides a more robust estimation of the camera pose $\mathbf{R}$ and $\mathbf{t}$. In addition, we can identify the inlier set $\mathcal{I}$ under the assumption that the correspondence $\mathcal{C}_i$ can be regarded as inlier if its inlier probability p_i is high, i.e. $p_i \geqslant \tau$ (τ is set as 0.8 in this paper).

V. Inner Module Based on the 2D Vector Field Consensus

The outer module can update the inlier probabilities at each E-step, but the updated probabilities are not accurate unless the iteration correctly converges. Low-accuracy probabilities will affect the following M-step, and further reduce the accuracy and robustness of the whole framework. In fact, we can refine the probabilities right after each E-step by the inner module which is embedded into the outer module. The inner module exploiting the 2D VFC is introduced as follows.

A. 2D Vector Computation

To leverage the 2D VFC, the first step is to obtain a set of 2D vectors. Given the 3D-to-2D point correspondence set $\mathcal{M} = \{(\mathbf{p}_i, \mathbf{q}_i)\}_{i=1}^N$, a set of 2D vectors are computed by the 2D points $\{\mathbf{q}_i\}_{i=1}^N$ and the projections of 3D points $\{\mathbf{p}_i\}_{i=1}^N$. Specifically, each M-step of the outer module estimates the rotation $\tilde{\mathbf{R}}$ and the translation $\tilde{\mathbf{t}}$ based on Eq. (7). We define a virtual camera by setting its pose as $\tilde{\mathbf{R}}$ and $\tilde{\mathbf{t}}$. Then we use this camera to project 3D points $\{\mathbf{p}_i\}_{i=1}^N$ to its image, obtaining a set of 2D projections $\{\mathbf{q}'_i\}_{i=1}^N$. Therefore, original 3D-to-2D correspondence set $\mathcal{M}$ are mapped as 2D-to-2D correspondence set $\mathcal{Q} = \{(\mathbf{q}'_i, \mathbf{q}_i)\}_{i=1}^N$. Based on $\mathcal{Q}$, we can obtain the 2D vector set $\mathcal{V} = \{\mathbf{v}_i\}_{i=1}^N$ where each vector $\mathbf{v}_i$ is computed as $\mathbf{v}_i = \mathbf{q}'_i - \mathbf{q}_i$. Intuitively, if the correspondence $(\mathbf{p}_i, \mathbf{q}_i) \in \mathcal{M}$ is inlier, corresponding vector $\mathbf{v}_i \in \mathcal{V}$ is also inlier complying with the 2D VFC. In the next section, we will use the vector set $\mathcal{V}$ to refine the inlier probabilities.

B. Inlier Probability Refinement

Based on the vector set $\mathcal{V}$ obtained in Section V-A, original inlier probabilities $\mathcal{P} = \{p_i\}_{i=1}^N$ obtained by the outer module are refined *one by one* by the Bayesian rule.

First, we use the original $\mathcal{P}$ to fit the vector field $\mathbf{f}$ that is applicable for the refinement of each probability in $\mathcal{P}$. The vector field fitting expressed in Eq. (2) is slightly adapted as

$$\arg\min_{\mathbf{f}} \left\{ \sum_{i=1}^{N} p_i \|\mathbf{v}_i - \mathbf{f}(\mathbf{q}_i)\|^2 + \lambda\|\mathbf{f}\|_{\mathcal{H}}^2 \right\}, \qquad (8)$$

which can be regarded as the *weighted* vector field fitting (the inlier probability p_i is the weight). Based on the representer theorem [19], we fit the optimal vector field $\mathbf{f}$ by the vector samples $\mathcal{V} = \{\mathbf{v}_i\}_{i=1}^N$. Arbitrary position $\mathbf{x}$ is assigned with an interpolated vector $\mathbf{f}(\mathbf{x})$ as:

$$\mathbf{f}(\mathbf{x}) = \sum_{i=1}^{N} e^{-\beta\|\mathbf{x}-\mathbf{q}_i\|^2}\mathbf{c}_i, \qquad (9)$$

where kernel $e^{-\beta\|\mathbf{x}-\mathbf{q}_i\|^2}$ encodes the effect of the vector sample $\mathbf{v}_i$ (start point is $\mathbf{q}_i$) for interpolating a vector $\mathbf{f}(\mathbf{x})$; coefficients $\{\mathbf{c}_i\}_{i=1}^N$ is determined by the linear system:

$$(\mathbf{\Gamma} + \lambda\mathbf{D}^{-1})[\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_N]^\top = [\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_N]^\top, \qquad (10)$$

where the kernel matrix $\mathbf{\Gamma}$ is composed of the kernel $e^{-\beta\|\mathbf{x}-\mathbf{q}_i\|^2}$, and the probability matrix $\mathbf{D}$ is defined in Section IV-B. Experiments have shown that our strategy is not sensitive to the selection of parameters β and λ.

When the sample number N is large, solving the above linear system is relatively time-consuming. We thus exploit the sparse approximation technique [14] for acceleration. This technique provides a sub-optimal solution of $\mathbf{f}(\mathbf{x})$ by using sparse basis functions to approximately represent the original space $\mathcal{H}$ (cf. Section III-B). It is efficient and the result is still accurate enough for the probability refinement.

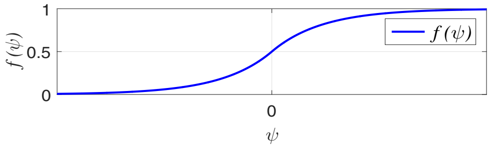


Fig. 2. Evolution of the function $f(\psi)$ with respect to ψ.

We take the probability $p_i \in \mathcal{P}$ as example to present how to refine each probability in $\mathcal{P}$. After obtaining the vector field $\mathbf{f}$ using the original $\mathcal{P}$, we make two hypotheses for the vector $\mathbf{v}_i$ and revise p_i accordingly. We first assume that $\mathbf{v}_i$ is definitely inlier, i.e., we set p_i as 1 and fix other probabilities of $\mathcal{P}$. Then a new vector field $\bar{\mathbf{f}}_i$ is fitted based on Eq. (8). Similarly, we assume that $\mathbf{v}_i$ is definitely outlier and set p_i as 0, then another vector field $\underline{\mathbf{f}}_i$ is fitted. Note that the variation of the probability p_i leads to the smoothness difference between the fitted vector fields. Specifically, we use the norm $\|\mathbf{f}\|^2_{\mathcal{H}} = \sqrt{\frac{1}{N}(\mathbf{C}^\top \mathbf{\Gamma} \mathbf{C})}$ where $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_N]^\top$ to evaluate the smoothness of a vector field $\mathbf{f}$ [19]. The smoothnesses $\|\mathbf{f}\|^2_{\mathcal{H}}$, $\|\bar{\mathbf{f}}_i\|^2_{\mathcal{H}}$ and $\|\underline{\mathbf{f}}_i\|^2_{\mathcal{H}}$ are computed by the same kernel matrix $\mathbf{\Gamma}$ and different $\mathbf{C}$. Since the probability matrix $\mathbf{D}$ composed of p_i directly determines $\mathbf{C}$ (cf. Eq. (10)), the variation of p_i leads to the smoothness difference between the fitted fields. We refine p_i by using above two hypotheses for the vector $\mathbf{v}_i$ that leads to the smoothness difference.

We use "I" or "O" to denote the *hypothesis* that the vector $\mathbf{v}_i$ is inlier or outlier. We use "V" to denote the *consequence* that $\mathbf{v}_i$ is identified as inlier. Besides, we rewrite p_i as $P_i(\mathrm{I})$ to represent the *prior* inlier probability of $\mathbf{v}_i$, and use $P_i(\mathrm{I}|\mathrm{V})$ to denote corresponding *posterior* probability. The prior $P_i(\mathrm{I})$ is updated to the posterior $P_i(\mathrm{I}|\mathrm{V})$ by the Bayesian rule as

$$P_i(\mathrm{I}|\mathrm{V}) = P_i(\mathrm{I}) \frac{P_i(\mathrm{V}|\mathrm{I})}{P_i(\mathrm{I})P_i(\mathrm{V}|\mathrm{I}) + P_i(\mathrm{O})P_i(\mathrm{V}|\mathrm{O})}, \quad (11)$$

where $P_i(\mathrm{O}) = 1 - P_i(\mathrm{I})$. Therefore, to refine $P_i(\mathrm{I})$, the key step is to determine the conditional probabilities $P_i(\mathrm{V}|\mathrm{I})$ and $P_i(\mathrm{V}|\mathrm{O})$. We use the function $f(\psi_i)$ to define $P_i(\mathrm{V}|\cdot)$ as

$$P_i(\mathrm{V}|\cdot) = f(\psi_i) = \frac{1}{2} \cdot \begin{cases} e^{\psi_i}, & \text{if} \quad \psi_i < 0, \\ 2 - e^{-\psi_i}, & \text{if} \quad \psi_i > 0. \end{cases} \quad (12)$$

where ψ_i is the smoothness difference defined as $\psi_i = \|\mathbf{f}\|^2_{\mathcal{H}} - \|\bar{\mathbf{f}}_i\|^2_{\mathcal{H}}$ for $P_i(\mathrm{V}|\mathrm{I})$ and as $\psi_i = \|\mathbf{f}\|^2_{\mathcal{H}} - \|\underline{\mathbf{f}}_i\|^2_{\mathcal{H}}$ for $P_i(\mathrm{V}|\mathrm{O})$.

Fig. 2 shows the evolution of $f(\psi_i)$ with respect to ψ_i. Positive (negative) ψ_i indicates the new vector field $\bar{\mathbf{f}}_i$ or $\underline{\mathbf{f}}_i$ is smoother (rougher) than the original field $\mathbf{f}$, i.e., the vector $\mathbf{v}_i$ has greater (smaller) probability to be identified as inlier. In addition, the larger absolute value of ψ_i is, the greater extent that $P_i(\mathrm{V}|\cdot)$ deviates from 0.5 should be.

C. Collaboration Between the Outer and Inner Modules

Our full outlier removal strategy for camera pose estimation has a nested structure composed of the outer and inner modules. Even if the inner module is not used, the outer module can work independently and reliably in most cases. When the number of correspondences N is very small, the outer module may be unstable, since two types of distributions in the mixture model (cf. Section IV-A) are not distinct enough. With the refinement of the inner module, the outer module converges more robustly, as we will show in the experiments. Specifically, the inner module can correct or accelerate the polarization of the inlier probability p_i obtained by the outer model (approaching to 1 for inliers and 0 for outliers). Besides, the inner module has less effect when p_i is polarized, so for efficiency, we only refine p_i by the inner module when $p_i \in [0.1, 0.9]$. The proposed strategy is generalized in Algorithm 1.

Algorithm 1: Outlier Removal Strategy for Camera Pose Estimation

```
Input: 3D-to-2D correspondences $\{(\mathbf{p}_i, \mathbf{q}_i)\}_{i=1}^N$.
Output: Rotation $\mathbf{R}$, translation $\mathbf{t}$, and inlier set $\mathcal{I}$.
Initialize inlier probabilities $\{p_i\}_{i=1}^N$ with 0.5;
Reformulate Eq. (1) for the outer module;
repeat
    E-step: update the probability $p_i$ by Eq. (5);
    Compute vectors $\{\mathbf{v}_i\}_{i=1}^N$ for the inner module;
    if $p_i \in [0.1, 0.9]$ then
        Fit vector fields $\mathbf{f}$, $\bar{\mathbf{f}}_i$, $\underline{\mathbf{f}}_i$ by Eq. (8);
        Refine the probability $p_i$ by Eq. (11);
    end
    M-step: update $\mathbf{R}$ and $\mathbf{t}$ by Eq. (7);
until $\min F(\Omega)$ in Eq. (4) converges;
Determine the inlier set $\mathcal{I}$ as $\mathcal{I} = \{(\mathbf{p}_i, \mathbf{q}_i) | p_i > 0.8\}$;
```

VI. EXPERIMENTS

To evaluate the proposed outlier removal strategy for camera pose estimation, we have conducted experiments on both synthesized data and real images. We compare our method with existing state-of-the-art approaches in terms of accuracy and efficiency.

A. Synthesized Data

We synthesize a set of 3D-to-2D point correspondences as follows. We define a virtual camera whose focal length is 800 pixels and image size is 640 × 480 pixels. The principal point is located at the center of image. For each trial, 3D points are randomly generated within the interval of $[-2, 2] \times [-2, 2] \times [4, 8]$. We set the ground-truth translation $\mathbf{t}_{\mathrm{true}}$ as the centroid of the synthesized 3D points, and randomly generate the ground-truth rotation $\mathbf{R}_{\mathrm{true}}$. The inlier correspondences are generated by projecting 3D points to the image using $\mathbf{R}_{\mathrm{true}}$ and $\mathbf{t}_{\mathrm{true}}$. Then we perturb projections by zero-mean Gaussian noises with standard deviation of 2 pixels; the outliers are generated by randomizing the locations of 2D points within the image. We conduct 1000 independent trials and report the average results.

We follow the criteria defined in OPnP [17] to quantitatively evaluate the accuracy of the estimated pose. Specifically, the "rotation error" for assessing the rotation $\mathbf{R}$ is defined as $E_{\mathrm{rot}}\,(\mathrm{deg}) = \max_{i=1}^{3}\{\mathrm{acos}\left(\mathrm{dot}(\mathbf{r}^{\mathrm{i}}_{\mathrm{true}}, \mathbf{r}^{\mathrm{i}})\right) \times 180/\pi\}$, where $\mathbf{r}^i_{\mathrm{true}}$ and $\mathbf{r}^i$ are the i-th columns of $\mathbf{R}_{\mathrm{true}}$ and $\mathbf{R}$, respectively; the "translation error" for evaluating the translation $\mathbf{t}$ is defined as $E_{\mathrm{trans}}\,(\%) = \|\mathbf{t}_{\mathrm{true}} - \mathbf{t}\| / \|\mathbf{t}\| \times 100$.

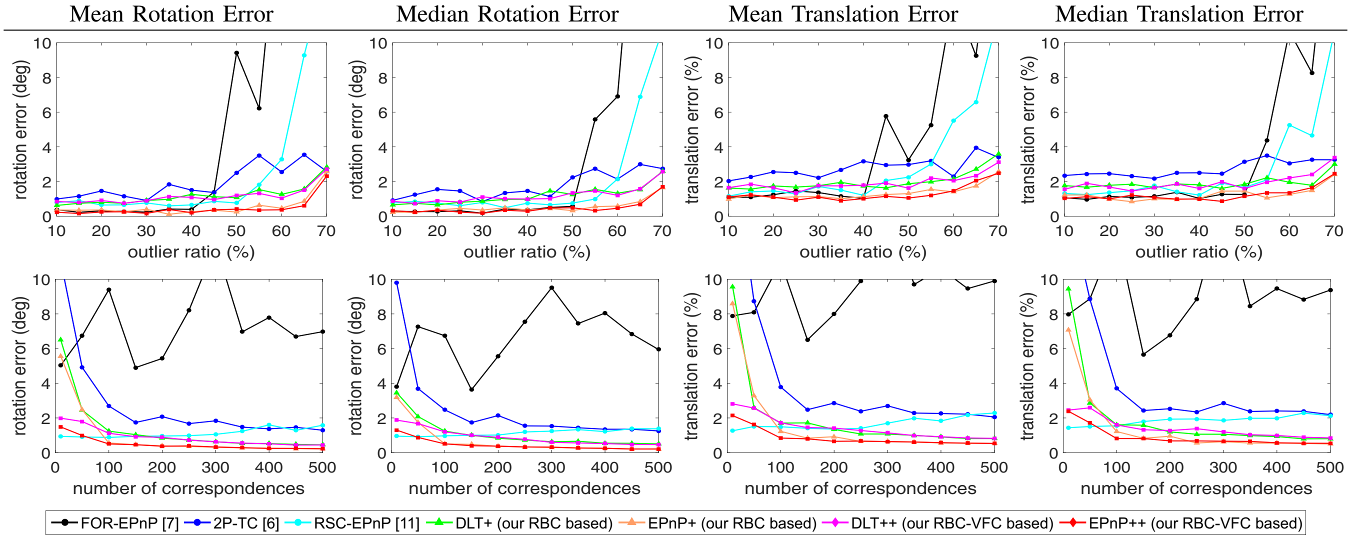


Fig. 3. Accuracy comparison with respect to the outlier ratio (first row) and the number of correspondences (second row). We compare **FOR-EPnP** [7], **2P-TC** [6], **RSC-EPnP** [11] with our **DLT+** and **EPnP+** from $\mathcal{S}_1$, as well as our **DLT++** and **EPnP++** from $\mathcal{S}_2$.

We denote our strategy based on the ray bundle consensus and the vector field consensus (cf. Section III) by **RBC-VFC**. Besides, the outer module of our strategy only leveraging the ray bundle consensus (cf. Section IV) is denoted as **RBC** and tested independently. We integrate our **RBC** and **RBC-VFC** strategies into two popular pose estimation methods: classical DLT [15] and widely-used EPnP [4], respectively. The integration forms two algorithm sets as follows:

- DLT, EPnP with **RBC**: $\mathcal{S}_1 = \{\textbf{DLT+}, \textbf{EPnP+}\}$;
- DLT, EPnP with **RBC-VFC**: $\mathcal{S}_2 = \{\textbf{DLT++}, \textbf{EPnP++}\}$.

We compare our methods from $\mathcal{S}_1$ and $\mathcal{S}_2$ with state-of-the-art ones. As introduced in Section II, approaches based on the maximum inlier set [2] [13] or the prior partial pose [8]–[10] are less practical due to low efficiency or strong assumptions. Therefore, we do not choose them for comparison. Instead, we use the following three methods that are relatively efficient and do not require any pose prior:

- Fast outlier removal strategy for EPnP [7], which is denoted as **FOR-EPnP**;
- Two-point localization method based on the toroidal constraint [6], which is denoted as **2P-TC**;
- Classical RANSAC [11] integrated into EPnP [4], which is denoted as **RSC-EPnP**[1]. This integration can be regarded as the representative of RANSAC-alike methods.

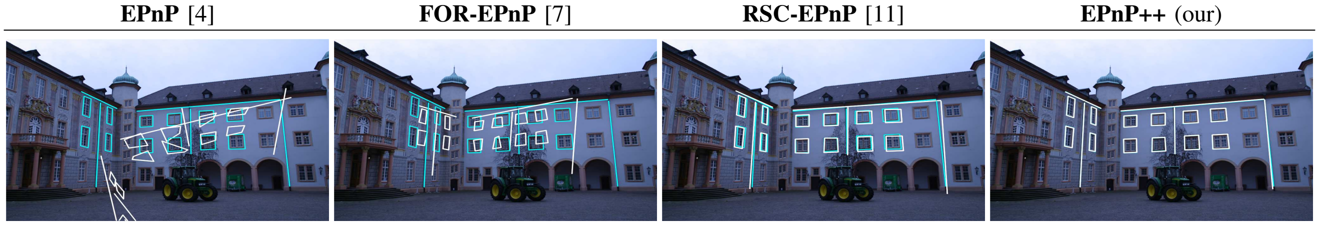
All the above methods are implemented by MATLAB and tested on an Intel Core i7 CPU with 2.40 GHz. In the following, we present comparisons on *accuracy* and *efficiency*.

First, we evaluate various approaches on *accuracy*. We design two groups of experiments with respect to the outlier ratio and the number of 3D-to-2D correspondences. Specifically, for the first group, we set the number of inliers as 50, and change the outlier ratio from 10% to 70%; for the second group, we fix the outlier ratio to 50%, and adjust the total number of correspondences from 10 to 500.

a) Test on the outlier ratio: The first row of Fig. 3 shows the accuracy for an increasing outlier ratio γ. The breakdown point of **FOR-EPnP** is about 50% outliers. Given more outliers, **FOR-EPnP** generates significant errors. While **2P-TC** demonstrates robustness when γ is high, its accuracy is unsatisfactory. The reason is that the estimated camera position of **2P-TC** is just an approximation obtained by only two point correspondences. **RSC-EPnP** becomes unstable with more than 60% outliers, since a high outlier ratio increases the difficulty to obtain an outlier-free minimal solution set. In contrast, our **DLT+** and **EPnP+** from $\mathcal{S}_1$ as well as our **DLT++** and **EPnP++** from $\mathcal{S}_2$ remain robust when γ increases to 65%, thanks to the proposed outlier removal strategy. Note that our **EPnP+** (**EPnP++**) slightly outperforms our **DLT+** (**DLT++**), because the original EPnP [4] is less sensitive to the noise than DLT [15].

b) Test on the number of correspondences: The second row of Fig. 3 reports the accuracy for an increasing number of correspondences N. **FOR-EPnP** consistently generates unsatisfactory results when N varies, since its breakdown point is close to the outlier ratio of this test (50%). **2P-TC** is robust only when N is relatively high. For small N, **2P-TC** hardly retrieves sufficient inliers because lots of inliers are mistakenly discarded by its strict outlier rejection criterion. **RSC-EPnP** maintains the steady but limited accuracy for different N. **RSC-EPnP** only exploits a small number of samples to estimate pose at each iteration, so the accuracy is highly affected by noise. Unlike above-mentioned methods, our **DLT+** and **EPnP+** from $\mathcal{S}_1$ are accurate in most cases, demonstrating the advantage of the outer module of our outlier removal strategy. Besides, their accuracies are improved as N increases, since more inliers are used to compensate for noise. Note that our **DLT+** and **EPnP+** are less stable when N is very small. Our **DLT++** and **EPnP++** from $\mathcal{S}_2$ overcome this limitation by leveraging the inner module of

[1] For RANSAC [11], the inlier threshold based on the re-projection error is set as 8 pixels; the number of iterations is computed with the probability 0.99 that at least one outlier-free minimal solution set is retrieved [15].



(70.85 deg, 105.29%) (18.61 deg, 87.26%) (1.68 deg, 33.80%) (0.28 deg, 5.25%)

Fig. 5. Experimental results on the *Castle-P19* of the EPFL dataset [21]. Results of pose estimation for the query image (with embedded contour) are obtained by **EPnP** [4], **FOR-EPnP** [7], **RSC-EPnP** [11], and our **EPnP++**. The pair of numbers below each image represents the rotation error E_{rot} and translation error E_{trans}. Cyan contours correspond to the manually constructed 2D contours; 3D contours are projected by the estimated poses and the projections are shown in white (cf. Section VI-B). A better alignment between cyan and white contours means more accurate pose.

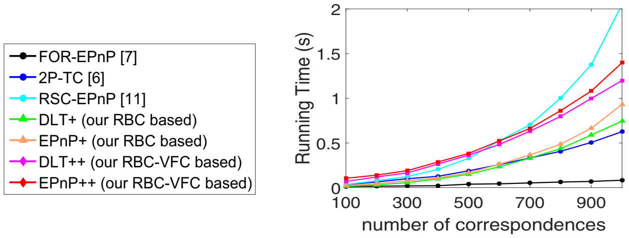


Fig. 4. Computational time with an increasing number of correspondences. We compare **FOR-EPnP** [7], **2P-TC** [6], **RSC-EPnP** [11] with our **DLT+** and **EPnP+** from $\mathcal{S}_1$, as well as our **DLT++** and **EPnP++** from $\mathcal{S}_2$.

our strategy for refinement, obtaining more reliable results.

Second, we compare various methods on *efficiency*. The number of correspondences N increases from 100 to 1000 with an outlier ratio of 50%. Fig. 4 presents the computational time of different approaches. **FOR-EPnP** is the fastest method, and its time cost approximates to a constant whatever N is. **2P-TC** maintains less run-time due to its simple geometric solver. **RSC-EPnP** becomes inefficient when N is high, which is caused by the high number of iterations. Our **DLT+** and **EPnP+** from $\mathcal{S}_1$ are more efficient than **RSC-EPnP**. While our **DLT+** and **EPnP+** are slightly more time-consuming than **2P-TC**, considering their high accuracies, they are still competitive. Compared with our **DLT+** and **EPnP+** from $\mathcal{S}_1$, our **DLT++** and **EPnP++** from $\mathcal{S}_2$ sacrifice partial efficiencies to improve their accuracies. Additional time cost mainly comes from the vector interpolation conducted by the inner module of our outlier removal strategy. Thanks to the sparse approximation technique (cf. Section V-B) for acceleration, the efficiencies of our **DLT++** and **EPnP++** are still acceptable.

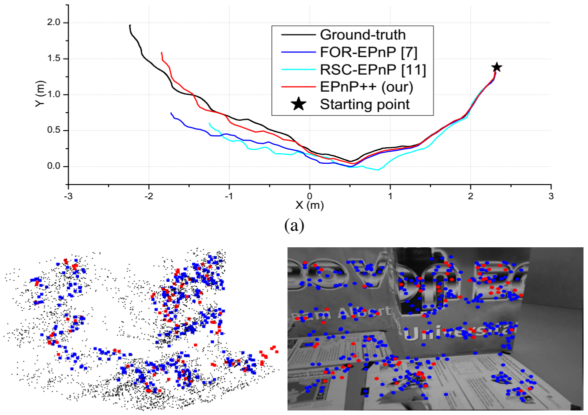
B. Real Images

We also test various methods on real images. We conduct two types of experiments for different purposes: (i) the tests on the EPFL dataset [21] aim at assessing methods using the images with large angular disparities; (ii) the tests on the TUM dataset [22] focus on evaluating approaches using long sequences whose adjacent frames are similar. Due to limited space, we report some representative results. Specifically, we compare the performances of original **EPnP** [4] and its robust versions presented in Section VI-A, including **FOR-EPnP** [7], **RSC-EPnP** [11] and our **EPnP++**. For additional results, readers are invited to refer to the link given above.

a) Tests on the EPFL dataset: We evaluate various methods on the *Castle-P19* of the EPFL dataset [21]. This image set is composed of 19 images of 3072 × 2048 pixels, and the ground-truth poses are given. Repetitive patterns and large angular disparities of these images are prone to lead to mis-matches. We randomly select an image as the query image (shown in Fig. 5), and estimate its pose by various methods. Specifically, we first match 2D points between different images using ORB descriptor [1]. Then we reconstruct a set of 3D points via triangulation [15], using the ground-truth poses, to build the 3D map. Next, we associate the query image with the 3D map to obtain the input 3D-to-2D correspondences following ORB-SLAM [1] (other methods could be used). After that, these correspondences, which inevitably contain outliers, are used to estimate the camera pose. Besides quantitative criteria E_{rot} and E_{trans} (cf. Section VI-A) for accuracy evaluation, we also provide an evaluation of visual alignment. We manually select some corners of windows and wall boundaries in the image and connect them as 2D contours (see cyan contours in Fig. 5). Then we manually match these selected 2D points and use the ground-truth poses for triangulation. The reconstructed 3D points are connected as 3D contours, and we project these 3D contours back to the image using the estimated pose (see the projected white contours in Fig. 5). A better alignment between the manually constructed 2D contours (in cyan) and the projected contours (in white) means more accurate pose.

Fig. 5 shows the representative results obtained by various methods on an input query image. There are over 750 input 3D-to-2D correspondences of which the outlier ratio is about 57.3% (the ground-truth inliers are obtained by manually refining the results of **RSC-EPnP**). Original **EPnP** estimates inaccurate pose, since it is highly vulnerable to outliers. **FOR-EPnP** fails to robustly reject outliers because its breakdown point (about 50% outliers) is lower than the outlier ratio of this test (about 57.3%). For **RSC-EPnP**, we report the average result of 1000 trials due to the uncertainty of random sampling. The overall accuracy of **RSC-EPnP** is relatively satisfactory, but affected by certain trials where no outlier-free minimal solution set is retrieved. In contrast, our **EPnP++** obtains the lowest error. The result proves the superiority of our outlier removal strategy over existing methods, for the challenging case of high outlier ratio and large number of correspondences.

b) Tests on the TUM dataset: Fig. 6 shows the ex-



(a)

(b)

Fig. 6. Test results on the sequence *fr3/structure_texture_far* of the TUM dataset [22]. (a) top view of the estimated trajectories obtained by **FOR-EPnP** [7], **RSC-EPnP** [11], and our **EPnP++**. The black line denotes the ground-truth trajectory; (b) 3D-to-2D point correspondences used for determining the camera pose of a randomly selected frame. The inliers (blue) and outliers (red) are identified by our **EPnP++**.

perimental results on the sequence *fr3/structure_texture_far* of the TUM dataset [22]. This sequence is captured by a hand-held camera moving along a zig-zag structure, and is composed of 938 images of 640 × 480 pixels. High similarity between adjacent frames contributes to correct matching. We evaluate the accuracy of the estimated trajectories by their deviations from the ground-truth trajectory (5.88 m). Note that we assess the *raw* results of the estimated camera positions (on purpose without bundle adjustment [15]) for a more fair, unbiased comparison of the original accuracy of the pose estimation algorithms.

As shown in Fig. 6(a), error accumulations of **FOR-EPnP** and **RSC-EPnP** are significant, while our **EPnP++** has the lowest drift. Specifically, **FOR-EPnP** rejects outliers based on the rank of null-space of certain linear system. This strategy is theoretically reliable, but sensitive to the noise in practice. **RSC-EPnP** may fail to retrieve a pure inlier set. Remaining outliers and great noises result in an inaccurate pose estimation. Both algebraic criterion of **FOR-EPnP** (eigen-vector based residual) and geometric criterion of **RSC-EPnP** (re-projection error) subject to the setting of inlier thresholds. In contrast, the proposed outlier removal strategy is not sensitive to the selection of parameters. Thanks to our strategy, our **EPnP++** is robust to outliers and keeps stable over time. Fig. 6(b) shows the outlier detection result obtained by our **EPnP++** for a randomly selected frame from the sequence.

VII. CONCLUSION

We proposed an outlier removal strategy for camera pose estimation. To detect outliers, we leverage the fact that only inliers comply with the 3D RBC and 2D VFC. Our strategy has two main advantages: (i) generality since it can be integrated into various existing camera pose estimation methods; (ii) high accuracy and robustness to high outlier ratios and small/large number of correspondences. Experiments have shown that our strategy outperforms state-of-the-art outlier rejection methods in terms of accuracy. The future work is to extend our strategy for the line feature and integrate it into the point and line-based pose estimation method [23].

REFERENCES

[1] R. Mur-Artal, J. M. M. Montiel, and J. D. Tards, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, 2015.

[2] D. Campbell, L. Petersson, L. Kneip, and H. Li, "Globally-optimal inlier set maximisation for simultaneous camera pose and feature correspondence," in *IEEE International Conference on Computer Vision*, 2017.

[3] S. Middelberg, T. Sattler, O. Untzelmann, and L. Kobbelt, "Scalable 6-DOF localization on mobile devices," in *European Conference on Computer Vision*, 2014.

[4] V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An accurate O(n) solution to the PnP problem," *International Journal of Computer Vision*, 2008.

[5] H. Li, J. Yao, J.C. Bazin, X. Lu, Y. Xing, and K. Liu, "A monocular SLAM system leveraging structural regularity in Manhattan world," in *IEEE International Conference on Robotics and Automation*, 2018.

[6] F. Camposeco, T. Sattler, A. Cohen, A. Geiger, and M. Pollefeys, "Toroidal constraints for two point localization under high outlier ratios," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[7] L. Ferraz, X. Binefa, and F. Moreno-Noguer, "Very fast solution to the PnP problem with algebraic outlier rejection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[8] V. Larsson, J. Fredriksson, C. Toft, and F. Kahl, "Outlier rejection for absolute pose estimation with known orientation," in *British Machine Vision Conference*, 2016.

[9] L. Svärm, O. Enqvist, F. Kahl, and M. Oskarsson, "City-scale localization for cameras with known vertical direction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[10] B. Zeisl, T. Sattler, and M. Pollefeys, "Camera pose voting for large-scale image-based localization," in *IEEE International Conference on Computer Vision*, 2015.

[11] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, 1981.

[12] F. A. Moreno, J. L. Blanco, and J. Gonzlez-Jimnez, "ERODE: An efficient and robust outlier detector and its application to stereo visual odometry," in *IEEE International Conference on Robotics and Automation*, 2013.

[13] P. Speciale, D. P. Paudel, M. R. Oswald, T. Kroeger, L. V. Gool, and M. Pollefeys, "Consensus maximization with linear matrix inequality constraints," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[14] J. Ma, J. Zhao, J. Tian, A. L. Yuille, and Z. Tu, "Robust point matching via vector field consensus," *IEEE Transactions on Image Processing*, 2014.

[15] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, second edition, 2003.

[16] J. A. Hesch and S. I. Roumeliotis, "A direct least-squares (DLS) method for PnP," in *IEEE International Conference on Computer Vision*, 2011.

[17] Y. Zheng, Y. Kuang, S. Sugimoto, K. Astrom, and M. Okutomi, "Revisiting the PnP problem: A fast, general and optimal solution," in *IEEE International Conference on Computer Vision*, 2013.

[18] F. Girosi, M. Jones, and T. Poggio, "Regularization theory and neural networks architectures," *Neural Computation*, 1995.

[19] C. A. Micchelli and M. Pontil, "On learning vector-valued functions," *Neural Computation*, 2005.

[20] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[21] C. Strecha, W. von Hansen, L. Van Gool, P. Fua, and U. Thoennessen, "On benchmarking camera calibration and multi-view stereo for high resolution imagery," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[22] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

[23] H. Li, J. Yao, X. Lu, and J. Wu, "Combining points and lines for camera pose estimation and optimization in monocular visual odometry," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2017.