A Fast and Deterministic Algorithm for Consensus Set Maximization

ZIRAN XING1,2,3, (Student Member, IEEE), AND ZHIRU SHI2
1Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai 200050, China
2School of Information Science and Technology, ShanghaiTech University, Shanghai 201210, China
3University of Chinese Academy of Sciences, Beijing 100049, China
Corresponding author: Ziran Xing (xingzr@shanghaitech.edu.cn)

This work was supported in part by the Program of Shanghai Subject Chief Scientist (A type) under Grant 17XD1402900 and Grant 15XD1502900, and in part by STCSM under Grant 17JC1403800.

ABSTRACT With the current booming applications of virtual reality, augmented reality, and robotics, efficiently extracting the maximum consensus set among large-scale corrupted data has become a critical challenge. However, existing methods typically focus on optimization and are rarely concerned about the running time. In this paper, we propose a new fast and deterministic algorithm to address the consensus set maximization problem. First, we propose a novel formulation that transforms the original problem into a sequence of decision problems (DPs). Second, we propose an efficient algorithm to assess the feasibility of these DPs. Comprehensive experiments on linear hyper-plane regression and non-linear homography matrix estimation show that our approach is fully deterministic and can effectively process large-scale and highly corrupted data without any special initialization. Under a pure MATLAB implementation and a laptop CPU, our method can successfully determine the maximum consensus set from 1000 input data points (with 70% of them being outliers) at 30 Hz.

INDEX TERMS Consensus set maximization, outlier rejection, robust model fitting, homography matrix estimation, hyper-plane regression, computer vision, system identification.

I. INTRODUCTION

Consensus set maximization (CSM) is a fundamental criterion for robust model fitting problems. In general, the consensus set represents a group of data that support a common model. The CSM problem must be solved for most applications that require performing robust model fitting. One representative example is homography matrix estimation, which is the most common component of vision-based localization and is widely used in robotics navigation and augmented reality (AR). For these real-time applications, there is still no algorithm that can deterministically produce an accurate estimation from large-scale and highly corrupted data within a limited amount of running time.

The most common approach to solving CSM is to perform a hypothesize-and-verify paradigm. RANSAC (Random Sample Consensus) [1] is a classical method within this framework. Its main operation is to hypothesize model parameters by fitting randomly selected minimal subsets of the data and verify these parameters by counting the number of data items that are satisfied by this model. After repeating this process many times, RANSAC will return the model that is supported by the largest consensus set. The most significant feature of RANSAC is that the probability of its solution being optimal can be guaranteed by the number of iterations; the probability of obtaining the optimal solution increases with an increasing number of iterations. However, the running time of RANSAC tends to be long because the quality of its solution cannot be guaranteed with a limited number of iterations. Several methods based on RANSAC have been proposed to reduce the running time. PROSAC [2] can reduce the number of iterations by utilizing the prior knowledge of the order of the probabilities that each datum is an inlier. However, PROSAC performs similarly to RANSAC when these priors are incorrect or difficult to estimate.

Another vital paradigm employs optimization algorithms, such as Norm-optimizers [3], [4] and M-estimators [5]–[7]. IRLS [8] a widely used algorithm for statistical cost optimization. One significant advantage of IRLS is its low computational complexity, as the weighted least squares (LSQ) can be solved efficiently and the robust distance functions are typically differentiable. However, the quality of an IRLS result is dependent on the selection of the robust distance function.
For computer vision applications, when selecting a good distance function, it is difficult to satisfy efficiency and optimality simultaneously [9]. References [10]–[13] focus on the optimality of solutions. To our knowledge, these methods inevitably use an exhaustive search to achieve the global optimum, which is not suitable for large-scale input problems because the computational complexity is exponential. Most recently, a deterministic and locally convergent algorithm was established for iteratively solving linear programming problems [9]. However, this algorithm relies on a good initialization and requires many iterations.

In this paper, we propose a new fast and deterministic algorithm to solve the CSM problem approximately. To derive this algorithm, we first define the general form of the original CSM problem (see (1)). Then, we introduce its relaxed problem (see (3)). Finally, we claim that solving these DPs is equivalent to solving (5), and we propose a new efficient algorithm to approximate (5). In summary, we reformulate and relax the consensus maximization as a sequence of DPs. Second, we will eventually obtain a local convergent solution by updating \( \theta \) and \( u \) until \( \|P\|_1 \) cannot be decreased. We summarize these steps in Algorithm 2. To solve (3), we must solve a sequence of problems (4) that have different consensus set sizes \( k \) and select the best one. Algorithm 1 summarizes the overall process for addressing (3). The original consensus maximization problem is identical to (2), and (3) is the relaxed version of (2). While solving each DP (except for

\[
\max_{\theta \cdot u} \|u\|_1 \\
\text{s.t. } \|P\|_1 \leq \varepsilon, \\
P_i = \rho(f(x_i, \theta), y_i) \cdot u_i
\]

Considering that the optimal value of (3) can only be integer and is possible only in the region \([0, N]\), we first consider the decision problem (DP) that is related to (3), which is defined as:

\[
\text{Given } (x_i, y_i)_{i=1}^N \text{ do there exist } u, \theta \\
\text{s.t. } \|u\|_1 = k, \\
\|P\|_1 \leq k \cdot \varepsilon, \\
P_i = \rho(f(x_i, \theta), y_i) \cdot u_i
\]  

Where \( k \) is the size of the consensus set. Obviously, if we can efficiently solve (4), then (3) can be easily solved by a one-dimensional searching for \( k \).

### C. ALTERNATIVE FITTING ALGORITHM

In this section, we focus on how to solve (4) efficiently. It is easy to select \( k \) items that satisfy \( \|u\|_1 = k \) automatically, but these items might not satisfy \( \|P\|_1 \leq k \cdot \varepsilon \) at the same time. Thus, solving (4) can be transformed into an optimization problem, which is to find values of \( \theta \) and \( u \) that can minimize \( \|P\|_1 \).

\[
\min_{\theta \cdot u} \|P\|_1 \\
\text{s.t. } \|u\|_1 = k, \\
P_i = \rho(f(x_i, \theta), y_i) \cdot u_i
\]

According to the definition of (5), we claim that the original DP (4) is feasible if and only if the optimal value of (5) is not larger than \( k \cdot \varepsilon \). Obviously, this condition is a sufficient condition. We provide a short verification that this condition is also necessary. Here, (4) is feasible means that there exist some \( \theta \) and \( u \) such that \( \|P\|_1 \leq k \cdot \varepsilon \). The optimal value of (5) must be equal to or less than \( k \cdot \varepsilon \) because the optimal solution is not worse than arbitrary solutions.

Observing that if we fix the model parameters \( \theta \), the optimal label variable \( u \) is to set the \( k \) items that have the smallest fitting error to 1. This operation is extremely efficient because the k-smallest items in an array can be found in \( O(N) \) time. If we fix the label variable \( u \), we can also efficiently obtain the optimal \( \theta \) by a least square approach. Alternatively, we will eventually obtain a local convergent solution by updating \( \theta \) and \( u \) until \( \|P\|_1 \) cannot be decreased. We summarize these steps in Algorithm 2.

### II. PROPOSED APPROACH

#### A. PROBLEM DEFINITION

We denote the consensus set maximization problem as follows: given \( N \) pairs of measurements \((x_i, y_i), i \in \{1, 2, \ldots, N\}\) under the system \( y = f(x, \theta), x \in \mathbb{R}^m, y \in \mathbb{R}^n \), we want to estimate the unknown parameters \( \theta \) that can be supported by the largest consensus set, i.e., the model fitting residual of each item in \( I \) is not larger than inlier threshold \( \varepsilon \).

Formally, we define the problem as:

\[
\max_{\theta} |I| \\
\text{s.t. } I = \{x_i, y_i\} | \rho(f(x_i, \theta), y_i) \leq \varepsilon
\]

Where \( \rho(\cdot, \cdot) \) represents the model transform and fitting residual metric function, respectively.

#### B. PROBLEM REFORMULATION

To make the formulation more straightforward, we introduce an indicator variable \( u \in \{0, 1\}^N \). Here, \( u_i = 1 \) means \((x_i, y_i)\) belongs to the inliers. We reformulate (1) as follows:

\[
\max_{\theta \cdot u} \|u\|_1 \\
\text{s.t. } \|P\|_\infty \leq \varepsilon, \\
P_i = \rho(f(x_i, \theta), y_i) \cdot u_i
\]

To establish an efficient algorithm, we relax the maximum residual constraint to the mean error restriction. This relaxation has very clear physical meaning, which is to require the average fitting error in the consensus set to be smaller than the threshold. A solution of the original CSM problem can also be a feasible solution of this relaxed problem. Formally, we define the relaxed problem as:

\[
\max_{\theta \cdot u} \|u\|_1 \\
\text{s.t. } \|P\|_1 \leq \varepsilon, \\
P_i = \rho(f(x_i, \theta), y_i) \cdot u_i
\]

(2)
the first one), we initialize the $\theta$ from the previous result. The initial $\theta$ for the first DP is identical to the initialization of Algorithm 1. To demonstrate the robustness, we apply LSQ (least square) over all measurement data as the initialization. In certain real-life applications, users can use some domain knowledge obtain a better initialization.

Algorithm 1 Alternative Fitting Algorithm for Solving (3)

| Input: $S = (x_i, y_i)_{i=1}^N$, $\theta_{init}$, $\epsilon$, $\delta$, $\tau_{min}$ |
| Output: $\theta$; |
| 1: Initialize: $C \leftarrow \infty$, $\tau = 1$, $\theta \leftarrow \theta_{init}$ |
| 2: while $\tau \geq \tau_{min}$ do |
| 3: $[\text{Isfeasible}, \theta, C] = \text{CheckFeasible}(S, \theta, N \cdot \tau, \epsilon, C)$ |
| 4: if Isfeasible = true then |
| 5: break |
| 6: else |
| 7: $\tau = \tau - \delta$ |
| 8: end if |
| 9: end while |
| 10: return $\theta$ |

III. EVALUATION OF EXPERIMENTS

In this work, our experiments are mainly focused on two types of model fitting problems. The first type of model fitting is hyper-plane estimation, in which the model function is $y = \theta^T \cdot \frac{x}{1}$, where $x \in \mathbb{R}^m$, $\theta \in \mathbb{R}^{m+1}$, $y \in \mathbb{R}$, and the residual metric is $\rho = |\tilde{\theta}^T \cdot \frac{x_i}{1} - y_i|$. This problem can be efficiently solved by least squares if no outliers exist. The second type of model fitting is to estimate the homography matrix [14]. More formally, we define the location of a key point in reference view as $x = \left(u \ v \right)^T$, and the corresponding point in the moving view is defined as $y = \left(u' \ v' \right)^T$. If these key points are projected from a planar surface in 3D world, then they satisfy

$$\lambda \left(\begin{array}{c} y \\ 1 \end{array}\right) = \theta \cdot \left(\begin{array}{c} x \\ 1 \end{array}\right)$$

where $\lambda \in \mathbb{R}$ and

$$\theta = \begin{pmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \\ \theta_{31} & \theta_{32} & \theta_{33} \end{pmatrix}.$$ 

Although (6) appears to be a linear system, it actually belongs to a non-linear transformation, which can be easily verified by writing the expanded formula as

$$u' = \frac{\theta_{11}u + \theta_{12}v + \theta_{13}}{\theta_{31}u + \theta_{32}v + \theta_{33}},$$

$$v' = \frac{\theta_{21}u + \theta_{22}v + \theta_{23}}{\theta_{31}u + \theta_{32}v + \theta_{33}}$$

The homography matrix estimation problem is to replace $f(\cdot, \cdot)$ and $\rho(\cdot, \cdot)$ in (1) with $y_i = \theta \left(\begin{array}{c} x_i \\ 1 \end{array}\right)$ and $|y_i - \tilde{y}_i|_2$, respectively, where $\left(\begin{array}{c} \tilde{y}_i \\ 1 \end{array}\right) = \tilde{\theta} \left(\begin{array}{c} x_i \\ 1 \end{array}\right)$ respectively.

For every experiment, we use the least square solution over all input data to initialize Algorithm 2. We implemented our algorithms under MATLAB R2017b, and the hardware platform was a laptop computer with Intel Core i7-7700HQ CPU of 2.8 GHz and 32 GB of DDR4 RAM. All experiments were executed on this platform. For each result shown in this paper, we set the internal parameters of Algorithm 1 to $\delta = 0.05$ and $\tau_{min} = 0.1$. All internal parameters needed in [9] were kept unchanged. Note that although we mainly focus on solving (3), we still use the $l_\infty$-norm metric (defined in (2)) to justify whether a datum can be classified into the consensus set.

Algorithm 2 Check Feasible Algorithm for Solving (5)

| Input: $(x_i, y_i)_{i=1}^N$, $\theta_{init}$, $k$, $\epsilon$, $C_{init}$ |
| Output: $\text{Isfeasible, } \theta, C$ |
| 1: Initialize: Isfeasible $\leftarrow$ false, $\tilde{\theta} \leftarrow \theta_{init}$, $\theta \leftarrow \theta_{init}$, $\tilde{C} \leftarrow C_{init}$, $C \leftarrow C_{init}$ |
| 2: while true do |
| 3: $r_i \leftarrow \rho(f(x_i, \tilde{\theta}), y_i), \forall i \in \{1, 2, \ldots, N\}$ |
| 4: $u_i \leftarrow \begin{cases} 1 & k - \text{th largest item of } r \\ 0 & \text{Otherwise} \end{cases}, \forall i \in \{1, 2, \ldots, N\}$ |
| 5: $\tilde{\theta} \leftarrow \text{ModelFitting}((x_i, y_i)_{i=1}^N, \forall i \in \{u_i = 1\})$ |
| 6: $P_i \leftarrow \rho(f(x_i, \tilde{\theta}), y_i) \cdot u_i, \forall i \in \{1, 2, \ldots, N\}$ |
| 7: $\tilde{C} \leftarrow P_1 \quad \text{if } \tilde{C} \leq k \cdot \epsilon$ |
| 8: Isfeasible $\leftarrow$ true |
| 9: break |
| 10: else |
| 11: if $\tilde{C} < C$ then |
| 12: $\theta \leftarrow \tilde{\theta}$, $C \leftarrow \tilde{C}$ |
| 13: else |
| 14: break |
| 15: end if |
| 16: end if |
| 17: end if |
| 18: end while |
| 19: return Isfeasible, $\theta, C$ |

A. HYPER PLANE REGRESSION

In this experiment, we perform an evaluation regarding solving the hyper-plane regression problem defined before. We use synthetic data in which the inliers follow a small-variance Gaussian distribution and the outliers are uniformly distributed over a large interval. We perform independent repeated trials under randomly generated model parameters. By accounting for the size of the consensus set, we compare our method with EP-LSQ [9] (both initialized with LSQ) under two different outlier distributions: $U(0, 10)$ and $U(0, 100)$.
We perform an evaluation under different outliers-ratio with fixed total number of data points fixed at as 1000. The model dimension is 9, and the inlier threshold is $\epsilon = 0.5$ ($\epsilon$ is the variance of Gaussian distribution). For each outliers-ratio, we run 100 independent trials and summarize the maximum, average, and minimum size of the inlier set. The results are shown in Figures 1 and 2. When the outlier ratio is 70%, the running time of our method is less than 26178 milliseconds.
than 31.63 ms. In other words, our method can successfully determine the maximum consensus set at 30 Hz with 70% outliers. As shown in Figure 2, EP-LSQ breaks down when the outliers-ratio is more than 20%. However, when tuning the distribution of outliers into a small interval, EP-LSQ can yield successful results with only 10% inliers. Compared to EP-LSQ (with Gurobi linear programming solvers), our method is less sensitive to outliers and more than 100 times faster.

B. HOMOGRAPHY ESTIMATION

In this experiment, we perform another evaluation regarding solving the homography matrix estimation problem defined above. The data that we used are from the VGG dataset [15]. We first used the MATLAB built-in function detectSURFPoints [16], [17] to extract the image key points. Then, we matched these points according to their SURF features. After obtaining the correspondences, we treated them as input for evaluating the algorithm. In each comparison, we use LSQ to initialize each algorithm, and set the inlier threshold \( \epsilon \) to 4 pixels. Because the VGG dataset has 6 images for each scene and provides a reference homography between the first image to five other images, we compared three homographies (ours, [15] and EP-LSQ) and summarize the size of their consensus sets in TABLE 1. In Figure 3, we provide several intuitive examples to illustrate the performance of our method, where the green lines denote correct matches. Our method has considerable advantages in terms of both robustness and running time.
In this paper, we present a new fast and deterministic method to approximately solve the CSM. We first reformulate it as maximizing the $l_1$-norm over the discrete label variable $u$. Then, we relax the original maximum fitting residual constraint to the average error bounded constraint, which can not only simplify the problem but also have an explicit physical meaning. Finally, we approximately solve the relaxed problem by checking the feasibility over its decision problems. Experiments on fitting linear hyper-planes and non-linear homographies illustrate that our method can efficiently handle large-scale input data and effectively address highly corrupted data (the outliers-ratio can be up to 80%).

REFERENCES


ZIRAN XING received the B.S. degree from the North University of China, Taiyuan, China, in 2013. He is currently pursuing the Ph.D. degree with the Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, and ShanghaiTech University, Shanghai, China. His research interests are 3-D vision, SLAM, and robotics.

ZHIRU SHI received the Ph.D. degree from the Centre for Vision, Speech and Signal Processing, University of Surrey, U.K., in 2014. He was a Post-Doctoral Researcher with the Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University. He is currently an Assistant Researcher with the School of Information Science and Technology, ShanghaiTech University, China. His current research interests include 3-D video coding, video quality assessment, high-performance video transcoding systems, and computer vision.