3FO: The Three-Frame-Only Approach for Fast and Accurate Monocular SLAM Initialization

PENG ZHANG AND WENFEN LIU
School of Computer and Information Security, Guilin University of Electronic Science and Technology, Guilin 541004, China
Corresponding author: Wenfen Liu (liuwenfen@guet.edu.cn)

This work was supported by the National Natural Science Foundation of under Grant 61862011.

ABSTRACT The monocular simultaneous localization and mapping (SLAM) system is one of the most important among all visual or visual-inertial SLAM (VSLAM or VISLAM) systems due to its low cost, easy calibration and identification. Initialization is always crucial to bootstrap the monocular SLAM system. With the rapid growth of some quick-start required SLAM applications, e.g., augmented reality (AR) and unmanned aerial vehicles (UAVs), devising faster and more accurate initialization has become a central problem. Traditional initialization uses the first two frames to create landmarks and then computes camera poses for the subsequential frames, using a 3D-to-2D perspective-n-points (PnP) mechanism. In this paper, we propose a novel three-frame-only (3FO) initialization approach for the monocular SLAM system, which consists of two steps. In the first step, we use the first two frames to preinitialize poses and landmarks, and in the second step, we use the second and third frames to improve the preinitialization by using the scale consistency of the landmarks generated in the first step to filter out outliers and using inliers to generate more robust landmarks. In both steps, we use the pretrained multilayer perceptron (MLP) combined with homotopy continuation to solve the essential matrices. Finally, we perform a global bundle adjustment (BA) to refine the three camera poses and all the created landmarks. The proposed 3FO initialization approach is evaluated experimentally on the EuRoC benchmark data set with the initialization time and trajectory metrics. The results show that, compared to the traditional ORB-SLAM2 initialization, the 3FO approach reduces the initialization time by 5 times and improves the accuracy by 36.7% on average.

INDEX TERMS SLAM, multilayer perceptron (MLP), homotopy continuation, monocular, initialization.

I. INTRODUCTION Simultaneous Localization and Mapping (SLAM) has been widely studied since it was first introduced in 1986 by R.Smith [1], and it has become the main technology in robotics to help the robot incrementally map the unknown environment and localize itself. A visual SLAM framework typically uses a visual sensor, (see, e.g., PTAM [2], LSD-SLAM [3], SVO [4], ORB-SLAM2 [5], DSO [6]) or a visual inertial sensor (see, e.g., VINS-MONO [7], ORB-SLAM3 [8]). The monocular camera, for its simple structure, low cost, and easy calibration and identification, has been widely applied as an important sensor in visual or visual inertial systems, with the ability to reconstruct the environment and concurrently estimate the camera poses from a single input video.

Initialization is crucial to bootstrap a monocular system. During the initialization, camera poses for initialization and 3D landmarks are built for the subsequent tracking and mapping process. Good initialization can make tracking more steady and create more accurate mapping.

The general method of initialization for monocular SLAM systems is to first solve the relative pose of the first two keyframes by decomposing the fundamental or homography matrices, set the first frame coordinate to be the world coordinate, then triangulate the feature points to form the initial map, and finally apply a 3D-to-2D method Perspective-n-Points (PnP) [9] to estimate other frame poses. This process can be followed by incremental and local structure-from-motion (SfM) [10], [11].
However, the aforementioned initialization has some drawbacks. First, a sufficient parallax is needed for triangulation accuracy. Second, incorrect matching of feature points will significantly affect the accuracy of pose calculation, especially in an environment with the same texture. Additionally, traditional ways of solving the fundamental or homography matrices will produce many spurious solutions, and the correct relative pose can be screened out through a chirality test. To overcome these drawbacks, the initialization either needs someone to move the device in a certain way or spends tremendous time finding the best solution, but these two conditions cannot be met by most real-time SLAM applications, e.g., AR.

To address the above problems, we propose a faster and more accurate initialization approach, which only uses three frames and is thus called the 3FO initialization approach. Our 3FO approach consists of two steps. In the first step, we use the first two frames to preinitialize poses and landmarks, and in the second step, we use the second and third frames to improve the preinitialization by using the scale consistency of the landmarks generated in the first step to filter out outliers, and using inliers to generate more robust landmarks to form the initial map for subsequent tracking and mapping. Finally, we perform a global bundle adjustment (BA) to refine the three camera poses and all the created landmarks. In both steps, we apply the pretrained multilayer perceptron (MLP) combined with homotopy continuation [12] to replace the traditional method of solving essential matrices. The input of the MLP in our 3FO approach is five-feature correspondences, the same as that in the five-point method [13], but the outputs from the MLP are more convenient to process compared to the five-point method. By using this pretrained MLP, we can perform a more efficient random sample consensus (RANSAC) and maintain high accuracy.

The core idea of our method is as follows. With the pretrained MLP combined with homotopy continuation, we can calculate the relative pose in a shorter time. Then, we use the prior knowledge of the solution to perform RANSAC more efficiently to obtain better 2D information, thereby reducing the complexity of processing 3D information.

The rest of the paper is structured as follows. Section II reviews the related literature. Section III illustrates the 3FO initialization approach. Section IV delineates our 3FO approach in a rigorous manner. Section V demonstrates some experimental results. Section VI concludes the paper and shows some further work results using the same idea.

**II. RELATED WORK**

Monocular SLAM has been widely studied in recent years, and can be classified into two categories: optimization-based methods and filtering-based methods. Optimization-based methods have higher computational complexity than filtering-based methods due to their iterative nature. Filtering-based methods can run faster because they marginalize old states to update new ones recursively. However, both of these monocular SLAM systems rely heavily on the initial values. Thus, a sophisticated initialization strategy with sufficient speed and accuracy is essential for a stable monocular SLAM system.

According to different scene assumptions, different SLAM frameworks adopt different strategies. SVO [4] supposes the scene to be a plane and uses homography to express the translation. The camera poses are evaluated by decomposing the homography matrix by using SVD [14]. For a more general scene, VINS-MONO first recovers the camera poses by decomposing the fundamental matrix; then, a triangulation and Perspective-n-Point (PnP) method is used to recover all the camera poses and landmarks in the sliding window. This method uses all ten frames in the sliding window to initialize and needs the first pair images to have enough parallax. ORB-SLAM2 decomposes both the homography and the fundamental matrices simultaneously and chooses the better one by conducting a series of rigorous validations. Thus, this process takes a long time to deal with different hypotheses for complex scenes during initialization.

From all the above strategies, we can see that, in general, the two main parts of the monocular initialization process are the visual localization (i.e., to estimate its camera poses) and the 3D structure recovery. The accuracy of visual localization algorithms and the correctness of feature point correspondences can significantly influence the quality of the 3D structure recovery.

Visual localization through the relative pose is always the first step in monocular initialization. The relative pose can be encoded by an essential matrix in a more general way. Point-based methods use feature points to construct an essential matrix and decompose it for relative pose estimation. For example, [13] imposes additional constraints on the five-point method to solve the essential matrix, but it needs to validate solutions through some geometric constraints. With the rise of neural networks, [15], [16], and [17] calculate relative poses by regressing the essential matrix from a convolutional neural network (CNN). In this way, directly decomposing the essential matrix to a relative pose is ambiguous, although the ambiguity can be handled in a RANSAC loop. [18] and [19] directly predict the relative pose rather than the essential matrix, but the main disadvantage of their methods is the scene-dependence. Moreover, the above CNN-based relative pose estimation is difficult to run in real time.

Additionally the deep-learned feature points and matching strategies have been introduced into the SLAM system to increase the robustness of feature point matching. [20] is the first to use the scale-invariant features transform (SIFT) [21] to train a neural network to learn the feature transform. [22] proposes a learned detector that is repeatable and reliable. [23] introduces homographic adaptation to train a network. Based on the aforementioned results, [24] proposes a graph neural network (GNN) to match feature points with learned detectors and descriptors by performing a self/cross attention mechanism. Although these methods achieve remarkable results, using a heavy network for extracting and matching feature points is too luxurious for an
SLAM system because these neural networks will influence their real-time performance. Another general way to improve the quality of feature correspondences is to use RANSAC. To improve the efficiency and accuracy of RANSAC, [25] filters out the outliers of the data set by assuming that the distance between the inliers is smaller than that between outliers. [26] uses a matching score scheme to sort the samples, and validating the samples from the highest score to the lowest can improve the speed of RANSAC. Similar to [25], [27] considers the in-class points to be closer, uses prior knowledge to group the points in the data set based on similarity of optical flow-based clustering or image segmentation, etc., and starts sampling with the largest clustering. [12] proposes a “pick & solve” approach (which we call Hruby’s method) to filter out outliers by using a simple MLP to predict whether the solution calculated by a sample is real or spurious, so their approach skips the validation step for spurious solutions in the RANSAC framework and thus greatly improves the efficiency of RANSAC. For an SLAM system or SfM task, the calculation of the essential matrix can be reduced to a hard minimal problem, which can be efficiently solved by Hruby’s method.

III. SYSTEM ILLUSTRATION

We illustrate our 3FO initialization approach in Figure 1. The calculation of camera poses in our method mainly depends on the MLP combined with homotopy continuation, which takes five-feature correspondences as input and returns ten depths for each feature point as outputs. Then, we split the whole initialization process into two steps. In the first step, the first and second frames come in, and then we perform a quick preinitialization by using the poses from MLP and triangulate all inliers to be landmarks. In this step, we use MLP to choose the optimal set of feature points, and then we directly triangulate all feature points without validation. In the second step, the third frame comes in, and we put it together with the second frame into the MLP to calculate the relative pose and obtain ten depths for each feature point as outputs. In this step, the five-feature points in the second frame, which are used to calculate the relative pose of the second and third frames, have their corresponding landmarks created in the first step, and we use these five corresponding landmark coordinates to compare with the second-frame output of MLP, i.e., the five depths of the feature points in the second frame, to recover the scale.
and validate the correctness of the solution. We put all the validated solutions into the RANSAC framework to choose the best solution. Then, we triangulate more landmarks from the second and third frames to enhance the robustness of initialization. Finally, a global BA is performed to refine the three camera poses and all the landmarks.

### IV. METHODOLOGY

We first define some notations and the coordinate frames that we use throughout this paper:

- $c_i$: image frame where $i \in \{1, 2, 3\}$ is the frame index.
- $R_{c_i c_j}$: rotation matrix that transforms the coordinates from $c_j$ to $c_i$.
- $t_{c_i c_j}$ is the translation from $c_i$ to $c_j$, represented in the coordinate system of $c_i$.
- $t^\wedge$ is the skew-symmetric matrix of vector $t$.
- $K$ is the intrinsic matrix of the pinhole camera.

#### A. ESSENTIAL MATRIX ESTIMATING AND LEARNING-BASED ALGORITHM

In this subsection, we introduce two different ways to solve the essential matrix and then compare them. The first is introduced by Nister [13], and the second is introduced by Hruby [12].

1) **Essential Matrix Estimating:** We first illustrated here the essential matrix formulation and the general way to solve it introduced by Nister [13]. The essential matrix can be estimated by the feature correspondences between any two frames. The correspondences to form the epipolar constraint are defined as follows:

$$p_i^T K^T t^\wedge_{c_i c_j} R_{c_i c_j} K^{-1} p_j = 0$$

and supposing we know the intrinsic matrix, the equation can be formed as:

$$x_i^T t^\wedge_{c_i c_j} R_{c_i c_j} x_j = 0$$

where $x_1 = (u_1, v_1, 1)^T$ and $x_2 = (u_2, v_2, 1)^T$ are the normalized plane coordinates for $p_1$ and $p_2$ respectively. In the case where we know the camera intrinsic parameters, solving Eqn. (2) reduces the parameters we need to estimate, and the essential matrix is defined as:

$$E = t^\wedge_{c_i c_j} R_{c_i c_j}$$

Since there are three degrees of freedom in each of the translations and the rotations, there are six degrees of freedom in total. However, due to scale equivalence, there are only five degrees of freedom in fact. In other words, we need to use at least five feature correspondences to solve the matrix, and the traditional method is the five-point method introduced by Nister [13]. In Nister’s way, the equation formed by the five feature correspondences is as follows:

$$E = xX + yY + zZ + W$$

where $x, y, z$ are the unknown parameters, and $X, Y, Z, W$ are the known $3 \times 3$ matrices. Then, the three unknown parameters can be solved by putting the additional constraint (5) from the intrinsic nature of the essential matrix on (4).

$$EE^T - \frac{1}{2} \text{trace}(EE^T)E = 0$$

More generally, when we use the fewest inputs to solve the problem, it can be treated as a hard minimal problem [28]. Inherently, solving a hard minimal problem will generate many spurious solutions. For the classic five-point problem, applying Nister’s method will create 10 solutions, wherein only one is the real solution. Thus, we need to set constraints to find the real solution. Moreover, we do not know whether the feature correspondences include outliers. Therefore, we apply the RANSAC framework to refine the real solutions and choose the optimal solution. The aforementioned process greatly influences the real-time performance of the algorithm.

2) **Learning-based algorithm:** Hruby [12] introduced a (deep) learning method as an alternative approach to solving the essential matrix. In such a learning method, we first form the five-point problem with five 3D landmarks and their corresponding projections in any two frames, as illustrated in Figure 2.

![FIGURE 2. $C_i$ and $C_j$ are two cameras. Landmarks $X_m$ and $X_k$ were projected to two cameras with their corresponding points $x_{m,i}$, $x_{k,i}$, $x_{m,j}$, $x_{k,j}$.

Whichever frames are observing the pair of 3D landmarks, the distance between them should be identical. Therefore, the equation can be formed as:

$$\|\lambda_{k,i}v_{k,i} - \lambda_{m,i}v_{m,i}\|_2 = \|\lambda_{k,j}v_{k,j} - \lambda_{m,j}v_{m,j}\|_2$$

where $k$ and $m$ are two different landmarks, $\lambda_{i,j}$ is the depth for the feature point $x_{i,j}$, and $v = [x : 1]$ is the corresponding coordinates of $x$ in the two camera normalized planes in the homogeneous form. In this formulation, we can directly calculate the depths of the five feature points.

To avoid spurious solutions, we use the neural network to generate the real solutions from the prior information learned from the data set. The general way to circumvent the complexity of real solution generation is to
change a regression problem into a classification problem. We first extract certain pairs that contain 10 feature point coordinates and their corresponding depths from the data set. In this paper, we call these pairs anchors. We use the neural network to find the anchor that is the nearest real solution to the input problem. Once we find the nearest anchor, we choose this anchor to be the starting point for finding the real solution by adopting homotopy continuation [29].

There are several advantages for incorporating the learning approach, instead of the Nister’s, with our 3FO initialization approach.

First, the nearest anchor and the data, which are used to train the MLP, are both extracted from the data set, whose 3D structure is known, and thus are suitable for serving as the starting point to solve the problem. As a result, we can greatly reduce the number of spurious solutions and thus improve the probability of finding the real solution.

Second, if the input of the feature correspondences includes outliers, the MLP can return a nonanchor solution, and thus, we do not need to put the nonanchor into the RANSAC framework. This can make RANSAC more efficient and increase the accuracy of the relative pose compared to the RANSAC framework with Nister’s method.

Third, the real solution is tracked by the anchors extracted from the data set. Therefore, we can directly obtain the depths of the feature points in the 3D structure, where the depths are in a convenient format for the validation step of our 3FO initialization process. Last, the MLP is lightweight and thus fast to train, and it is easy to prepare the training data set. Therefore, we can customize the training data set and anchors according to the application scenario and thus improve the performance of finding the real solution. We train our MLP on the EuRoC data set and compare the performances under the EuRoC data set and Eth3D data set in Table 1.

![Figure 3: Structure of learning-based algorithm.](image)

### Table 1. Comparison of success rate tracked by MLP.

| Eth3D/EuroC | Numbe of anchors |
|-------------|------------------|
| Ransac Sampling times | 8 | 15 | 32 | 70 | 124 |
| 5 | 0.5/4.3 | 1.1/7.2 | 6.7/14.1 | 6.4/14.6 | 8.1/16.2 |
| 10 | 2.1/5.4 | 5.1/11.3 | 13.2/23.6 | 13.9/23.2 | 15.4/26.3 |
| 30 | 3/6.8 | 4.7/10.2 | 12.1/24.4 | 12.2/24.8 | 16.5/27.8 |
| 50 | 2.7/4.2 | 2.9/6.8 | 7.5/20.9 | 7.4/20.1 | 8.5/22.3 |
| 100 | 2.9/4.5 | 2.7/7.5 | 6.3/17.1 | 6.5/17.4 | 7.1/18.4 |
where Eth3d/EuRoC[%] indicates the probability of finding the real solution (or equivalently, the success ratio) under the EuRoC data set versus that under the Eth3D data set. We can see that the success ratio under EuRoC is twice as good as that under Eth3D.

**B. THE 3FO INITIALIZATION APPROACH**

In our 3FO method, the initialization process can be split into two steps. In the first step, when the first and second frames come in, we use the learning model to calculate the relative pose of these two frames. We use RANSAC to account for noisy relative pose predictions. Finally, we obtain the optimal solution, which can be decomposed into a rotation $R_{c1c2}$ and a translation $t_{c1c2}$. Such an optimal solution has the largest number of inliers set $\Omega_{12}$, where inliers are feature correspondences within the Sampson error threshold defined by the essential matrix:

$$
||e||^2 = \frac{(x'Ex)^2}{(Ex)_1^2 + (Ex)_2^2 + (E^T x)_1^2 + (E^T x)_2^2}
$$

(7)

Then, we triangulate all the feature correspondences by constructing Eqn. (8) for both camera poses and solving them with SVD [14].

$$
\begin{bmatrix}
x \\
y \\
z
\end{bmatrix} = 
\begin{bmatrix}
R & t \\
3x3 & 3x1
\end{bmatrix}
\begin{bmatrix}
X \\
Y \\
Z \\
1
\end{bmatrix}
$$

(8)

In the second step, when the third frame comes in, we use the feature points in $\Omega_{12}$ and their correspondences in the third frame as inputs of the learning model to calculate the rotation and the translation of the second and the third frames, and we denote them by $R_{c2c3}$ and $t_{c2c3}$. Note here that we only use the 2D information of images, so the unified translation should be $st_{c2c3}$, where $s$ is the unknown parameter.

To be more specific, in this step, the input of the MLP is five feature correspondences (five points in the second frame and five corresponding points in the third frame) in $\Omega_{12}$. We denote the five points in the second frame by $x_0, x_1, x_2, x_3,$ and $x_4$. The corresponding 3D landmarks of these five points, which have been triangulated in the first step, are denoted by $X_0, X_1, X_2, X_3,$ and $X_4$. The output of the MLP is ten depths of the five feature correspondences (five depths for the second frame and five for the third frame), whose scale differs from the scale in the first step. We denote the five depths of the second frame by $Sol_0[0], Sol_1[1], Sol_2[2], Sol_3[3],$ and $Sol_4[4]$.

Because the coordinates in the first and the second steps only differ by a scale factor, ideally, for $i \in \{0, 1, 2, 3, 4\}$, the ratio of $X_{i}[2]$ and $Sol_{i}[4]$ should be the same. Hence we derive the $s$ from Eqn. (9).

$$
\begin{align*}
    s &= \text{mean} \left[ \frac{X_{0}[2]}{Sol_{0}[0]}, \frac{X_{1}[2]}{Sol_{1}[1]}, \frac{X_{2}[2]}{Sol_{2}[2]}, \frac{X_{3}[2]}{Sol_{3}[3]}, \frac{X_{4}[2]}{Sol_{4}[4]} \right] \\
    \end{align*}
$$

(9)

In the calculation of $s$, we set a deviation threshold for $\frac{X_{i}[2]}{Sol_{i}[0]}$, $i \in \{0, 1, 2, 3, 4\}$. If there exists a ratio that is an obvious outlier, we call it an unreal solution. We then put all the real solutions into RANSAC and find the maximum inliers set $\Omega_{23}$ from all the feature correspondences in the second and the third frames. Next, we triangulate the new inliers, which are contained in $\Omega_{23}$ but not in $\Omega_{12}$, i.e., $\Omega_{23} - \Omega_{12}$, to create more robust landmarks. After all the above steps, a global BA is used to refine all three camera poses and landmarks. The whole process is illustrated in Figure 4.

**FIGURE 4. Flowchart of 3FO initialization process.**

**V. EXPERIMENTAL RESULTS AND ANALYSIS**

**A. EXPERIMENT PREREQUISITES AND DATA SETS**

In this section, we validate both the efficiency and the accuracy of our 3FO initialization approach through experiments, and we study how the number of anchors and the number of RANSAC samples affect the efficiency and accuracy.

The EuRoC data set [30], as a well-accepted benchmark for evaluating the performance of SLAM, contains 11 stereo and IMU sequences from a micro aerial vehicle (MAV) with their ground truth poses. The images are recorded by two global shutter monochrome cameras at 20fps with a resolution of $752 \times 480$. The experiment is built on 5 sequences from the EuRoC data set, one sequence (i.e., MH-02-easy sequence) for extracting anchors and training the MLP and the rest (i.e., MH-01-easy, MH-03-medium, MH-04-difficult, and MH-05-difficult) for evaluating both the initialization efficiency and accuracy.

To launch our 3FO approach, we first use COLMAP’s [31] point triangulator to generate the 3D model of the MH-02-easy sequence and extract the best anchors from the generated 3D model by projecting 3D landmarks to cameras.
Then, we generate training data for training MLP. We replace the default initialization of ORB-SLAM2 with our 3FO initialization and compare this newly acquired system with the original ORB-SLAM2 to quantify the improvements of initialization on the remaining 4 sequences. All the experiments are conducted in Ubuntu version 18.04 operating system equipped with Intel® Core™ i7-7700 CPU @ 3.60GHz×8 CPU, NVIDIA GeForce GTX 1060 5GB/PCIe/SSE2 Graphics, and 16GB Memory.

To better understand the performance of our 3FO initialization under different MLP and RANSAC parameters, i.e., anchor number and RANSAC sample size, we introduce the following metrics to evaluate the overall performance of the SLAM system.

- **The absolute trajectory error (ATE)**, which reflects the consistency of the trajectory, for initialization frames is defined in (10), with $m$ being the number of frames used to initialize (3 frames for our 3FO initialization and 2 frames for the original ORB-SLAM2).

\[
\varepsilon_{ATE} = \sqrt{\frac{1}{m} \sum_{i=1}^{m} \left| p[i] - p_{GT}[i] \right|^2} \tag{10}
\]

The smaller the ATE is, the closer the estimated trajectory and the ground truth trajectory. Thus, a smaller ATE implies a more accurate initialization.

- **Reprojection error** reflects the quality of the 3D landmarks. This quality is also important for initialization because we need it to localize the camera for the subsequent frame. The ATE metric can only measure the accuracy for estimated poses, and thus, we need this metric. The lower the reprojection error is, the greater the initialization accuracy.

- **The root mean square error (RMSE) of the ATE** of the whole trajectory reflects the accuracy and robustness of the system better than the mean and median values of the ATE and is used to evaluate the positional accuracy and system stability of the whole trajectory. The smaller the RMSE is, the more accurate the estimated trajectory poses.

- **Time** is the total run-time of calculating the relative pose and the triangulation.

- **Track length** reflects the finishing time of initialization. The longer the track length is, the earlier the finishing time.

### B. EXPERIMENTAL RESULTS

Tables 2 to 5 show the quantitative comparison results on the MH-01-easy, MH-03-medium, MH-04-difficult, and MH-05-difficult sequences, respectively. In each table, the first column presents the different numbers of RANSAC samples, and the first row presents the different number of anchors. The values of time for calculating the relative pose and triangulation, RMSE, and track length are presented under different sets of anchors/RANSAC samples.

From the four tables, we can see that with the increase in the anchor number and the RANSAC sampling times, the RMSE of the four trajectories decreases, but the time for initialization increases. Compared with the original ORB-SLAM2 initialization approach, which requires 200 RANSAC sampling times to find the best relative pose solution, 3FO initialization only needs 30 RANSAC sampling times to achieve higher accuracy. In Table 6, we compare the number of landmarks for the initial map, as well as their reprojection error, resulting from the original ORB-SLAM2 with that resulting from our 3FO approach. The difference in feature correspondences for triangulating landmarks between the two initialization approaches is shown in Figure 5. We can see that our 3FO approach can find better feature correspondences and compute the relative pose more accurately in less time.
Moreover, we can see that in the cases where RANSAC sampling times are under 30, the time is very short, but the accuracy of calculation cannot be guaranteed. In the cases where the sampling times are over 50, the improvement of the system performance from increasing RANSAC sampling times seems to converge, and the decrease of RMSE caused by more sampling times can not compensate for the increase of time.
When comparing different trajectories, according to Table 2, we can see that the MH-01-easy sequence, which contains an obvious up and down movement at the beginning of the trajectory and is in a relatively bright environment, is very friendly to initialization. The accuracy of the two initialization approaches is very close, and the improvement...
of RMSE under (32 anchors/50 ransac samples) is only 17.9% from ORB-SLAM2 (RMSE = 0.0489) to 3FO (RMSE = 0.0401), but the speed improvement, which is 77.4% from ORB-SLAM2 (10072 µs) to 3FO (2273 µs), is significant. This is because the bright condition can provide relatively accurate feature correspondences and the wide range movement can reduce the triangulation error. Therefore, for the MH-01-easy sequence, the advantage of our 3FO approach cannot be fully manifested. We will see later that, as the difficulty of initialization increases, the advantage of our 3FO approach becomes more evident.

For the MH-05-difficult sequence, the trajectory is in a relatively dark environment, and the beginning of the trajectory is only slightly shaky, which leads to error-prone feature point matching and very small parallax for triangulation. Our 3FO initialization can greatly improve the performance of initialization, with accuracy improvement from ORB-SLAM2 (RMSE = 0.1110) to 3FO (RMSE = 0.056) under (32 anchors/50 ransac samples) at 49.5% and speed improvement from ORB-SLAM2 (time = 13769 µs) to 3FO (time = 2356 µs) at 83.4%. Figure 6 shows that our 3FO approach finishes the initialization at Frame 4, whereas the original ORB-SLAM2 finishes at Frame 45. The reason why the improvement is more remarkable for a more difficult sequence is that our 3FO approach is more efficient in filtering out a large portion of 2D outliers that are typically contained by difficult sequences.

Moreover, we use the EVO tool to compare the two trajectories under different initialization approaches, as illustrated in Figure 7. We can see that the ATE at the beginning of the 3FO trajectories is significantly lower than that of ORB-SLAM2, and the RMSE error, mean error, median error and standard deviation (S.D.) are also lower, which can be largely attributed to the better performance of initialization.

From the above experiments, we can see that the reduction in RMSE converges as the anchor number approaches 32. Thus, we next conduct a detailed experiment with a wider range of RANSAC sampling times, given 32 anchors fixed, and we compare both the ATE and the initialization time resulting from ORB-SLAM2 with that resulting from our 3FO approach. As illustrated in Figure 8, our 3FO approach has time superiority across all RANSAC sampling times, and when the sampling times are greater than 30, our 3FO approach beats the original ORB-SLAM2 on ATE.

VI. CONCLUSION AND FURTHER WORK

In this paper, we propose a novel initialization approach, 3FO, for a monocular SLAM system. This approach replaces the traditional method of solving the essential matrix by using MLP combined with homotopy continuation, which can calculate the relative pose more quickly. At the same time, we perform RANSAC twice to filter outliers to obtain better 2D information, which can shorten the time taken for triangulation and improve the accuracy of 3D information.
In both RANSAC processes, we use the prior information of the solution to improve the efficiency of RANSAC. We validate our method on the EuRoC data set. The experimental results show that compared with the traditional ORB-SLAM2 initialization method, the 3FO approach can complete the initialization earlier, while shortening the time for initialization by 5 times and improving the RMSE by 36.7% on average. Considering the strong outlier filtration effect of our approach. Future work will seek to apply our method for large-scene visual localization tasks by leveraging better 2D information to reduce the localization time and improve the accuracy.

REFERENCES

[1] R. Smith, M. Self, and P. Cheeseman, “Estimating uncertain spatial relationships in robotics,” in Autonomous Robot Vehicles. New York, NY, USA: Springer, 1990, pp. 167–193.

[2] G. Klein and D. Murray, “Parallel tracking and mapping for small AR workspaces,” in Proc. 6th IEEE ACM Int. Symp. Mixed Augmented Reality, Nov. 2007, pp. 225–234.

[3] J. Engel, T. Schöps, and D. Cremers, “LSD-SLAM: Large-scale direct monocular SLAM,” in Proc. Eur. Conf. Comput. Vis., Cham, Switzerland: Springer, 2014, pp. 834–849.

[4] C. Forster, M. Pizzoli, and D. Scaramuzza, “SVO: Fast semi-direct monocular visual odometry,” in Proc. IEEE Int. Conf. Robot. Autom. (ICRA), May 2014, pp. 15–22.

[5] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, “ORB-SLAM: A versatile and accurate monocular SLAM system,” IEEE Trans. Robot., vol. 31, no. 5, pp. 1147–1163, Oct. 2015.

[6] X. Gao, R. Wang, N. Demmel, and D. Cremers, “LDSO: Direct sparse odometry with loop closure,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), Oct. 2018, pp. 2198–2204.

[7] T. Qin, P. Li, and S. Shen, “VINS-Mono: A robust and versatile monocular visual-inertial state estimator,” IEEE Trans. Robot., vol. 34, no. 4, pp. 1004–1020, Aug. 2018.

[8] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, “ORB-SLAM3: An accurate open-source library for visual, visual–inertial, and multimap SLAM,” IEEE Trans. Robot., vol. 37, no. 6, pp. 1874–1890, Dec. 2021.

[9] V. Lepetit, F. Moreno-Noguer, and P. Fua, “EPnP: An accurate O(n) solution to the pnp problem,” Int. J. Comput. Vis., vol. 81, no. 2, pp. 155–166, Feb. 2009.

[10] Z. Cui and P. Tan, “Global structure-from-motion by similarity averaging,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 864–872.

[11] O. Ozyesil and A. Singer, “Robust camera location estimation by convex programming,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 2674–2683.

[12] P. Hruby, T. Duff, A. Leykin, and T. Pajdla, “Learning to solve hard minimal problems,” in Proc. IEEE/IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2022, pp. 5532–5542.

[13] D. Nistér, “An efficient solution to the five-point relative pose problem,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 26, no. 6, pp. 756–770, Jun. 2004.

[14] Z. Zhang and A. R. Hanson, “3D reconstruction based on homography mapping,” in Proc. ARPA, 1996, pp. 1007–1012.

[15] V. Balntas, S. Li, and V. Prisacariu, “RelocNet: Continuous metric learning relocalisation using neural nets,” in Proc. Eur. Conf. Comput. Vis. (ECCV), 2018, pp. 751–767.

[16] I. Melekhov, J. Ylioinas, J. Kannala, and E. Rautu, “Relative camera pose estimation using convolutional neural networks,” in Proc. Int. Conf. Adv. Concepts Intell. Vis. Syst. Cham, Switzerland: Springer, 2017, pp. 675–687.

[17] Q. Zhou, T. Sattler, M. Pollefeys, and L. Leal-Taixe, “To learn or not to learn: Visual localization from essential matrices,” in Proc. IEEE Int. Conf. Robot. Autom. (ICRA), May 2020, pp. 3319–3326.

[18] Z. Laskar, I. Melekhov, S. Kalia, and J. Kannala, “Camera relocalization by computing pairwise relative poses using convolutional neural network,” in Proc. IEEE Int. Conf. Comput. Vis. Workshops, Oct. 2017, pp. 929–938.

[19] S. Saha, G. Varma, and C. V. Jawahar, “Improved visual relocalization by discovering anchor points,” 2018, arXiv:1811.04370.

[20] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, “LIFT: Learned invariant feature transform,” in Proc. Eur. Conf. Comput. Vis., Cham, Switzerland: Springer, 2016, pp. 467–483.

[21] A. Witkin, “Scale-space filtering: A new approach to multi-scale description,” in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., vol. 9, Mar. 1984, pp. 150–153.

[22] J. Revaud, C. De Souza, M. Hunenberger, and P. Weinzaepfel, “R2D2: Reliable and repeatable detector and descriptor,” in Proc. Adv. Neural Inf. Process. Syst., vol. 32, 2019, pp. 1–11.

[23] D. DeTone, T. Malisiewicz, and A. Rabinovich, “SuperPoint: Self-supervised interest point detection and description,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2018, pp. 224–236.

[24] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, “SuperGlue: Learning feature matching with graph neural networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 4938–4947.

[25] D. R. Myatt, P. H. S. Torr, S. J. Nasuto, J. M. Bishop, and R. Craddock, “NAPSAC: High noise, high dimensional robust estimation—It’s in the bag,” in Proc. Brit. Mach. Vis. Conf., vol. 2, 2002, p. 3.

[26] O. Chum and J. Matas, “Matching with PROSAC—Progressive sample consensus,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR05), vol. 1, Jun. 2005, pp. 220–226.

[27] K. Ni, H. Jin, and F. Dellaert, “GroupSAC: Efficient consensus in the presence of groupings,” in Proc. IEEE 12th Int. Conf. Comput. Vis., Sep. 2009, pp. 2193–2200.

[28] Z. Kukelova, M. Bujnak, and T. Pajdla, “Automatic recognition of minimal view sets for nonfrontal views,” in Proc. IEEE Int. Conf. Robot. Autom. (ICRA), May 2009, pp. 1421–1446, Jul. 2019.

[29] T. Duff, C. Hill, A. Jensen, K. Lee, A. Leykin, and J. Sommers, “Solving polynomial systems via homotopy continuation and monodromy,”IMA J. Numer. Anal., vol. 39, no. 3, pp. 1421–1446, Jul. 2019.

[30] M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, “The EuRoC micro aerial vehicle datasets,” Int. J. Robot. Res., vol. 35, no. 10, pp. 1157–1163, 2016.

[31] J. L. Schonberger and J. M. Frahm, “Structure-from-motion revisited,” in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 4104–4113.

PENG ZHANG received the B. E. degree in computer science and technology from the Hangzhou University of Electronic Science and Technology, Zhejiang, China, in 2019. He is currently pursuing the master’s degree with the School of Computer and Information Security, Guilin University of Electronic Science and Technology, Guangxi, China. His research interests include slam, 3D reconstruction, computer vision, and computer graphics.

WENFEN LIU was born in Henan, China. She received the M.S. and Ph.D. degrees from The PLA Information Engineering University, Zhengzhou, China, in 1992 and 1999, respectively. She was introduced as a high-level personnel to take a full-time job with the Guilin University of Electronic Science and Technology, in 2017. She has authored or coauthored more than 100 scientific articles. Her research interests include computer vision, machine learning, data security, and privacy protection.