Sparse Visual Localization in GPS-Denied Indoor Environments

Chengyi Zhang
Tsinghua International School
Beijing, China
chriszhang2020905@hotmial.com

***Abstract*—This paper presents a real time feature based sparse visual localization algorithm for indoor environments such as shopping mall. ORB feature is used for feature detection and descriptor computation in this project. The features extracted from ORB are mainly points in corners, and edges. Since this project is used for indoor localization, the dataset for experiment is TUM RGBD dataset. This dataset contains various indoor environments which can fulfill our requirements. The essential and fundamental matrices are solved based on the matching of ORB features and Eight-point algorithm. We show certain advantages of sparse visual localization algorithms in our experiment's results.**

Keywords-Visual SLAM; Localization; ORB feature; Indoor Environment

I. INTRODUCTION

The motivation for this project is to provide a solution based on visual Simultaneous Localization and Mapping (visual SLAM) for indoor navigation. Visual SLAM is now increasingly popular in fields like Visual Reality and Augmented Reality. In indoor environments, lack of GPS signal has been a problem for localization and navigation. For example, in shopping malls, it is difficult to locate our desired shops by just using GPS from mobile phones. However, at the same time, indoor environments make visual SLAM a viable solution because of its consistent lighting condition. Visual SLAM includes image localization and mapping, in which localization is used to estimate the camera trajectory, and mapping is used for reconstructing the 3D environment. Therefore, we consider visual SLAM as a more reliable and accurate solution for indoor navigation.

The reasons for the choice of using sparse visual localization algorithm for this project rather than other solutions like dense visual localization algorithm is due to its limited computing resources, such as cell phones or tablets. Sparse visual localization only requires minimal computing power to make real time localization and navigation possible. Also, sparse visual localization algorithm includes elements like SIFT and ORB features that are reliable for frame to frame matching. Although the environment for this project is set in indoor, we still need to be considering unstable circumstances. With this in mind, we chose to use feature based visual localization algorithm rather than direct visual localization algorithm because it is more robust for illumination variation. This is because direct visual localization algorithm basically assumes that the environment has no illumination changes, which is impractical for environments like shopping mall.

In this project we utilize ORB feature as feature detector and feature descriptors. The matching based on ORB may have outliers. In order to refine these noisy matchings, we also implemented Random Sample Consensus (RANSAC) to exclude outliers from our feature set. Using the eight-point algorithm, we can formulate eight equations that are necessary for solving the fundamental matrix. After computing the fundamental matrix, using the camera intrinsic, we can deduce the camera's rotation and translation by conducting singular value decomposition on the essential matrix. Throughout this process of calculations with fundamental matrix, we assumed that the camera is calibrated so the camera intrinsic parameters are known. This completes our processing of a pair of frames captured by our camera. After this algorithm processes multiple consecutive frames, we can obtain camera trajectory by accumulating camera rotation and transformation.

II. RELATED WORKS FEATURE BASED METHODS

Feature based methods for visual SLAM essentially simplifies the process of approximating and extracting geometric information into two parts. First is feature detection and the second is the calculation of camera position based on the features observed through the previous step.

The first step requires the extraction of features, also known as key points, from the image, which are typically edges or corners. Then we can retract information by processing consecutive frames and calculate the fundamental and essential matrices.

A. Direct Method

Direct visual odometry methods directly works and processes the image through the image intensities, which includes all information in this image to process. Therefore, instead of processing the feature points like the feature-based methods, the direct methods process the pixel directly. Comparing to other visual odometry methods, the disadvantages of direct methods is its lack of robustness for change in illumination.

B. Deep Learning Method

Another conventional method for relocation is through Neural Network. These methods typically use depth images to create some labels which maps camera pixel to world coordinates, which is similar to the feature-based method. Later

978-1-7281-6494-6/19/$31.00 ©2019 IEEE
DOI 10.1109/ITCA49981.2019.00027

these information and labels will be used to train a regression forest which would regress the labels for localization. The advantage of the deep learning method is that versions of this method only trains a neural network, which doesn't require the storage of a map like direct or feature based methods. This means that the storage needed to store a larger scene doesn't grow linearly.

III. Theoretical Approach

A. Background

1) Pinhole Camera Projection Model

Projecting an arbitrary 3D real-world point perceived by the camera consists of two steps. First, a transformation of coordinates will be performed. The points from the world coordinates will be transformed to camera coordinate system.

Then, the image from the camera coordinate system is transformed to the image coordinate system.

In our research, since the camera is moving and it is continuously shooting the picture, the camera coordinate system varies while the world coordinate system for an object is static. The transformation of the camera consists of rotation (R) and translation (t), which together forms the translation matrix. Since rotation in 2D can be represented by the equation

$$R = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \text{(Special Orthogonal group)} \quad [1]$$

rotation in 3D can be defined as

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad [2]$$

Assuming a point P is transformed to point P1 with rotation and translation, its movements could be written in the form $P = R_1 P + t_1$. Suppose then we were to transform P1 to point P2, then

$$P_2 = R_2(R_1 P + t_1) + t_2 \quad [3]$$

This relation is complicated to express in terms of equations, but relatively easy if we express P in terms of homogeneous coordinates, where,

$$\begin{bmatrix} P_1 \\ 1 \end{bmatrix} = \begin{bmatrix} R_1 & t_1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} P \\ 1 \end{bmatrix} \quad [4]$$

which means that we can express P2 with the expression

$$\begin{bmatrix} P_2 \\ 1 \end{bmatrix} = T_1 T_2 \begin{bmatrix} P \\ 1 \end{bmatrix} \quad [5]$$

In this equation the translation matrix

$$T = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \text{ (Special Euclidean Group)} \quad [6]$$

[R, t] transform a 3D point in the world coordinate system (Xw) to the camera coordinate system (Xc). In homogeneous coordinates,

$$X_c = T X_w = \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} X_w \quad [7]$$

After world to camera transformation is conducted, we need to transform the point into 2D image coordinate system. As shown below, our goal is to map the point Xc on to the image plane to the point Xim.

First, we only consider the Z-Y plane first, then the mapping would look like below.

If we apply geometry, we can obtain the Y coordinates for Xim with similar triangle, which

$$y_{xim} = f_y Y / Z \quad [8]$$

Then we can go through the same process with the XZ plane, which gives us. Therefore, in homogeneous coordinates,

$$\tilde{x_{im}} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{f_x X}{Z} + C_x \\ \frac{f_y Y}{Z} + C_y \\ 1 \end{bmatrix} \quad [9]$$

We then have to express this relation between 3D point Xc and Xim in an expression. The expression is in the form[1], where K is also known as the camera intrinsic matrix, which we assumed we already knew in this project. In order to satisfy the expression

$$x_{im} = K x_c \quad K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad [10]$$

In conclusion, the entire process of world to image transformation can be summarized by the equation

$$x_{im} = K[R|t] x_w \text{[1]}.$$

2) Fundamental and Essential Matrix

To track camera rotation and translation, we need to adopt techniques in epipolar geometry. Consider if there are two pinhole camera models that project the same world point. In this model the point X need to pass through the optical center 1 and these points project on to image plane 2. These points connect through one line called the epipolar line. In this case, x and x' forms correspondence. Therefore, what this means is that after we determine the epipolar line, we can find x', which corresponds with x on the epipolar line by conducting a 1-dimension search. We can express this relation with the epipolar constraint. This is when the essential matrix steps in, because essential matrix is what encodes the information about the epipolar lines.

According to the pinhole camera model described in the previous section, we can obtain the equation set

$$\tilde{X^{}} = K[I, 0] \tilde{X_W} \quad [11]$$

$$\tilde{X^{}}' = K'[R, t] X'_w \quad [12]$$

We can express the two 3D world coordinates in the camera coordinate systems.

$$\tilde{X_c^{}} = [I, 0] \tilde{X_W} = \tilde{X_W} \quad [13]$$

$$\tilde{X_c^{}}' = [R, t] \tilde{X_W} = [R, t] \tilde{X_W} \quad [14]$$

Therefore, we reached the relation between and.

$$X_c' = RX_c + t. \quad [15]$$

We will find the essential matrix based on the following operations:

$$t \times X_c' = t \times RX_c + t \times t. \quad [16]$$

$$t \times X_c' = t \times RX_c \quad [17]$$

$$X_c'^T (t \times X_c') = X_c'^T (t \times RX_c). \quad [18]$$

$$X_c'^T [t_\times] RX_c = 0. \quad [19]$$

If we suppose that K and K' are all known, and we set them as identity matrix then we will have the equation

$$x^\sim [t_\times] R x'^{\sim T} = 0. \quad [20]$$

The essential matrix (E) is defined as

$$E = [t_\times] R = t \times R. \quad [21]$$

And in which the epipolar constraint is defined as

$$x'^{\sim T} E x^\sim = 0. \quad [22]$$

When calculating the essential matrix, we assumed that the camera intrinsic matrices, K and K', are all known, and we set them as identity matrix. The calculations would be a bit different if we consider these camera intrinsic matrices differently. We would start with the relation below

$$x^\sim = KX_c \quad [23]$$

$$x^{\sim\prime} = K' X_c'. \quad [24]$$

Thus we can obtain the relation below

$$X_c = x^\sim K^{-1} \quad [25]$$

$$X_c{}' = x^{\sim\prime} K^{-1\prime} \quad [26]$$

Remember that previously we had the relation

$$X_c'^T [t_\times] RX_c = 0 \quad [27]$$

We can substitute Xc and Xc'' into the previous relation. Then we have

$$x^{\sim\prime T} (K^{-1\prime})^T [t_\times] R K^{-1} x^\sim = 0 \quad [28]$$

Then the definition of the fundamental matrix (F) is

$$F = (K^{-1\prime})^T [t_\times] R K^{-1} \quad [29]$$

And equation above can be represented using essential matrix

$$F = (K^{-1\prime})^T E K^{-1} \quad [30]$$

This shows that the essential matrix is a special case of F, because that is when the camera intrinsic matrices are identity matrices.

B. Feature detection and matching

For feature identification, we are using the Oriented FAST and Rotated BRIEF (ORB feature). ORB feature includes two parts, which is FAST, a detector that doesn't require the computation of image gradients and BRIEF.

ORB feature starts by collecting FAST points in the image where FAST takes the parameter of the intensity threshold between the center pixel and the pixels in a circular ring around it. Then, a Harris corner measure is implemented to pick out a number of N key points. However, FAST does not have an orientation operator. The response to this problem from ORB feature is using the intensity centroid, which basically assumes that the intensity of a corner is offset from its corner and thus can be used for orienting the corner. The moments of a patch are defined as:

$$m_{pq} = \sum_{x,y} x^p y^q I(x,y) \quad [31]$$

And from these moments the centroid can be found as

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}}\right). \quad [32]$$

Then a vector can be created from the corner's center to the centroid, and the orientation is

$$\theta = atan2(m_{01}, m_{10}). \quad [33]$$

BRIEF descriptor describes an image patch that is constructed through a number of binary intensity tests. The binary intensity test is defined below, as p is a smoothed image patch

$$\tau(p; x, y) \coloneqq \begin{cases} 1 : p(x) < p(y) \\ 0 : p(x) \geq p(y) \end{cases} \quad [34]$$

The feature then is defined as of these binary intensity tests

$$f_n(p) \coloneqq \sum_{1 \leq i \leq n} 2^{i-1} \ \tau(p; x, y) \quad [35]$$

However, BRIEF's performance falls drastically even with only a few degrees of in-plane rotation. The solution to this is to construct a 2 by matrix for any feature set with binary intensity tests.

$$S = \begin{bmatrix} x_1 & \cdots & x_n \\ y_1 & \cdots & y_n \end{bmatrix} \quad [36]$$

In order to create a steered S, we need to use the patch orientation and the corresponding rotation matrix. is defined as below.

$$S_\theta = R_\theta S. \quad [37]$$

After these operations, the rotation-aware BRIEF is

$$g_n(p, \theta) \coloneqq f_n(p) | (x_i, y_i) \in S_\theta \quad [38]$$

C. Eight point Algorithm

After we collect matching points from a pair of frames, we can recover the fundamental matrix from these data using the eight-point algorithm. We know that the relationship between the fundamental matrix and a set of matched points is

$$x'^{\sim T} F x^\sim = 0. \quad [39]$$

So now we have m sets of these matched points

$\{x_m, x_m'\}$, $m = 1, \cdots, M$ and each correspondence satisfy the relation

$$x_m'^T F x_m = 0. \quad [40]$$

Which expands to

$$[x'_m \quad y'_m \quad 1]\begin{bmatrix} f_1 & f_2 & f_3 \\ f_4 & f_5 & f_6 \\ f_7 & f_8 & f_9 \end{bmatrix}\begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} = 0. \qquad [41]$$

This expands to one equation

$$x_m x'_m f_1 + x_m y'_m f_4 + x_m f_7 + y_m x'_m f_2 + y_m y'_m f_5 + y_m f_8 + x'_m f_3 + y'_m f_6 + f_9 = 0. \qquad [42]$$

Each correspondence will generate an equation like this, which means that form correspondences we will have a matrix like below

$$\begin{bmatrix} x_1x_1' & x_1y_1' & x_1 & y_1x_1' & y_1y_1' & y_1 & x_1' & y_1' & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_Mx_M' & x_My_M' & x_M & y_Mx_M' & y_My_M' & y_M & x_M' & y_M' & 1 \end{bmatrix}\begin{bmatrix} f_1 \\ f_4 \\ f_7 \\ f_2 \\ f_5 \\ f_8 \\ f_3 \\ f_6 \\ f_9 \end{bmatrix} = 0. \qquad [43]$$

Then we can solve the fundamental matrix using Singular Value Decomposition. With fundamental matrix, we can solve for the essential matrix, then the rotation and translation of the camera can be extracted.

D. Ransac

Random Sample Consensus (RANSAC) is used in this project for excluding outliers from our matchings. The pseudo code for RANSAC in this project is shown below

for each iteration:

rand ($[x_1 \quad \cdots \quad x_8]$) from$\{x_m, x'_m\}$

calculate fundamental matrix $[f_1 \quad \cdots \quad f_8]$ from$[x_1 \quad \cdots \quad x_8]$

compute the number of inliers n, keep the correspondence that satisfy

$$x'^{\sim T} F x^{\sim} \leq Threshold$$

choose the F with the greatest number of inliers

The entire processing procedure for the visual localization algorithm is shown with the figure below.

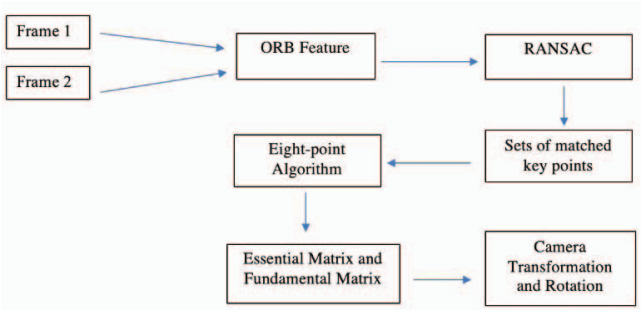

Figure 1. Algorithm Flowchart.

IV. Experimental Results

The following experimental result was conducted under the following processing power:

CPU: 2.3 GHz Intel Core i5

Memory: 16G

Graphics: Intel Iris Plus Graphics 655 1536 MB

To test our Visual SLAM algorithm for estimating the rotation and translation of camera, we used TUM data sets, including freiburg1_xyz, frieburg1_rpy, freiburg1_desk, freiburg2 rpy, and freiburg2_xyz. We approximated that we will make our algorithm estimate the rotation and transformation of the camera using 2000 key points for each pair of frames. To check the accuracy of our estimation, we calculated the Average Absolute Trajectory Error (ATE) between our estimation and the ground truth provided by the dataset. The formula for calculating the ATE is:

$$ATE = \frac{1}{N}\sqrt{(\hat{t}_x - t_x)^2 + \left(\hat{t}_y - t_y\right)^2 + (\hat{t}_z - t_z)^2} \qquad [44]$$

The image bellow shows the key points that were identified by ORB feature on a particular frame of frieburg1_desk of the TUM dataset.

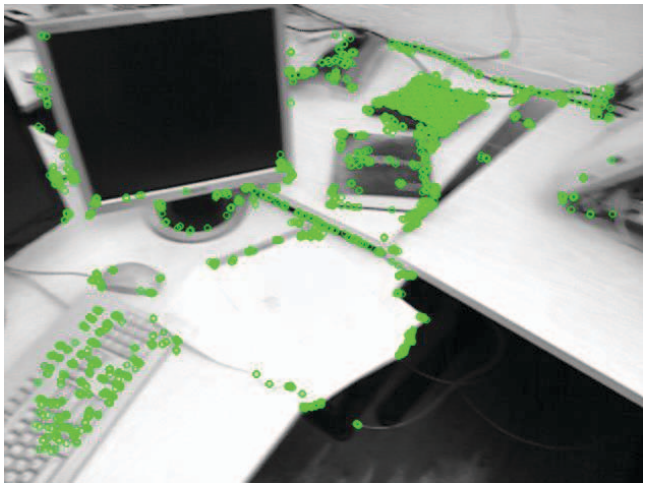

Figure 2. Key Points identified by ORB feature.

The image bellow shows our algorithm matching the key points from consecutive frames on a particular pair of frames of frieburg1_desk of the TUM dataset.

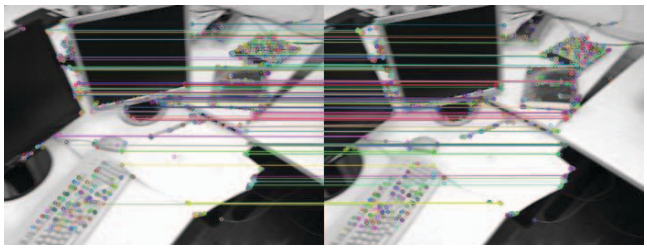

Figure 3. algorithm matching the key points from consecutive frames.

The results of the experiment are presented in the table below

TABLE I. RESULTS OF THE EXPERIMENT

	Dataset				
	rgbd_dataset_freiburg1_xyz	***rgbd_dataset_freiburg1_rpy***	***desk***	***2rpy***	***2xyz***
ATE Error	0.16724	0.07565	0.8411	0.07565	0.34676

Normally, any ATE error below 1 would be considered a relatively accurate prediction, in which our algorithm succeeded in providing an accurate prediction for all five data sets selected.

V. CONCLUSION AND DISCUSSION

Through this paper, we presented a sparse feature based visual localization algorithm that uses ORB feature as the feature detector. This system is able to conduct real time visual localization with moderate processing power, proving that the algorithm can theoretically be transformed to high-end mobile devices. The SLAM algorithm used in this research is specifically combined with other algorithms to tackle indoor camera rotation and translation issues, where GPS cannot be accessed. The overall performance of our method reaches state-of-the-art level as the ATEs for all five datasets are below 1. However, this algorithm still presents weaknesses as its requirement of processing power cannot be omitted. The performance in terms of accuracy of this algorithm largely relies on the number of key points extracted and compared for every frame, where extracting too many key points at a time requires too much processing power. Future work would be directed towards optimizing the efficiency of calculations.

REFERENCES

[1] J Engel, T Schöps, D Cremers, "LSD-SLAM: Large-Scale irect Monocular SLAM,"European conference on computer vision, 2014.

[2] T Zhou, M Brown, N Snavely, "Unsupervised Learning of Depth and Ego-Motion from Video,"The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[3] A Kendall, M Grimes, R Cipolla, "PoseNet: A Convolutional network for Real-Time 6-DOF Camera Relocation,"The IEEE International Conference on Computer Vision (ICCV), 2015.

[4] C Forster, M Pizzoli, D Scaramuzza, "SVO: Fast Semi-Direct Monocular Visual Odometry," 2014 IEEE International Conference on Robotics and Automation (ICRA), 2014.

[5] E Rublee, V Rabaud, K Konolige, GR Bradski,"ORB:an efficient alternative to SIFT or SURF,"The IEEE International Conference on Computer Vision (ICCV), 2011.