A Stable and Accurate Marker-less Augmented Reality Registration Method

Qing Hong Gao
College of Electronic and Information
Xian Polytechnic University, Xian, China

Tao Ruan Wan
Faculty of Informatics and Engineering
University of Bradford, Bradford, UK

Wen Tang
Faculty of Science and Technology
Bournemouth University,Bournemouth, UK

Long Chen
Faculty of Science and Technology
Bournemouth University,Bournemouth, UK

***Abstract*—Markerless Augmented Reality (AR) registration using the standard Homography matrix is unstable, and for image-based registration it has very low accuracy. In this paper, we present a new method to improve the stability and the accuracy of marker-less registration in AR. Based on the Visual Simultaneous Localization and Mapping (V-SLAM) framework, our method adds a three-dimensional dense cloud processing step to the state-of-the-art ORB-SLAM in order to deal with mainly the point cloud fusion and the object recognition. Our algorithm for the object recognition process acts as a stabilizer to improve the registration accuracy during the model to the scene transformation process. This has been achieved by integrating the Hough voting algorithm with the Iterative Closest Points(ICP) method. Our proposed AR framework also further increases the registration accuracy with the use of integrated camera poses on the registration of virtual objects. Our experiments show that the proposed method not only accelerates the speed of camera tracking with a standard SLAM system, but also effectively identifies objects and improves the stability of marker-less augmented reality applications.**

Keywords: Augmented Reality, SLAM Algorithm, Point Clouds, Hough voting algorithm

I. Introduction

Augmented Reality (AR) is the technology of mixing real scenes with virtual information. As an emerging field with huge application potentials, AR technology enhances human perception of the world and adds novel interactions between human and computers. Azuma [1] has defied that AR is an integration of the virtual world and the real world with real-time interactions via three-dimensional registrations. Therefore, a stable real-time registration performance between the real and the virtual world via 3D mapping and objection recognition is at the core of the marker-less AR technology.

The rapid development in software and hardware technologies in virtual reality and computer vision has made the AR technology applicable to a wider range of applications from medicine, military,entertainment to many others [2] [3]. Virtual registrations, however, remain a challenge issue in AR research. Initially the Simultaneous Localization and Mapping (SLAM) algorithm has been used mainly in robotics for positioning robots in unknown environments [4] [5]. More recently, researchers have started to utilize the state-of-the-art SLAM for virtual information and virtual object registrations in AR. Davison *et. al.* [6] [7] have used a monocular camera to achieve fast 3D modeling and camera pose tracking in unknown environments, which has shown potentials of the SLAM algorithm to be used in many other applications, such as AR. Klein [8] has applied a SLAM algorithm to create three-dimensional point clouds, and Reitmayr [9] has demonstrated the use of SLAM and sensor fusion techniques to improve marker-less tracking for virtual object registrations.

A method for computing the homography matrix in AR systems for three-dimensional registrations has been shown in [10] [11]. Although simple and efficient, this method has to detect coordinates of four points of a plane in order to determine the camera pose (translation and rotation) w.r.t. the world coordinate system. In spite of its simplicity and efficiency, the fundamental principle of the algorithm was based on the 2D plane registration. Hence, the four points detection algorithm is prone to the error of misplacement of virtual objects during the registration process, resulting in the virtual objects being unstable with distracting visual artifacts (i.e. flashing visual effects). To deal with this issue, previous approaches [8] [9] have attempted to make the use of three-dimensional map information generated by a SLAM algorithm to improve the registration accuracy. Despite great stride has been made in recent years regarding the improvement to AR techniques, both in software and hardware techniques, the stability and performance issues in AR registration remain as an unsolved problem.

In this paper, we present a new method to improve the registration between the real world and the virtual world and also the tracking of virtual objects. Our method also utilize the 3D map information generated by SLAM. Because of the problem of using sparse point clouds for identifying objects, RGBD sensor can used to achieve a dense map. The KinectFusion framework [12] is well known for the real-time reconstruction of dense map obtained from RGBD sensors, then the ElasticFusion algorithm [13] can be used to achieve the fast and accurate real time reconstruction. However, the both algorithms must rely on GPU acceleration for real-time performance, demanding higher hardware hardware require-

ments than normal commodity PC and real-time software build in the CPU are still mainly working with sparse maps [8] [14].

We develop our new approach based on our observations that, although RGBD images can be used [15], the advantage of the depth information provided by the RGBD image has only been used during initialization step. During the map construction process, however, the map is still using the conventional ORB feature points to built the map, but did not use the depth of information. Therefore, we design our algorithm by adding a real-time dense map to the sparse point clouds to increase the accuracy of the registration. Furthermore, we integrate the Hough voting [16] algorithm with the SLAM framework for detecting and recognizing objects, and add the Iterative Closest Points(ICP) [17] algorithm to improve the precision of the transformation matrix. Not only can our proposed new algorithm accurately detect the object, it can also located the exact location of the object. Finally, we use the precise positioning function of the Visual SLAM (V-SLAM) and the transformation matrix to set camera poses in order to render the coordinate system under the camera frame for the three-dimensional registration of the virtual object. The main contribution of this paper is to add a real-time dense map to improve the Hough voting algorithm and we have developed a novel approach that can effectively produce stable and high registration accuracy for virtual reality fusion in AR.

II. AR System Overview

Our proposed new AR framework consists of two software modules: a V-SLAM module and a registration module as shown in Fig. 1 for an overview of the system. Tracking in the V-SLAM module is for locating the camera position by processing each image frame and decide when to insert a new keyframe. Firstly, the feature matching process is initialized with the previous frame and the Bundle Adjustment (BA) [18] is used to optimize the camera poses. While the 3D map is initialized and the map is successfully created by the V-SLAM module, the registration module is called. The RGBD data is added to fuse the point cloud by the previously calculated pose to produce a dense map. We used the three-dimensional model to identify the object and get the transformation matrix. After this process together with the V-SLAM, the camera position is obtained and the pose is converted to the OpenGL coordinate system under the ModelView matrix. The final step is to register the 3D virtual object to the real world scene to achieve augmented reality.

A. Tracking

Tracking in our system is achieved via a visual simultaneous mapping and tracking strategy by extracting and matching the Oriented Features From Accelerated Segment Test (FAST) and the Rotated Binary Robust Independent Elementary Features (BRIEF) (ORB) [19]. We compute two models:i) a homography matrix that is used to compute a planar scene; ii) a fundamental matrix that is used to compute a non-planar scene. Each time the two matrices are calculated and scores ($M = H$ for the homography matrix and $M = F$ for the fundamental

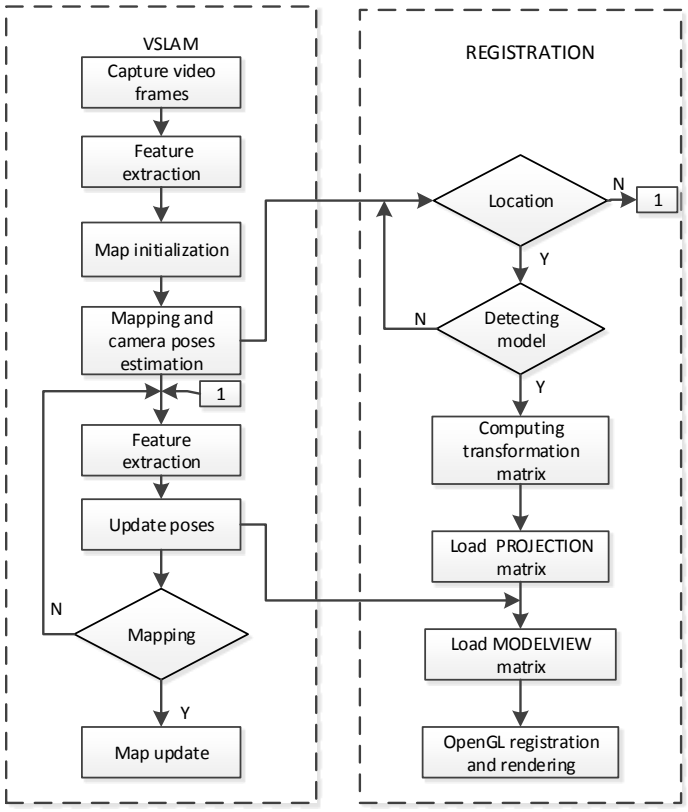


Fig. 1. System overview shows the workflow of our proposed AR framework and the components of the system

matrix) are also calculated as shown in equation 1. The scores are used to determine which model is more suitable for the camera posture.

$$S_M = \sum_i \left(\rho_M \left(d^2_{cr,M} \left(x^i_c, x^i_r \right) + \rho_M \left(d^2_{rc,M} \left(x^i_c, x^i_r \right) \right) \right) \right) \tag{1}$$

$$\rho_M \left(d^2 \right) = \begin{cases} \Gamma - d^2 \ if & d^2 < T_M \\ 0 \qquad if & d^2 \geq T_M \end{cases}$$

where d_{rc} and d_{cr} is the measure of symmetric transfer errors [20],T_m is the outlier rejection threshold based on the χ^2, Γ is equal to T_m, x_c is the features of the current frame, and x_r is the features of the reference frame. The BA is used to optimize camera poses, which gets a more accurate camera position as shown in the following equation:

$$\{R, r\} = \arg\min_{R,t} \sum_{i \in \chi} \rho \left(\left\| x^i - \pi \left(RX^i + t \right) \right\|^2_\Sigma \right) \tag{2}$$

where $R \in \mathcal{SO}^3$ is the rotation matrix, $t \in \mathbb{R}^3$ is the translation vector, $X^i \in \mathbb{R}^3$ is a three-dimensional point in space,$x^i \in \mathbb{R}^2$ is the key point, and ρ is the Huber cost function. Sigma item is the covariance matrix associated to the key point and π is the projection function.

After obtaining the accurate position estimation of the camera, the three-dimensional map of the point cloud is obtained by triangulating the key frames through the camera poses, and finally the local BA is used to optimize the map. A detailed description of the approach is given in [14].

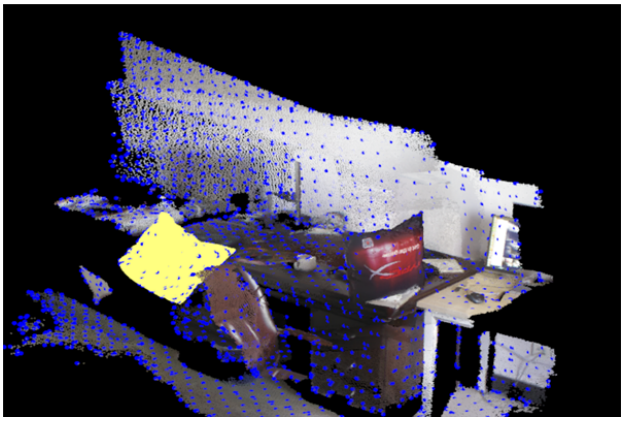
B. Dense mapping and 3D object recognition

In the dense map building process, we use a Kinect sensor to extract RGBD information, so that V-SLAM poses can be used to combine point clouds. The core of this method is to add a dense point cloud processing thread when the system is at the initialization stage, which creates a visual window for displaying a dense map. The map is not used to capture each frame of the image, but only with key frames. When the key frames of the system are updated, the RGB and the depth information of the current frame are extracted. Therefore, the point clouds are reconstructed from key-frame images. The pose of the current frame can be also obtained when processing the key frames. After that, we can transform the point cloud of the corresponding key frame into the same coordinate system according to the pose of the current key frame to generate a global point cloud map.

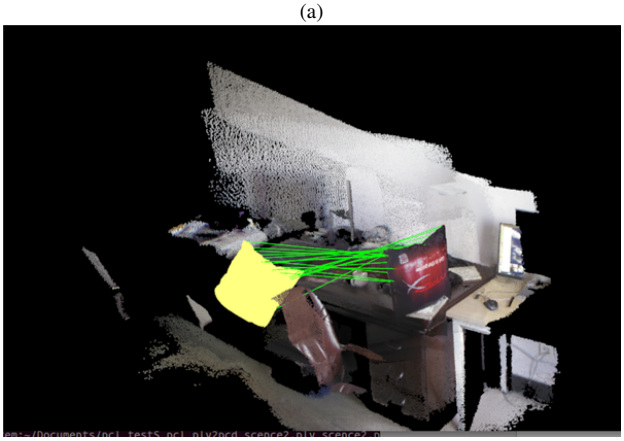
In our AR system, when the V-SLAM system is in the building mode, the object recognition is not performed. Object recognition is only running in the location mode. The Hough voting method is used for three-dimensional object recognition to increase the performance and accuracy of the ICP algorithm that maps the model to the corresponding model of the transformation matrix. The recognition results are shown in Fig 2. Fig. 2(a) shows the key points obtained using the uniform sample, whereas Fig. 2 (b) shows the descriptors of the model and the scene w.r.t. the matching of the corresponding points, and Fig. 2(c)shows the result of the final match. It can be seen that the algorithm can effectively identify the object. The details of the specific process is listed in Algorithm 1.

The virtual object is finally registered in the real world, which go through a series of coordinate system transformations (from the world coordinate system to the camera coordinate system to the crop coordinate system, and to the screen coordinate system). The transformation sequences can be described by equation 5 from left to right: the world coordinate system is transformed into the camera coordinate system by a rotation matrix $R_{3\times3}$ and a translation matrix $T_{3\times1}$. Those matrices are constructed by the camera's position and the detected plane information. Then the camera coordinate system is then transformed into the screen coordinate system (u, v) by the focal length $(f_x, f_y)$and the principal point (d_x, d_y). These parameters are obtained by the camera calibration. Finally, the virtual object is registered on the screen to the real world.

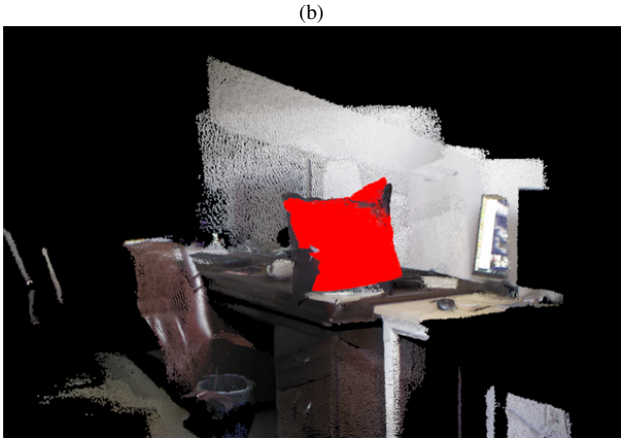
$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & d_x & 0 \\ 0 & f_y & d_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R_{3\times3} & T_{3\times1} \\ 0_{1\times3} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (5)$$



(a)



(b)



(c)

Fig. 2. Improved Hough voting (the red region is the matched model, the yellow region is the original model and the blue dots are the key points)

III. Experiment and Evaluation

Our experiment is run under a Ubuntu 14.04 system, CPU clocked at 2.3GHz, 8GB memory and NVIDIA GeForce GTX 960MB graphics card. The camera resolution is 640 by 480 pixels at 30 Hz. Fig. 3 (a) and (b) show the system identifies and registers a virtual table for a real table. Fig 3 (c) shows the system identifies and registers virtual laptop for a real laptop. Fig. 4 (a) also shows the identification and registration of the virtual tables. Fig. 4 (b) and (c) show the identification and registration of the 3D model reconstruction from a real chair.

Algorithm 1 Improved Hough vote

1: Using the nearest neighbor method to calculate the surface normal of the model and the scene separately. Calculating the surface normal can be done by solving eigenvectors and eigenvalues of a covariance matrix, which is created by neighboring elements of query points.The normal of each point can be obtained by equations 3 and 4;

$$C = \frac{1}{k}\sum_{i=1}^{k}\left(P_i - \overline{P}\right)\left(P_i - \overline{P}\right)^T \tag{3}$$

$$C \cdot \overrightarrow{v_j} = \lambda_i \cdot \overrightarrow{v_i}, j \in \{0, 1, 2\} \tag{4}$$

where C is the covariance matrix,k is the number of point neighbors considered in the neighborhood of P_i, $\overline{P}$ represents the 3D centroid of the nearest neighbors,$\lambda_i$is the j-th eigenvalue of the covariance matrix, and the j-th eigenvector.

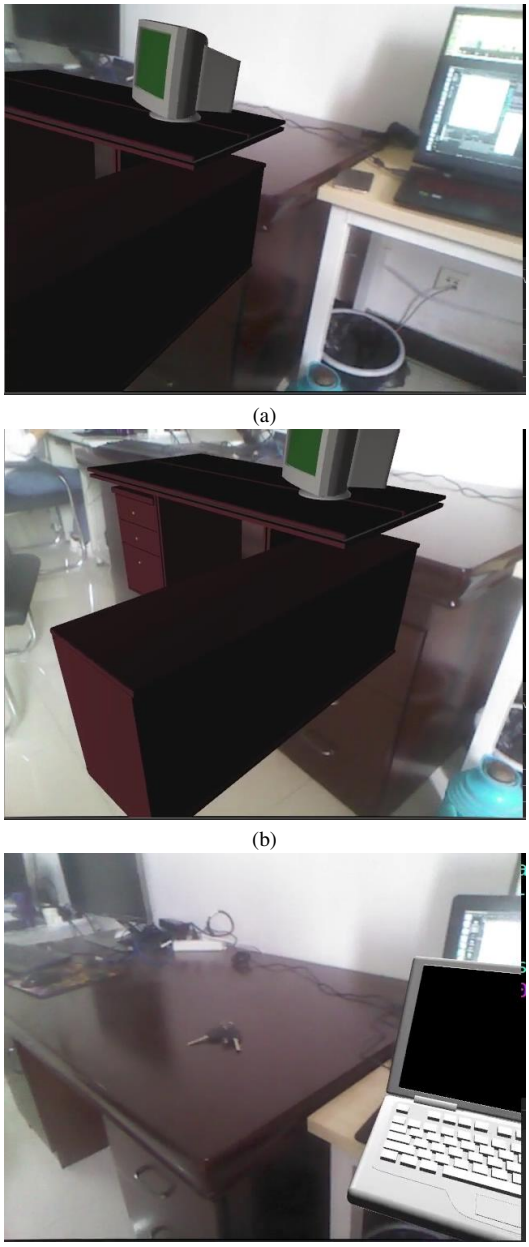
2: The Uniform Sampling algorithm is used to calculate the key points of the model and the scene. The Uniform Sampling algorithm mainly creates a 3D voxel grid, which calculates the centroid of each mesh within the grid, using the centroid of each grid to represent the entire point cloud;

3: Using the above-mentioned surface normal and the key points to calculate the Signature of Histograms of Orientations (SHOT) descriptors for models and scenes. A detailed description of the approach is given in [21];

4: By calculating the similarity(squared distance) between the model and the scene description point, the corresponding description points can be found;

5: then using the Hough voting to identify the object and calculate the corresponding transformation matrix;

6: The transformation matrix in step 5 is further processed by ICP to obtain a more accurate transformation matrix. =0



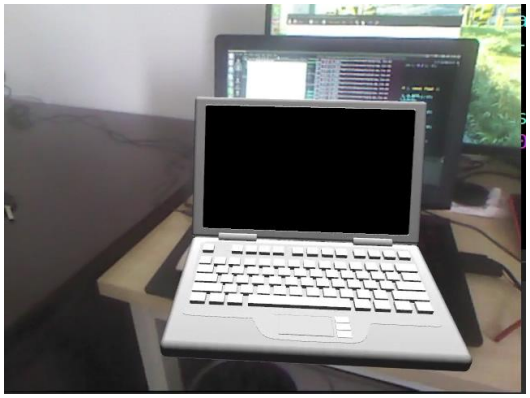
(a) (b) (c)

Fig. 3. AR tracking ,recognition and registration

As can be seen that the tracking,recognition and registration have been effectively performed correctly.

A. Object Recognition Analysis

The experiment is to split the model in the scene, and then match it with the original scene, then get the transformation matrix as the unit matrix. We give a fixed degree or a fixed translation distance on the basis of the unit matrix to obtain a new matrix, which called the reference matrix that is used in the experimental comparison. Then the model is multiplied by the reference matrix to get the new model. By using this new model, we use the Hough voting algorithm and the improved Hough voting algorithm respectively to obtain the transformation matrix of the model transformation to the scene. Here we use the similarity of the matrix (equation 6) for the rotation matrix and the European distance for the translation matrix respectively. Experimental results are shown in Fig. 5. Fig 5(a) shows the rotation angle fixed at 45 degrees, the abscissa indicates the increased distance (xyz components while increasing the same distance). The ordinate represents the error between the calculated and ground truth values. In Fig.5 (b), the translation component is fixed at 0.1cm, the abscissa represents the increased degree, and the ordinate represents the similarity. Although we can see that the rotation matrix does not change substantially from Fig.5 (b), the error of the translation matrix obtained by our improved method in Fig.5 (a) is much smaller.

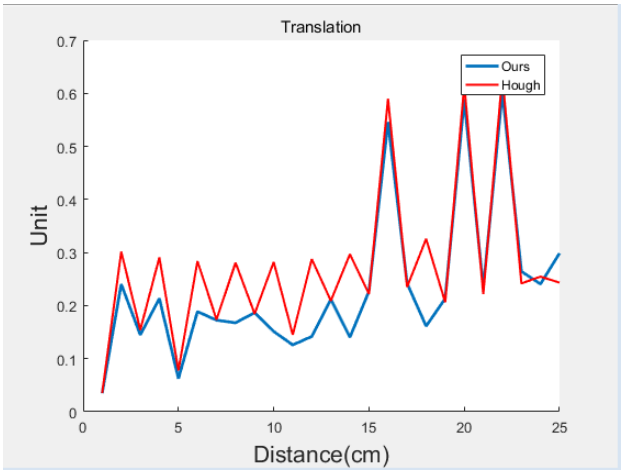


(a)

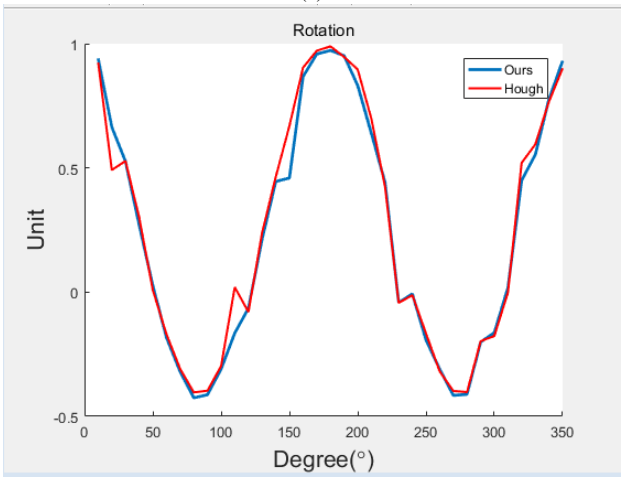
(b)

(c)

Fig. 4. AR tracking ,recognition and registration



(a)



(b)

Fig. 5. Recognition Analysis

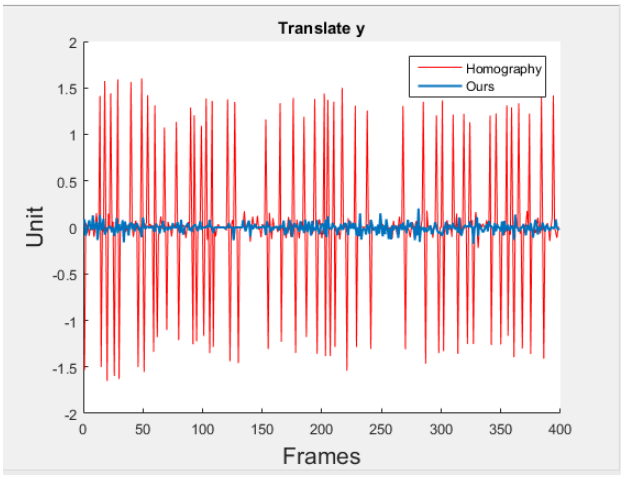
$$r = \frac{\sum_m \sum_n \left(A_{mn} - \overline{A}\right)\left(B_{mn} - \overline{B}\right)}{\sqrt{\left(\sum_m \sum_n \left(A_{mn} - \overline{A}\right)^2\right)\left(\sum_m \sum_n \left(B_{mn} - \overline{B}\right)^2\right)}} \tag{6}$$

where $\overline{A}$ and $\overline{B}$ are the means of matrix elements,mn is the m rows and n columns of the matrix,r is correlation coefficient of the matrix(-1 and 1 represent exactly the same matrix, 0 represent the two matrices are completely different).

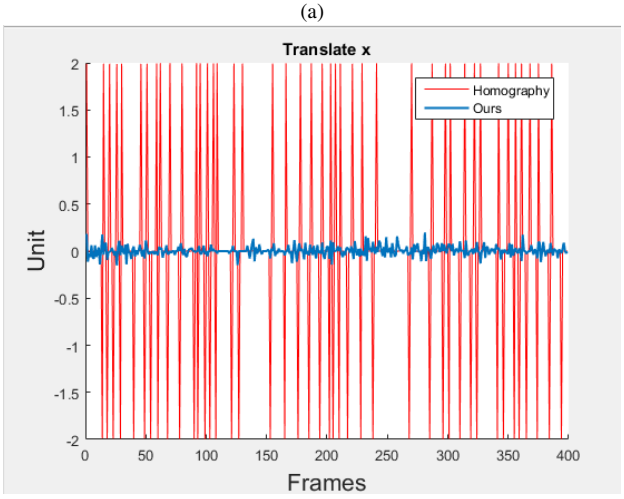
B. Registration Error Analysis

A comparison method is used with fixed camera positions to evaluate the robustness of our proposed method. The three-dimensional registration of the virtual object is carried out by using the described method and the standard homography matrix method. Six components of the three-dimensional registration results are analyzed. The difference between the transformation matrix of the current frame and the corresponding component of the transformation matrix of the previous frame is used as the basis for the comparison. The results are shown in Fig.6 and Fig. 7, where Translate x, Translate y and Translate z are the errors of the translation components, respectively, and Rotate x, Rotate y, Rotate z are relative to the x, y, z axis of the rotation component errors which are obtained by subtracting the previous frame from the current frame. The result of the rotation component is obtained by dividing the respective components with the dot product of the corresponding coordinate axis, and the translation component is the result obtained by the normalization process.

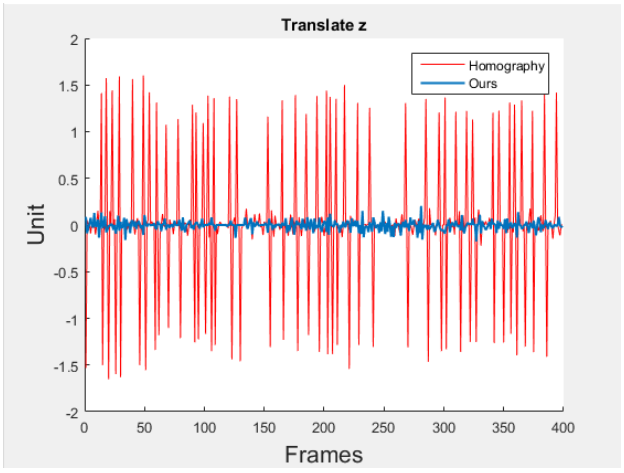
In Fig. 6 and Fig.7, the red curves are the results of using only the homography matrix, whereas the blue curves are the results of the new registration method used in this paper. As it can be seen from Fig. 6 and Fig.7, the use of the homography matrix method to register the virtual objects has produced



(a)

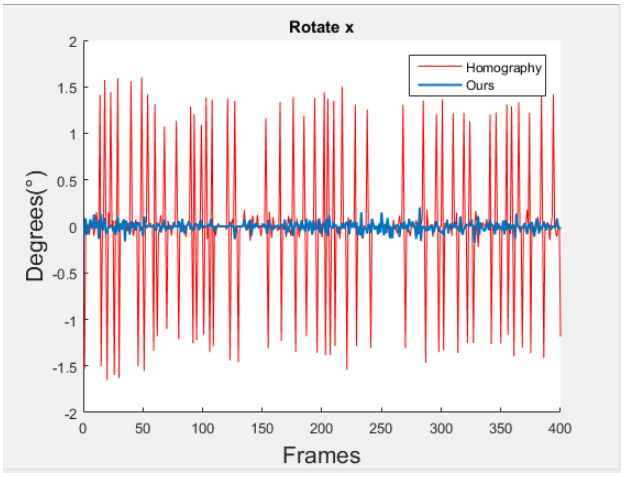


(b)

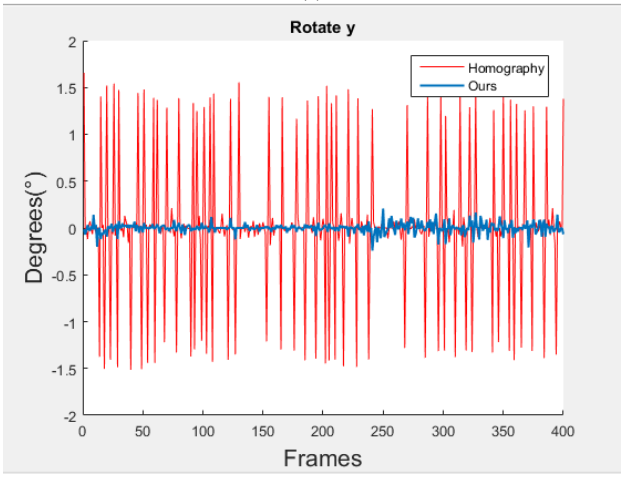


(c)

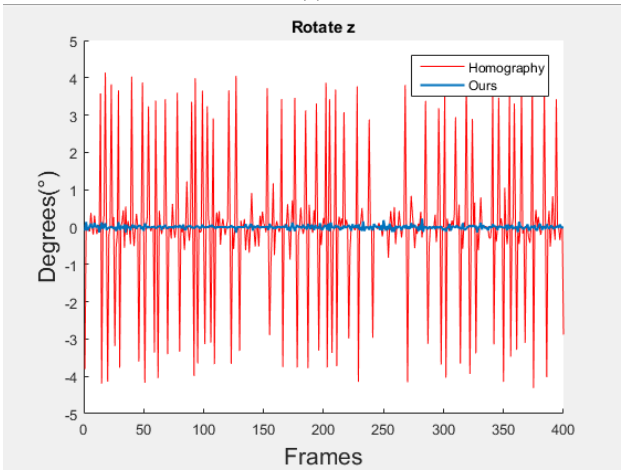
Fig. 6. Registration error



(a)



(b)



(c)

Fig. 7. Registration error

large registration errors that are equivalent to the virtual object registration instability. However, the new method tested on each rotation component has been kept the error in a small range below 0.5 degrees. The errors with Translate x, Translate y and Translate z are also small similar to the result of the rotation components.

Through the experimental results, it can be seen that the new method produces stable virtual registration and solve the flickering phenomenon in the virtual reality registration, hence, improves the stability of the AR system.

IV. Conclusions and Future Work

This paper presents a stable and high performance realistic tracking and recognition method in AR based on three-dimensional map information generated by V-SLAM. The method allows the tracking and the registration of virtual objects to ensure a stable and real-time performance of markerless AR applications. Our proposed method is faster than the standard methods and is able to achieve more accurate registration results compared with the state-of-the-art approaches. The experimental results show that the proposed method can effectively suppress the virtual object jittering, having a higher tracking accuracy with good performance.

At present, we are using object recognition based on model recognition and only one object can be identified for each recognition. Therefore, we will consider multimodel 3D object recognitions based on deep learning in our future work.

Acknowledgement

This work is supported by Shanxi Province Science and Technology Department of International Cooperation Projects (2016JZ026)

References

[1] R. T. Azuma, "A survey of augmented reality," *Presence: Teleoperators and Virtual Environments*, vol. 6, no. 4, pp. 355–385, aug 1997.

[2] Z. Szalavri and M. Gervautz, "The personal interaction panel a two-handed interface for augmented reality," *Computer Graphics Forum*, vol. 16, no. 3, p. C335C346, sep 1997.

[3] R. Bimber, O.and Raskar, *Spatial Augmented Reality Merging Real and Virtual Worlds*. A K Peters Ltd Isbn 306, 2005.

[4] H. Strasdat, J. Montiel, and A. J. Davison, "Visual SLAM: Why filter?" *Image and Vision Computing*, vol. 30, no. 2, pp. 65–77, feb 2012.

[5] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): part ii," *IEEE Robotics Automation Magazine*, vol. 13, no. 3, pp. 108–117, Sep. 2006.

[6] A. J. Davison, W. W. Mayol, and D. W. Murray, "Real-time localization and mapping with wearable active vision," in *Proc. Second IEEE and ACM Int. Symp. Mixed and Augmented Reality*, Oct. 2003, pp. 18–27.

[7] ——, "Real-time workspace localisation and mapping for wearable robot," in *Proc. Second IEEE and ACM Int. Symp. Mixed and Augmented Reality*, Oct. 2003, pp. 315–316.

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

[9] G. Reitmayr, E. Eade, and T. W. Drummond, "Semi-automatic annotations in unknown environments," in *Proc. 6th IEEE and ACM Int. Symp. Mixed and Augmented Reality*, Nov. 2007, pp. 67–70.

[10] M. Fiala, "Artag, a fiducial marker system using digital techniques," in *Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, Jun. 2005, pp. 590–596 vol. 2.

[11] S. J. D. Prince, K. Xu, and A. D. Cheok, "Augmented reality camera tracking with homographies," *IEEE Computer Graphics and Applications*, vol. 22, no. 6, pp. 39–45, Nov. 2002.

[12] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *IEEE International Symposium on Mixed and Augmented Reality*, 2012, pp. 127–136.

[13] T. Whelan, R. F. Salas-Moreno, B. Glocker, A. J. Davison, and S. Leutenegger, "Elasticfusion: Real-time dense slam and light source estimation," *International Journal of Robotics Research*, 2016.

[14] R. Mur-Artal, J. M. M. Montiel, and J. D. Tards, "Orb-SLAM: A versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[15] R. Murartal and J. D. Tardos, "Orb-slam2: an open-source slam system for monocular, stereo and rgb-d cameras," 2016.

[16] F. Tombari and L. D. Stefano, "Object recognition in 3d scenes with occlusions and clutter by hough voting," in *Fourth Pacific-Rim Symposium on Image and Video Technology*, 2010, pp. 349–355.

[17] P. J. Besl and N. D. Mckay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 14, no. 3, pp. 239–256, 1992.

[18] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment — a modern synthesis," in *Vision Algorithms: Theory and Practice*. Springer Nature, 2000, pp. 298–372.

[19] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Proc. Int. Conf. Computer Vision*, Nov. 2011, pp. 2564–2571.

[20] H. R. Z. A, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[21] F. Tombari, S. Salti, and L. D. Stefano, "Unique signatures of histograms for local surface description," *Lecture Notes in Computer Science*, vol. 6313, pp. 356–369, 2010.