

Research Article

An Improved 3D Registration Method of Mobile Augmented Reality for Urban Built Environment

Yong Wu,^{1,2} Weitao Che ,^{3,4} and Bihui Huang⁵

¹*Institute of Geography, Fujian Normal University, Fuzhou 350007, China*

²*Fujian Provincial Engineering Research Center for Forest Carbon Metering, Fujian Normal University, Fuzhou 350007, China*

³*Beijing Freedo Technology Co., Ltd., Beijing 100080, China*

⁴*Institute of Space and Earth Information Science, The Chinese University of Hong Kong, Shatin, Hong Kong*

⁵*Fuzhou City Construction Design and Research Institute Co., Ltd., Fuzhou 350001, China*

Correspondence should be addressed to Weitao Che; cheweitao@link.cuhk.edu.hk

Received 19 May 2020; Revised 24 October 2020; Accepted 25 January 2021; Published 10 February 2021

Academic Editor: Michael J. Katchabaw

Copyright © 2021 Yong Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

3D registration plays a pivotal role in augmented reality (AR) system. The existing methods are not suitable to be applied directly in the mobile AR system for the built environment, with the reasons of poor real-time performance and robustness. This paper proposes an improved 3D registration method of mobile AR for built environment, which is based on SURFREAK and KLT. This method increases the building efficiency of algorithm descriptors and maintains the robustness of the algorithms. To implement and evaluate the registration method, a smart phone-based mobile AR system for built environment is developed. The experimental result shows that the improved method is endowed with higher real-time performance and robustness, and the mobile AR 3D registration can realize a favorable performance and efficiency in the complex built environment. The mobile AR system could be used in building recognition and information augmentation for built environment and further to facilitate location-based games, urban heritage tourism, urban planning, and smart city.

1. Introduction

Augmented reality (AR) refers to the organic integration of virtual information into the real-world scenes seen by the user [1], in which, mobile AR can enhance the user's understanding and perception of the real world and, meanwhile, strengthen the user's interaction with the real environment especially the built environment [2]. AR technology is characterized by virtuality-reality combination, real-time interaction, and 3D registration [3]. Among these, 3D registration is the core of the AR system and a pivotal indicator for evaluating the performance of AR systems, which is to solve the real-time mapping between 3D target object in the real world and 2D screen. Generally, 3D registration methods are mainly divided into hardware-based 3D registration and vision-based 3D registration.

The hardware-based 3D registration method realizes the positioning of ground objects in the 3D coordinate system of the device mainly by GPS positioning, map services, accel-

eration sensor, magnetic sensor, and other hardware devices, and it is mainly used in the navigation field. For example, Wu et al. [4] proposed a spatial information virtual-real registration method for hybrid hardware tracking and positioning to satisfy the demand for outdoor AR navigation; Li and Cheng [5] put forward a multisensor registration method based on the Android platform to be used in outdoor navigation. However, compared with the vision-based 3D registration method, it is greatly affected by the precision of the sensor and the outdoor environment, and the matching precision cannot easily satisfy the requirements of 3D registration in the outdoor environment.

Vision-based 3D registration identifies the features and locates the target objects with the help of the video stream acquired by the camera devices; it focuses on the image matching algorithm based on the natural feature points. However, it is mainly applied indoors; there are few image matching algorithms of mobile devices that could be used directly for built environment. The application in built

environment is susceptible to factors such as illumination, scale, and angle, thus resulting in poor matching effects. As a result, the immediate problem faced by mobile AR is to achieve the fast and efficient matching of targets under the context of complex built environment and premised on ensuring matching precision.

After a comprehensive analysis of the deficiencies of the existing algorithms, this paper proposes an improved mobile AR 3D registration method, which is aiming to achieve efficient 3D registration of built environment. The method effectively improves the real-time performance of the existing algorithms while maintaining a high matching precision and realizes fast and effective recognition of built environment and calculation of the initial 3D matrix. Meanwhile, the method takes full advantage of the time sequence and correlation of the frames in video streams and tracks and predicts the natural feature points of the built environment using the optical flow tracking algorithm, thereby improving the real-time performance of 3D registration.

To this end, this paper was organized in the following manner: the related works are introduced in Section 2. Section 3 provides the overall process for the improved 3D registration method of mobile AR for urban built environment. Experimental results and analysis are presented in Section 4. A pilot system is also implemented in Section 5. Finally, Section 6 provides the conclusions of this study and discussions on future prospects.

2. Related Works

The natural feature-based visual registration method [6] uses visual information related to the real scene, such as features of points, lines, and textures, with extraction and recognition of these features; camera tracking registration can be realized. This registration method does not need calibration objects and has high registration accuracy. It can be applied to camera tracking registration in large outdoor scenes. The key technology of 3D registration includes two parts: image matching and target tracking [7].

The image matching algorithm based on natural feature points is one of the key technologies of realistic 3D registration [8]. It realizes real-time detection and matching of target images and provides the required feature points for subsequent target tracking and registration. At present, SIFT [9], SURF [10], and other image matching algorithms are extensively applied based on natural feature points, and such algorithms are characterized by good matching effects and strong robustness. However, their real-time performance is relatively poor, which then leads to a low recognition efficiency and poor fluency of the mobile AR system. ORB [11], BRISK [12], FREAK [13], and other algorithms based on binary descriptors are successively proposed by researchers, and they have improved the computation speed compared with the floating-point descriptors used in previous algorithms. However, both the matching precision and robustness of these algorithms are poor, which in turn leads to poor matching effects and stability of the mobile AR system. Compared with other algorithms, SURF has better real-time performance and robustness. Dai et al. [14] segmented images

based on the dichotomy method and improved the real-time performance of the tracking by combining SURF with ORB, but the robustness of the algorithm was decreased; Gui et al. [15] proposed an AR tracking registration algorithm based on online learning of natural scenes, in a bid to increase the registration efficiency by improving the SURF descriptor for online learning, but the real-time performance was still low because the SURF descriptors were still of floating-point type.

3. Improved 3D Registration of Mobile AR for Built Environment

3.1. Improved SURF Algorithm. SURFREAK is combined by the SURF and FREAK algorithm to improve performance for mobile AR system. It is known that the SURF algorithm follows the SIFT scale-space theory in replacing the SIFT algorithm's Gaussian filter with a box filter and extracts feature points using the integral image and the HESSIAN matrix technique, so as to reduce the computation load of the SURF algorithm. As a result, feature point extraction of the SURF algorithm realizes a significant improvement in the calculation speed compared with the SIFT algorithm, but its floating-point 64D descriptor still fails to satisfy the real-time demand. The descriptor of the FREAK algorithm is provided with the binary computation and storage method, which has a greater advantage in the computation speed for descriptor generation, storage, and matching compared to the floating-point descriptor of the SURF algorithm. As a result, this paper proposes the improved algorithm SURFREAK by combining the features of small calculation amount and good robustness of the FREAK descriptor algorithm, so as to solve the SURF algorithm's problems of poor real-time performance and poor image matching in the complex built environment.

In general, SURF is used to extract feature points, and FREAK is used to extract descriptors based on the feature points. The implementation of SURFREAK is shown in Figure 1. First, image graying is conducted for the target image and the video stream image from the mobile phone. Then, the SURF algorithm is used to extract the feature points, and the descriptor of the extracted feature points is constructed by the FREAK algorithm. Finally, the K -nearest neighbor algorithm (KNN algorithm) is performed for feature point matching on the descriptor of the target image and the mobile video.

After KNN algorithm filtering, there are still cases of mismatching feature points, and therefore, the RANSAC algorithm is used to eliminate the mismatching. The algorithm has stronger noise immunity and can better complete the refinement of the matching point pairs.

3.2. KLT Optical Flow Tracking Algorithm. The 3D registration method based on the SURFREAK algorithm has increased the calculation speed to a certain extent, but it still fails to track the targets in the video scene in real time, and the registration method has not fully utilized the time sequence and correlation of the frames in video streams. Whereas, in the common mobile AR applications, the

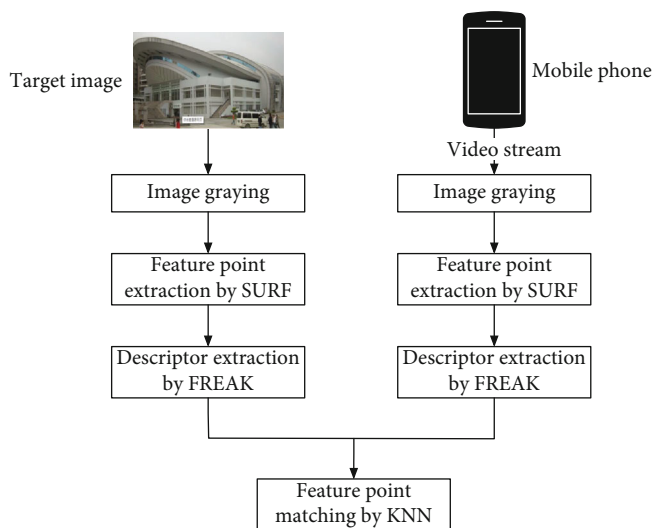


FIGURE 1: The flow chart of the SUFREAK algorithm.

variation of two adjacent frames in a video stream is typically small, so this feature can be used to obtain the transformation matrix of the current frame and previous frame by using the local search method of the KLT algorithm, thereby completing the tracking registration of the targets and making the AR 3D registration more efficient.

KLT (Kanade-Lucas-Tomasi) is the image feature optical flow tracking algorithm based on the optimal estimation [16], which speculates the rough position of the next frame's feature points using the feature points of the known image frame. The algorithm of this method can realize good noise immunity and real-time performance and is widely used in various fields.

For the image sequence with larger video scale transformation and motion amount, the combination of Gaussian pyramid layering and LK optical flow algorithm can better achieve the tracking of targets. The video stream acquired by the mobile terminal is layered by Gaussian sampling, and the LK optical flow computation starts from the lowest layer of the pyramid, and the low-layer computation result is used as the parameter of the next layer, and this process repeats until reaching the top layer of the pyramid. With tracking at the mobile terminal and through simulation experiments, the tracking computation speed can be effectively enhanced where the Gaussian pyramid layer $L = 4$ is set.

3.3. Framework of the Improved 3D Registration Algorithms.

In this paper, a real-time tracking registration method for feature points based on SURFREAK and KLT algorithms [17] (Table 1) is proposed. The 3D registration method first performs the image matching by SUFREAK and provides the feature points with higher accuracy for subsequent operations. Then, it tracks the feature points provided during the image matching process using the KLT tracking algorithm, so as to achieve the tracking of targets. The position, attitude, dimension, illumination, and other factors of the targets will change during the entire process of target tracking. These factors will lead to the tracking reduction and loss of feature points. On this point, the algorithm designs a dynamic

TABLE 1: Pseudocode of 3D registration algorithms.

Input: video frame	
Output: registration matrix	
Step 1:	Function CaculateRegistrationMatrix(video frame)
Step 2:	If exist a matching tag then
Step 9:	target tracking calculation using KLT with preImg, prePts and current frame curImg;
Step 9:	obtain the feature point nextPts of the tracking target;
Step 10:	if feature points are greater than the set threshold then
Step 11:	update the tracking parameters: PreImg=curImg, prePts=nextPts;
Step 13:	calculate the 3D registration matrix based on PNP algorithm of target feature points;
	else
	delete matching tag;
Step 12:	end if
	return;
Step 3:	else
Step 4:	SUFREAK algorithm extraction of feature points description of feature points;
Step 5:	carry out feature matching with offline target images;
Step 6:	if matched feature points are greater than the set threshold then
Step 7:	store the current frame image with feature points of preImg and prePts;
Step 7:	add matching tags;
Step 13:	calculate the 3D registration matrix based on PNP algorithm of garget feature points;
Step 8:	else
Step 8:	delete matching tag;
	end if
	return;
	end if
	End function

update strategy for feature points: when at least 30 of the tracked feature points are lost, the target recognition will be performed again by the image matching module and the feature points will be extracted again. At the same time, to improve the efficiency of image matching, the feature points of the target image can be extracted and the descriptors can be built in advance to reduce the time consumed by image matching.

The specific steps are as follows:

Step 1. Obtain a video stream from the mobile terminal, and perform grayscale processing on the video frame to determine whether there is a matching tag in the frame.

Step 2. If there is a matching tag, it is considered that the image matching has been performed. Proceed to step 8.

Step 3. If there is no matching tag, it is considered that no image matching has been performed.

Step 4. Using the SUFREAK algorithm, extract the feature points and build the descriptors for the video frame of the current scene.

Step 5. Perform feature matching with the offline target image, and eliminate mismatching using the RANSAC algorithm.

Step 6. Determine whether the scene is successfully matched with the offline target image according to the number of matching point pairs.

Step 7. If the matching succeeds, save the feature points and images extracted from the scene frame as prePts and preImg, respectively, and add the matching tag. Proceed to step 13.

Step 8. If the matching fails, delete the matching tag. Proceed to step 1.

Step 9. Perform the KLT target tracking calculation with stored preImg, prePts, and current frame curImg as parameters, and obtain the feature point nextPts of the tracking target.

Step 10. Determine whether the tracking is lost according to the number of tracking points nextPts.

Step 11. If the number of tracking points is greater than the set threshold, update the tracking parameters: preImg = curImg and prePts = nextPts. Proceed to step 13.

Step 12. If the number of tracking points is not greater than the set threshold, delete the matching tag. Proceed to step 1.

Step 13. Calculate the 3D registration matrix using the PNP algorithm according to the target feature points, so as to perform virtual-real integration or model attitude adjustment.

4. Experiment and Analysis

The experiment environment is as follows: (1) software development environment: Windows 7 (Intel (R) Core i3-2120 CPU 3.3 GHz, 4G memory) operating system and Eclipse development platform, development package OpenCV2.4.9 and ARToolKit for Android 5.2; (2) experimental equipment: Vivo V3L mobile phone, with 1.5 GHz eight-core processor and 3 G memory.

4.1. SUFREAK Algorithm

4.1.1. Experiment Scheme. To verify the performance of the SUFREAK algorithm proposed in this paper, the experiment selects three algorithms—SURF, ORB, and FREAK (extraction of feature points using the FAST algorithm) to carry out a comparative experiment. The experiment compares and analyzes the matching performance and stability from two aspects of matching speed and algorithm robustness (the robustness of image matching in complex built environments is susceptible to factors such as rotation, scale, illumination, and viewpoint shift; thus, the robustness experiments will be carried out as per these four main influencing factors), and the comprehensive performance evaluation of the algorithms is carried out using the two indicators of correct matching point pairs and matching fraction.

The matching performance is mainly measured by the matching speed indicator, which refers to the comprehensive time consumption of feature point detection, descriptor extraction, and feature matching. The stability is measured

by two indicators of correct matching point pairs (extracted under the four transformation images of rotation, scale, illumination, and viewpoint) and matching fraction. The correct matching point pair embodies the performance of the algorithm for image matching detection. The more matching point pairs there are, the better the matching effect [18]. The matching fraction is the ratio of the correct matching point pairs filtered from two matched images to the lesser number of feature points extracted in the two matched images [19], and it is used to measure the invariance of various algorithms under various image transformations. A larger value indicates better discrimination of the zone detector. The calculation equation of the matching fraction is as follows:

$$r_m = \frac{N}{\min(n_1, n_2)}, \quad (1)$$

where r_m is the value of the matching fraction, N is the number of correct matching point pairs extracted from two matched images, n_1 and n_2 represent the respective number of feature points detected in the same area on the two images to be matched, and $\min(n_1, n_2)$ is the minimum value of n_1 and n_2 .

4.1.2. Experimental Data. The experiment selects 10 building scenes as image data to test the real-time performance and robustness of the algorithms.

The test effects are similar; therefore, only one of the scenes is selected for description. As shown in Figure 2, it is composed of four sets of building scene images and each group consists of six images (size 960 * 720).

4.1.3. Matching Speed. To verify the real-time performance of the SUFREAK algorithm, this paper compiled the average matching time consumption of SUFREAK, SURF, FREAK, and ORB algorithms, as shown in Table 2.

From the comparison of average time consumption and matching point number of the algorithms in Table 1, the average time consumed by the improved SUFREAK algorithm in this paper is about 27.81% of the time consumed by the SURF algorithm, showing significant improvement in the calculation speed and overcoming the SURF algorithm's deficiency of poor real-time performance. The improved SUFREAK algorithm consumes more time in constructing a scale pyramid in order to ensure the scale invariance of the algorithm; hence, compared with ORB and FREAK algorithms, the algorithm is inferior in terms of computation speed, but it has a larger number of matching point pairs and a better matching effect.

4.1.4. Image Transformation Robustness. The algorithm comparison test is performed with the image transformation data set to verify the robustness of the algorithm in terms of viewpoint, illumination, angle of view, and rotation invariance. The experimental results of matching point pairs and matching fraction are as shown in Figure 3.

Under the illumination transformation, the number of matching point pairs of the algorithm decreases with the

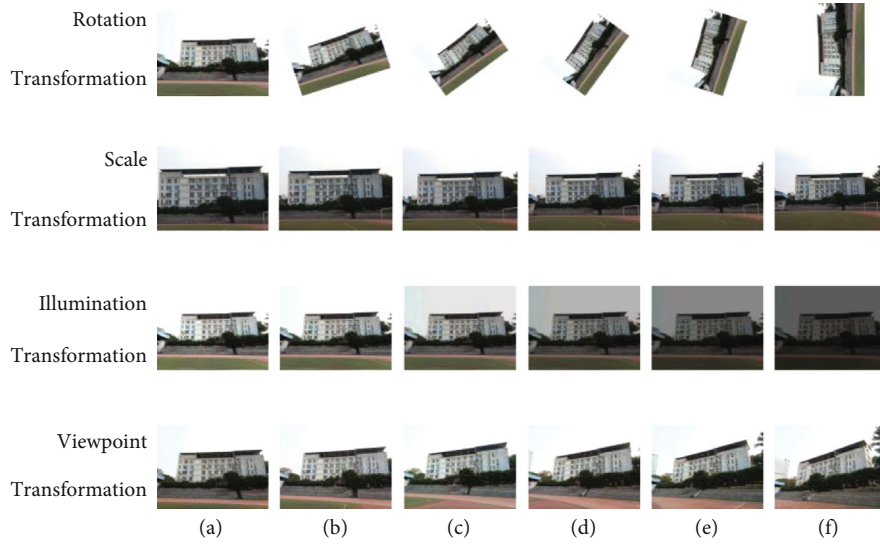


FIGURE 2: Outdoor scene test images for the experiment.

TABLE 2: Statistics of time consumption for image matching of each algorithm.

Time consumed (ms)	SUFREAK	SURF	FREAK	ORB
A to B	1,365	4,143	313	263
A to C	1,189	4,052	296	260
A to D	1,283	3,937	300	266
A to E	1,282	4,008	336	281
A to F	1,210	4,170	320	286
Average time consumed	1,266	4,062	313	271

Note: *‘‘A to B’’ refers to the image matching of photo A and photo B, and likewise for the others.

decrease of illumination brightness. The number of matching point pairs of the FREAK algorithm decreases significantly, while the number variation of matching point pairs of the ORB algorithm is relatively small; however, its matching fraction is low. In this paper, the SUFREAK algorithm has a higher average matching fraction than the SURF algorithm, and the SUFREAK algorithm presents a better illumination robustness. The reason lies in that the SUFREAK descriptors are generated by comparing the gray values of sampling point pairs after being subjected to Gaussian smoothing, and the Gaussian smoothing can reduce the influence of illumination. They are more statistical compared with the SURF algorithm, in which descriptors are created with the gray difference of adjacent pixel blocks.

The algorithms are more sensitive to the viewpoint transformation. As the viewpoint variation of the image is increased, the number of matching point pairs and the matching fraction are decreased more significantly. Compared with the performance of other algorithms under the viewpoint transformation, the SUFREAK algorithm in this paper has the largest number of matched point pairs and the highest matching fraction. Experimental results show that the SUFREAK algorithm proposed in this paper has better viewpoint transformation robustness than SURF and other algorithms.

Under the rotation transformation, each matching algorithm sees the biggest drop in the number of matching point pairs after the image is rotated by about 45 degrees, but when the image is rotated by close to 90 degrees, the number of matching point pairs is increased rapidly. Similarly, the matching fraction curve shows a trend of decreasing first and then increasing. Under the rotation transformation, both FREAK and ORB show strong robustness. Meanwhile, the number of matching point pairs of the SUFREAK algorithm has a small difference with that of the SURF algorithm, but the matching fraction curve of the SUFREAK algorithm is higher overall than that of the SURF algorithm, and consequently, the rotation robustness is better than that of the SURF algorithm. The reason may lie in that the SURF algorithm determines the main direction of feature points by calculating the vector composed of the Haar wavelets in its circular neighborhood sliding with a 60° fan-shaped window. In the SUFREAK algorithm, the main direction of feature points is determined by the longest and symmetrically connected 45 sampling points around the 43 sampling points surrounding the feature points in the entire neighborhood window, which is more easily affected by the angle variation.

Under the scale transformation, the matching point pairs and matching fraction of various algorithms present a downward trend as a whole, while the SUFREAK algorithm and SURF algorithm see a small increase during the course of decline. Among them, the SURF and SUFREAK algorithms show a better robustness in terms of matching point pairs and matching fraction. During the image transformation, the SUFREAK algorithm has a slightly weaker robustness than the SURF algorithm in overall scale transformation, but in small-scale transformation, the scale transformation robustness of the SUFREAK algorithm is better than that of the SURF algorithm. Meanwhile, both the ORB and FREAK algorithms present a faster decline in both the number of scale matching point pairs and the matching fraction curve, and the robustness of the scale invariance is relatively poor.

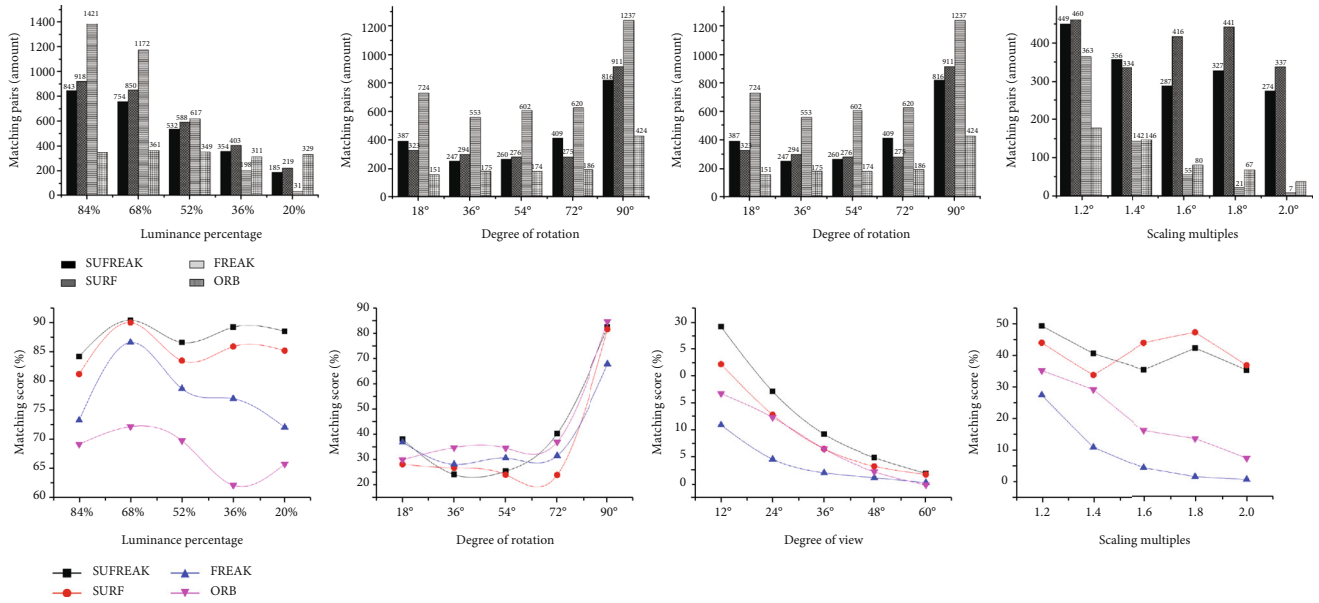


FIGURE 3: Comparison of matching point pair number and matching fraction curve in image transformation.

The experimental results demonstrate as follows: compared with the SURF algorithm, the improved SUFREAK algorithm manifests a higher operating efficiency and satisfies the real-time demand, and the robustness of the algorithm under viewpoint and illumination transformation outperforms the SURF algorithm to a certain extent. At a certain level, this is helpful in solving the problem of poor image matching resulting from different angles of view and illumination differences in complex outdoor environments.

4.2. 3D Registration Experiment Based on SURFREAK and KLT. To verify the efficiency of the KLT optical flow tracking algorithm for the mobile AR system, a comparative experimental scheme based on the SURFREAK and KLT algorithms is designed: Firstly, on the basis of the SURFREAK and KLT feature tracking method, recognize and track the images for the frame sequence composed by the same image in the video stream. Then, record the time, respectively, consumed by the image matching module and the target tracking module. Finally, carry out a comparative analysis for the time consumption statistics of the two modules. The experimental video data is acquired by moving the front-end camera of the Android phone. The average frame frequency of the video stream is about 27-28 frames/s, and the video image size is $480 * 320$ pixels. The tracking effect of the KLT optical flow algorithm is as shown in Figure 4. The left figure shows the tracked feature points of the previous frame, the right figure shows the feature point offset after tracking (right), and the length of the red lines represents the offset (motion amount) between the previous frame and the next frame of the target image in the video stream.

The experimental results are as shown in Tables 3 and 4, which are the time consumption statistics of the SURFREAK image matching algorithm and KLT tracking algorithm, respectively. The average time consumed to match an image based on the SURFREAK algorithm is about 222.11 ms, while



FIGURE 4: Feature point tracking effect of the KLT optical flow algorithm.

TABLE 3: Time consumption statistics of the SURFREAK algorithm.

SURFREAK (/times)	1	2	3	4	5	6
Time consumed (ms)	202	210	213	253	225	228

TABLE 4: Time consumption statistics of the KLT tracking algorithm.

KLT algorithm	11- frame	22- frame	33- frame	44- frame	55- frame	66- frame
Time consumed (ms)	109	100	97	108	84	87

the average time taken by the KLT optical flow tracking algorithm is about 97.85 ms. The time consumed by the KLT tracking algorithm accounts for about 44.05% of that of the SURFREAK image matching algorithm. The image matching

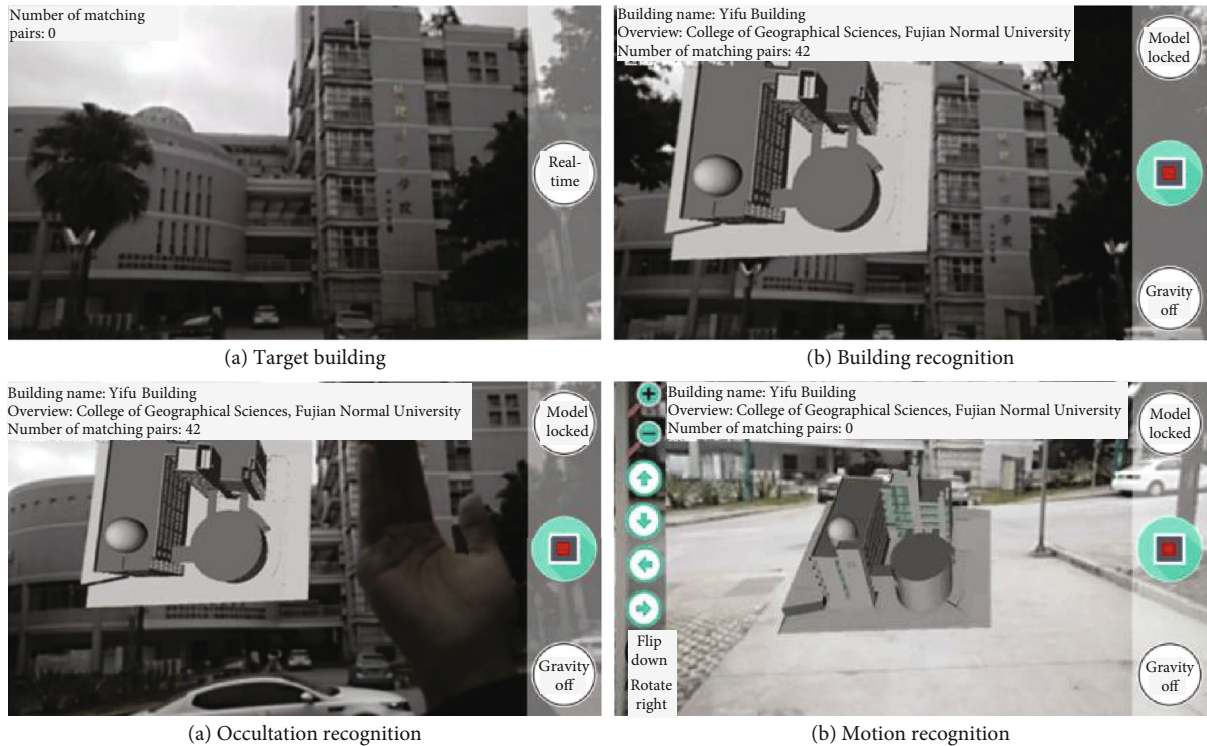


FIGURE 5: Rendering of mobile AR based on improved SURF and KLT algorithms.

algorithm can be effectively replaced for target tracking, thus increasing the fluency of the mobile AR system.

5. System Implementation and Results

Based on the improved 3D registration method, this paper designs and implements a prototype Android-based mobile AR system and applies it to the recognition and information augmentation in outdoor built environment. In the first step, the current position information of the user is obtained by GPS of the mobile phone, and spatial retrieval is performed within a certain radius with this position as the center, to search for the building images in this space area, so as to carry out a coarse filtering of the matched images (Figure 5(a), number of matching pairs is 0). Then, the mobile phone is aligned with the building image to match and track the building image, and the augmented reality is performed on the building by superimposing text, 3D models, and other information (Figure 5(b), building name: Yifu Building; overview: College of Geographical Sciences, Fujian Normal University; number of matching pairs: 42). Lastly, the system implements the quick recognition of the building through occultation (Figure 5(c)) and motion experiments (Figure 5(d)). The recognition and tracking efficiency is relatively high.

6. Conclusion

The robustness of the image matching algorithm of mobile AR application is highly required because of the lighting and viewpoint shift. However, the existing SURF algorithm is too inefficient due to the performance of mobile device.

This paper proposes an improved 3D registration method of mobile AR for built environment based on the SURF-FREAK and KLT algorithms. The method improves the SURF algorithm by combining the FREAK algorithm, so that the improved algorithm inherits both the strong robustness of the SURF algorithm and the quick extraction advantage of FREAK descriptors, thereby effectively remedying the SURF algorithm's poor real-time performance and achieving highly efficient recognition of outdoor built environment. After image recognition is completed, the adoption of the optical flow tracking method effectively improves the real-time and processing capabilities of the system. The experimental results show that the 3D registration method features strong robustness, high registration precision, and good real-time performance, satisfying the requirements of the mobile AR system for recognition efficiency and real-time performance of 3D registration. The contribution of this work is that the problems of the tracking target in the tracking process are solved, such as deformation and partial occlusion, which make the target tracking more robust. The mobile AR system is expected to be used in the building recognition and information augmentation for built environment and further to provide a useful tool for location-based games, urban heritage tourism, urban planning, smart city, etc.

The experimental results show that the 3D registration method features strong robustness, high registration precision, and good real-time performance, satisfying the requirements of the mobile AR system for recognition efficiency and real-time performance of 3D registration, which is a very important factor for AR game experience. Although the improved SURF-FREAK algorithm proposed in this paper manifests significantly improved matching speed, there are still

some future works to be undertaken. That is, the dichotomy method may be adopted to segment the images without affecting the robustness of the algorithm to further increase the matching efficiency.

Data Availability

There are no synthetic data used for this research.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

This research was funded by the Special Fund for Public Welfare Scientific Institutions of Fujian Province, No. 2020R11010009-2.

References

- [1] G. Liestol and A. Morrison, "Views, alignment and incongruity in indirect augmented reality," in *2013 IEEE International Symposium on Mixed and Augmented Reality - Arts, Media, and Humanities (ISMAR-AMH)*, pp. 23–28, Adelaide, SA, Australia, 2013.
- [2] H. Zhigang, Y. Kong, and Q. Yaochen, "Research on geographic representation: a review," *Progress in Geography*, vol. 30, no. 2, pp. 141–148, 2011.
- [3] L. Xiaomei, W. Zhongdong, Y. Jiu, and J. Xiaohong, "Artoolkit 3d registration based on dynamic feature template matching," *Computer Applications and Software*, vol. 34, no. 7, pp. 104–109, 2017.
- [4] X. Wu, F. Ren, and D. Qingyun, "Virtual-real registration of spatial information based on hybrid hardware tracking and location," *Geography and Geo-Information Science*, vol. 26, no. 3, pp. 15–18, 2010.
- [5] L. Dan and G. Cheng, "Application and research on android platform-based mobile augmented reality," *Computer Applications and Software*, vol. 32, no. 1, pp. 16–19, 2015.
- [6] Z. Xu and G. Wang, "Research on the method of 3D registration technology," *Earth and Environmental Science*, vol. 440, no. 3, pp. 1–8, 2020.
- [7] W. Ma, H. Xiong, X. Dai, X. Zheng, and Y. Zhou, "An indoor scene recognition-based 3D registration mechanism for real-time AR-GIS visualization in mobile applications," *ISPRS International Journal of Geo-Information*, vol. 7, no. 3, pp. 112–126, 2018.
- [8] M. Ozuysal, M. Calonder, V. Lepetit, and P. Fua, "Fast keypoint recognition using random ferns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 3, pp. 448–461, 2010.
- [9] R. Blaine, G. Wang, M. Wu, and J. R. Cavallaro, "A fast and efficient SIFT detector using the mobile GPU," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2674–2678, Vancouver, BC, Canada, 2013.
- [10] F. Guo, X. Luo, and L. Yi, "Research on feature extraction and match method based on the surf algorithm for mobile augmented reality system," in *2015 International Industrial Informatics and Computer Engineering Conference*, pp. 615–619, Xi'an, Shaanxi, China, 2015.
- [11] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *2011 International Conference on Computer Vision*, pp. 2564–2571, Barcelona, Spain, 2011.
- [12] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: binary robust invariant scalable keypoints," in *2011 International Conference on Computer Vision*, pp. 2548–2555, Barcelona, Spain, 2011.
- [13] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: fast retina keypoint," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 510–517, Providence, RI, USA, 2012.
- [14] D. Xuemei, L. Lang, and C. Mengyuan, "Research of image feature point matching based on improved ORB algorithm," *Journal of Electronic Measurement and Instrumentation*, vol. 30, no. 2, pp. 233–240, 2016.
- [15] Z. W. Gui, Y. Liu, J. Chen, and Y. T. Wang, "Online learning of tracking and registration based on natural scenes," *Journal of Software*, vol. 27, no. 11, pp. 2929–2945, 2016.
- [16] R. J. Mstafa and K. M. Elleithy, "A video steganography algorithm based on Kanade-Lucas-Tomasi tracking algorithm and error correcting codes," *Multimedia Tools and Applications*, vol. 75, no. 17, pp. 10311–10333, 2016.
- [17] H. Xiang and M. Lianquan, "A robust watermarking algorithm using SURF feature regions," *Geomatics and Information Science of Wuhan University*, vol. 42, no. 3, pp. 421–426, 2017.
- [18] L. Hui, L. Qingquan, and C. Gao, "Moving target detection using C_SURF registration," *Geomatics and Information Science of Wuhan University*, vol. 39, no. 8, pp. 951–955, 2014.
- [19] F. Yidong and S. Yue, "Image matching algorithm based on SURF feature extraction and FLANN search," *Journal of Graphics*, vol. 4, pp. 650–654, 2015.