ACTIVE CAMERA RELOCALIZATION WITH RGBD CAMERA FROM A SINGLE 2D IMAGE

Dongxu Miao[1,3*] *Fei-Peng Tian*[2,3*] *Wei Feng*[2,3†]

[1] School of Computer Software, Tianjin University, Tianjin, 300350, China
[2] School of Computer Science and Technology, Tianjin University, Tianjin, 300350, China
[3] Key Research Center for Surface Monitoring and Analysis of Cultural Relics, SACH, China

ABSTRACT

Active camera relocalization (ACR) focuses on dynamically and physically relocating camera to a previous pose, effectively supporting many applications in computer vision and robotics, such as automated picking and stowing, fine-grained change detection. Previous work [1] uses barely 2D images to realize ACR, bringing about unknown translation scale problem. To solve this problem, they use bisection approach to guess translation scale, which leads to reciprocating motion and slows down the convergence process. In this paper, we utilize additional depth information from an RGBD camera to solve the real translation scale problem. Via iteratively and sequentially adjusting 3D translation and rotation, our ACR approach greatly reduces the iteration number and speeding up the process. To cope with imprecise pose estimation and achieve high relocalization accuracy, we propose a bounding strategy to restrict camera motion. Experiments validate the proposed method is much efficient and its accuracy is on par with previous ACR method.

***Index Terms*—** Active camera relocalization, Depth camera, Camera pose estimation, Hand-eye calibration

1. INTRODUCTION

Active camera relocalization (ACR), which focuses on estimating camera pose from 2D or 3D images and actively recurring it, is one of the important problems in computer vision and robotics [2, 3, 4, 5], but is not yet studied extensively. Most previous work focuses on camera pose estimation, which is also referred to as camera registration, ego-motion estimation, camera (re-)localization, or camera regression in different literatures. However, camera pose estimation only concerns statically estimate the camera pose in a known reference coordinate which is defined by another camera pose or by a 3D scene constructed from SfM [6, 7], SLAM [8, 9, 10]. In this paper, as shown in Fig. 1, we treat camera relocalization as the process of dynamically and physically relocating the camera to a target pose of the single 2D reference image, beyond merely estimating the relative pose.

†W. Feng is the corresponding author. Email: wfeng@tju.edu.cn. *The first two authors are joint first author, who equally contribute to this work.

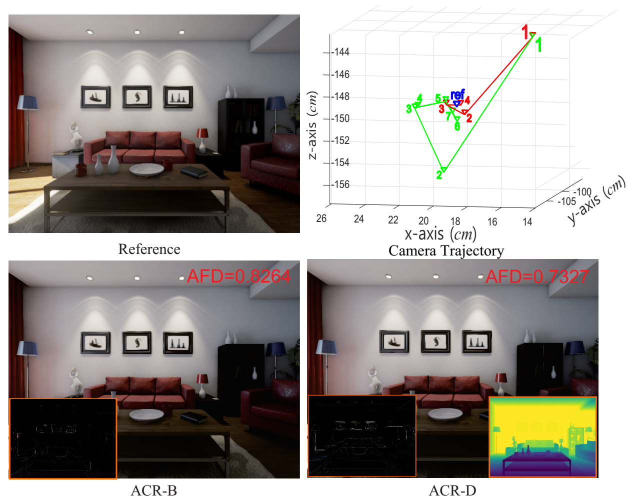


Fig. 1. Comparision of the proposed ACR-D with ACR-B [1]. ACR-D has comparable relocalizaton accuracy (by AFD value and difference image in the left-bottom), and what is more, as shown in camera trajectory, ACR-D (red) needs much fewer iterations and avoids detours compared with ACR-B (green).

Comparing with just estimating camera pose, physically relocating to original camera pose can benefit many applications in computer vision and robotics. One application is automated picking and stowing. As pointed out by the Amazon Robotics Challenge committee [11], viable automated picking and stowing in unstructured environments still remains a difficult challenge. ACR can play a key role in driving the robot arm to a known good position for picking and stowing using an eye-in-hand system. Another application is fine-grained change detection [4] which aims to accurately detect very minute changes. This requires two-time captured images are well aligned. Existing image-align methods [12, 13] can lead to unreal distortions which result in unreliable change detection. However, we can obtain highly and physically aligned images by ACR, thus provide a much more robust and accurate solution for fine-grained change detection.

Our paper builds on an existing ACR method [1]. We both wish to precisely relocate the camera to reference pose. Different from us, that work uses a normal RGB camera as the input source. By applying 5-points algorithm [14] for relative pose estimation, it iteratively adjusts the camera to reach reference pose. However, with only 2D input images, there exists the unknown translation scale problem for pose estimation. To address it, they design a bisection strategy to gradually approach actual real translation magnitude. The bisection method truly works in their situation, yet it leads to reciprocating motion and slows down the convergence process. In this work, we utilize additional depth information from an RGBD camera (ZED stereo camera) and estimate the camera pose with real translation scale. Thus the proposed method greatly speeds up the convergence process and can get comparable relocalization accuracy compared with [1]. In addition, without losing generality and make our method be applicable to existing 2D images, we also take an RGB image as the reference image and estimate the relative pose by EPnP [15] with 2D-3D correspondences. For a clear description, we refer the existing ACR approach [1] as ACR-B, and the proposed method in this paper as ACR-D.

Ideally and theoretically, when relative camera pose is estimated by PnP methods, we can just move the camera by an inverse camera pose in one shot to finish relocalization. However, there exist two *challenges* that make one-shot relocalization unpractical. **C1:** Unknown hand-eye calibration problem [16, 17, 18]. Since camera motion is actually executed by the hand (motion platform) and there is a displacement $\mathbf{X}$ between camera coordinate $\mathbf{A}$ and the hand coordinate $\mathbf{B}$ which satisfies $\mathbf{AX} = \mathbf{XB}$. With unknown hand-eye calibration $\mathbf{X}$, we cannot get precise hand motion merely from the estimated camera pose. However, existing hand-eye calibration is complicated and not reliable which limits the application of ACR on a common device that is often disassembled/reassembled. **C2:** The estimated pose is often not precise enough, thus one-shot relocalization cannot lead to the desirable result. In reality, various noise e.g. image noise, matching error and computation noise etc. can make this problem more severe, even with state-of-the-art pose estimation methods and bundle adjustment. In this paper, we deal with challenge C1 by iteratively adjustment and give rigid convergence proof based on [1]. To address challenge C2, we propose a bounding strategy to limit camera motion in each iteration.

There are several works related to our ACR methods. One of them is computational re-photography [19, 20]. Rephotography is often used for the study of history, such as urban change [21] or geological erosion [22] over time. It aims to recapture an existing photograph from the same viewpoint. However, since it cares mostly about a large scene and great changes, its relocalization accuracy is very low. Besides, existing method [20] only physically recurs 3D translation and the rotation is wrapped by a homography transformation. In contrast, we realize 6D camera relocalization with high accuracy. Another close research topic is visual servoing in robotics [2, 23, 24]. It does similar thing in restricted environments and uses manual marks as feature points and its precision highly relies on hand-eye calibration. Thus it is not applicable on a common platform and natural scene. On the contrary, our method is hand-eye calibration free and can be used in many common motion platforms.

Our major contributions are three-fold. First, we propose an active camera relocalization method with RGBD camera to dynamically and physically relocate camera into the previous pose with high accuracy. Compared with existing ACR-B, the proposed method greatly speeds up the convergence process with comparable relocalization accuracy. Second, based on [1], we give rigid convergence analysis and prove the camera pose can converge well without hand-eye calibration in an easy-to-realize condition. Third, we propose a bounding strategy for camera motion and eliminate the effect of imprecise pose estimation problem.

2. PROBLEM FORMULATION

We can express camera or hand pose by a 3D rotation $\mathbf{R} \in \mathrm{SO}(3)$ and a 3D translation $\mathbf{t} \in \mathbb{R}^3$, and it belongs to the special Euclidean group $\mathrm{SE}(3)$. In this paper, we use $\mathbf{P} \simeq \langle \mathbf{R}, \mathbf{t} \rangle$ to indicate the equivalence of the two representations, where $\mathbf{P} \in \mathrm{SE}(3)$ represents a camera or hand pose. We use $\mathbf{P}_\mathrm{A}$ and $\mathbf{P}_\mathrm{B}$ to represent camera and hand pose respectively.

In ACR, we want to dynamically and physically relocate the camera from initial pose $\mathbf{P}_\mathrm{A}^0$ to reference pose $\mathbf{P}_\mathrm{A}^\mathrm{ref}$. Correspondingly, we need to move the hand from initial pose $\mathbf{P}_\mathrm{B}^0$ to reference pose $\mathbf{P}_\mathrm{B}^\mathrm{ref}$. Following [1], we assume reference eye and hand coordinate being the world coordinate of eye and hand respectively, i.e., $\mathbf{P}_\mathrm{A}^\mathrm{ref} = \mathbf{I}$ and $\mathbf{P}_\mathrm{B}^\mathrm{ref} = \mathbf{I}$. As described above, the hand-eye relation is

$$\mathbf{P}_\mathrm{A}\mathbf{X} = \mathbf{X}\mathbf{P}_\mathrm{B}, \tag{1}$$

where $\mathbf{X} \simeq \langle \mathbf{R}_\mathrm{X}, \mathbf{t}_\mathrm{X} \rangle$ is hand-eye relative pose [16, 17]. Since physical motion is executed by hand, without known hand-eye relative pose $\mathbf{X}$ (challenge C1), we cannot obtain precise corresponding hand motion. Moreover, the estimated camera pose is not accurate (challenge C2). Thus we cannot realize one-shot camera relocation in practice. A feasible solution is gradually adjusting the camera to approach reference pose. Thus ACR will find and execute a series of "reasonable" hand motions to achieve equivalent effect of moving hand from $\mathbf{P}_\mathrm{B}^0$ to reference $\mathbf{P}_\mathrm{B}^\mathrm{ref} = \mathbf{I}$, we have

$$\mathbf{P}_\mathrm{B}^0 = \prod_{i=0}^{n} \dot{\mathbf{P}}_\mathrm{B}^i = \dot{\mathbf{P}}_\mathrm{B}^0 \dot{\mathbf{P}}_\mathrm{B}^1 \cdots \dot{\mathbf{P}}_\mathrm{B}^i \cdots \dot{\mathbf{P}}_\mathrm{B}^{n-1} \dot{\mathbf{P}}_\mathrm{B}^n, \tag{2}$$

where $\dot{\mathbf{P}}_\mathrm{B}^i$ is relative hand pose corresponding to ith motion. With hand being gradually relocated from P_B^0 to reference, we can precisely relocate the camera from initial pose P_A^0 to reference pose $\mathbf{P}_\mathrm{A}^\mathrm{ref} = \mathbf{I}$, thus the ACR is completed.

In the following, we will prove by just guessing hand-eye calibration $\mathbf{X}$ with $\dot{\mathbf{X}} = \mathbf{I}$, this iterative ACR approach will finally lead to convergence. Let us first assume we can estimate *ideal* camera pose $\mathbf{P}_\mathrm{A}^i$ at i iteration. By Eq. (1), we have *ideal* hand relative pose $\mathbf{P}_\mathrm{B}^i = \mathbf{X}^{-1}\mathbf{P}_\mathrm{A}^i\mathbf{X}$. Since $\mathbf{X}$ is unknown and we guess $\mathbf{X}$ by $\dot{\mathbf{X}} = \mathbf{I}$, then we can obtain ith

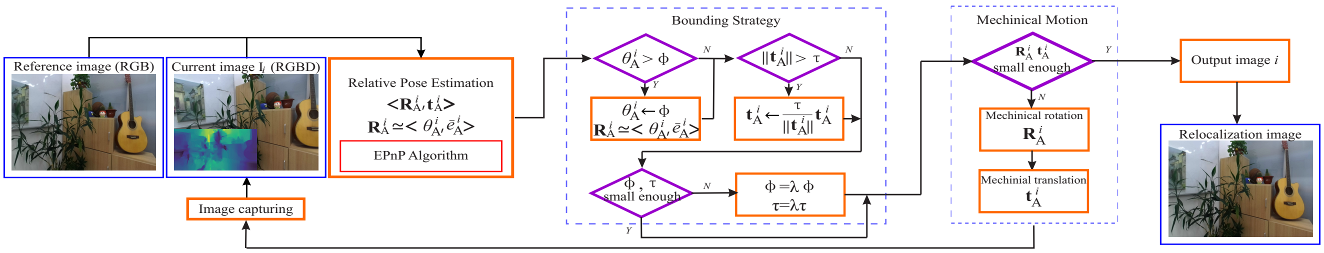


Fig. 2. Working flow of the proposed ACR algorithm.

"guessed" hand relative pose $\dot{\mathbf{P}}_{\mathrm{B}}^{i} = \dot{\mathbf{X}}^{-1}\mathbf{P}_{\mathrm{A}}^{i}\dot{\mathbf{X}} = \mathbf{P}_{\mathrm{A}}^{i}$ and corresponding hand motion $\dot{\mathbf{M}}_{\mathrm{B}}^{i} = (\dot{\mathbf{P}}_{\mathrm{B}}^{i})^{-1}$. It actually leads to ith camera motion $\dot{\mathbf{M}}_{\mathrm{A}}^{i} = \mathbf{X}\dot{\mathbf{M}}_{\mathrm{B}}^{i}\mathbf{X}^{-1} = \mathbf{X}(\mathbf{P}_{\mathrm{A}}^{i})^{-1}\mathbf{X}^{-1}$, and camera moves from $\mathbf{P}_{\mathrm{A}}^{i}$ to $\mathbf{P}_{\mathrm{A}}^{i+1}$, we have

$$\mathbf{P}_{\mathrm{A}}^{i+1} = \dot{\mathbf{M}}_{\mathrm{A}}^{i}\mathbf{P}_{\mathrm{A}}^{i} = \mathbf{X}(\mathbf{P}_{\mathrm{A}}^{i})^{-1}\mathbf{X}^{-1}\mathbf{P}_{\mathrm{A}}^{i}. \tag{3}$$

Eq. (3) gives the actual camera pose relation before and after i-th motion. We can split Eq. (3) into rotation and translation parts, which yields

$$\mathbf{R}_{\mathrm{A}}^{i+1} = \mathbf{R}_{\mathrm{X}}(\mathbf{R}_{\mathrm{A}}^{i})^{-1}\mathbf{R}_{\mathrm{X}}^{-1}\mathbf{R}_{\mathrm{A}}^{i}, \tag{4}$$

$$\begin{aligned}\mathbf{t}_{\mathrm{A}}^{i+1} &= \mathbf{R}_{\mathrm{X}}(\mathbf{R}_{\mathrm{A}}^{i})^{-1}(\mathbf{R}_{\mathrm{X}}^{-1}\mathbf{t}_{\mathrm{A}}^{i} - \mathbf{t}_{\mathrm{A}}^{i}) \\ &\quad + [\mathbf{I} - \mathbf{R}_{\mathrm{X}}(\mathbf{R}_{\mathrm{A}}^{i})^{-1}\mathbf{R}_{\mathrm{X}}^{-1}]\mathbf{t}_{\mathrm{X}}.\end{aligned} \tag{5}$$

As proved by ACR-B [1], if $\theta_{\mathrm{X}} \leq \frac{\pi}{3}$, with enough ACR iterations, the rotation $\mathbf{R}_{\mathrm{A}}^{i}$ will gradually reduce to $\mathbf{I}$. Since our translation portion is different to ACR-B, here we give strict proof of $\mathbf{t}_{\mathrm{A}}$ convergence in Theorem 1.

Theorem 1 ($\mathbf{t}_{\mathrm{A}}$ convergence)**.** *When eye relative rotation $\mathbf{R}_{\mathrm{A}}^{i}$ converges to $\mathbf{I}$, if $\theta_{\mathrm{X}} < \frac{\pi}{3}$, with enough iterations, $\|\mathbf{t}_{\mathrm{A}}^{i}\|$ finally converges to zero.*

Proof. When eye relative rotation $\mathbf{R}_{\mathrm{A}}^{i}$ converges to $\mathbf{I}$, Eq. (5) can be reduced to

$$\mathbf{t}_{\mathrm{A}}^{i+1} = \mathbf{t}_{\mathrm{A}}^{i} - \mathbf{R}_{\mathrm{X}}\mathbf{t}_{\mathrm{A}}^{i}, \tag{6}$$

Using axis-angle representation, we have $\mathbf{R}_{\mathrm{X}} \simeq \langle\theta_{\mathrm{X}}, \bar{\mathbf{e}}_{\mathrm{X}}\rangle$. Then by Rodrigues' formula,

$$\begin{aligned}\mathbf{t}_{\mathrm{A}}^{i+1} &= \mathbf{t}_{\mathrm{A}}^{i} - [\cos\theta_{\mathrm{X}}\mathbf{t}_{\mathrm{A}}^{i} + \sin\theta_{\mathrm{X}}(\bar{\mathbf{e}}_{\mathrm{X}} \times \mathbf{t}_{\mathrm{A}}^{i}) \\ &\quad + (1-\cos\theta_{\mathrm{X}})(\bar{\mathbf{e}}_{\mathrm{X}} \cdot \mathbf{t}_{\mathrm{A}}^{i})\bar{\mathbf{e}}_{\mathrm{X}}].\end{aligned} \tag{7}$$

We measure $\|\mathbf{t}_{A}^{i+1}\|$ by

$$\begin{aligned}\|\mathbf{t}_{A}^{i+1}\|^2 &= \mathbf{t}_{A}^{i+1} \cdot \mathbf{t}_{A}^{i+1} \\ &= 2\|\mathbf{t}_{A}^{i}\|^2(1-\cos\theta_{\mathrm{X}})[1 - (\bar{\mathbf{e}}_{\mathrm{X}} \cdot \bar{\mathbf{t}}_{\mathrm{A}}^{i})^2] \\ &\leq 2\|\mathbf{t}_{A}^{i}\|^2(1-\cos\theta_{\mathrm{X}}),\end{aligned} \tag{8}$$

by mathematical induction, we can get

$$\|\mathbf{t}_{A}^{n}\|^2 \leq \|\mathbf{t}_{A}^{0}\|^2[2(1-\cos\theta_{\mathrm{X}})]^n, \tag{9}$$

If $\theta_{\mathrm{X}} < \frac{\pi}{3}$, $0 \leq 2(1-\cos\theta_{\mathrm{X}}) < 1$, with enough large n, we have $[2(1-\cos\theta_{\mathrm{X}})]^n \to 0$, so $\|\mathbf{t}_{A}^{n}\|^2 \to 0$. That is, with enough iterations, $\|\mathbf{t}_{\mathrm{A}}^{i}\|$ finally converges to zero. □

3. THE METHOD

We have proved by just guessing hand-eye calibration $\mathbf{X}$ with $\dot{\mathbf{X}} = \mathbf{I}$ and if $\theta_{\mathrm{X}} < \frac{\pi}{3}$, after $\mathbf{R}_{\mathrm{A}}$ convergence, $\mathbf{t}_{\mathrm{A}}$ will finally reduce to zeros. So we can provide a strategy that adjusts the rotation to convergence in the first stage and then iteratively adjusts translation in the second stage. However, this strategy requires the mechanical rotation and translation are significantly independent. To speed up the process and get high accuracy, similar to ACR-B [1], we jointly adjust rotation and translation in each iteration. Moreover, to handle the imprecise camera pose estimation problem, we propose a bounding strategy to limit the motion to be smaller and smaller. Specifically, given an initial boundary $\langle\phi = \phi^0, \tau = \tau^0\rangle$, we reduce the boundary by a ratio of $\lambda \in (0, 1)$ until it reaches a minimal value $\langle\phi^{\min}, \tau^{\min}\rangle$, where ϕ and τ are the respective upper bound of rotation angle and translation size. Detailed working flow is shown in Fig. 2. Different from ACR-B method, the proposed method does not require $\mathbf{t}_{\mathrm{A}}$ bisection since it estimates real translation scale with RGBD image, thus efficiently improves convergence rate. The bounding strategy makes the motion more accurate over time and improves the robustness of relocalization. In practice, we set ϕ^0 and τ^0 as two times of θ_{A}^0 and $\|\mathbf{t}_{\mathrm{A}}^0\|$ from camera pose estimation and λ to be 0.8. Experiments validate the proposed method is much more efficient and its accuracy is comparable to ACR-B.

4. EXPERIMENTS

4.1. Environments setup

Experiments are conducted in both virtual environments and real world. Virtual environments are built on Unreal Engine 4 with UnrealCV [25] for interaction. It is ideal for proof-of-concept since all the data are precise with little noise, and fully controllable for repeatable tests. As for real world, we use ZED camera for RGBD image capturing and a common assembled 6D platform for motion. We use two recent methods for comparison, ACR-B [1] and ACR-H [4]. Between them, ACR-H generate homography matrix and navigation rectangles to guide user to relocate camera to reference.

All experiments use one single RGB image as reference with default resolutions (640×480 for UnrealCV and 1280×720 for ZED). The proposed method use depth information in ACR process. In virtual environment, we set $\theta_{\mathrm{X}} = 15°$ and $\mathbf{t}_{\mathrm{X}} = [5, 5, 5]$ as hand-eye displacement. In real world, hand-eye displacement is unknown and can be changed in different

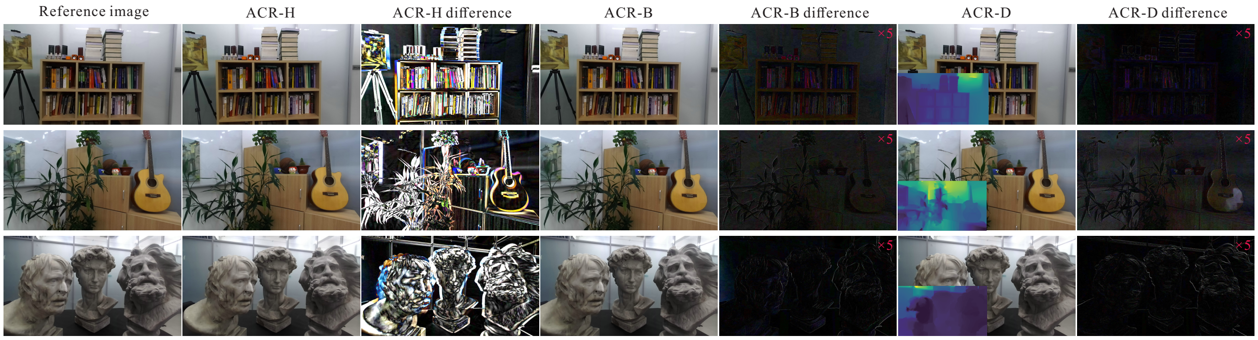
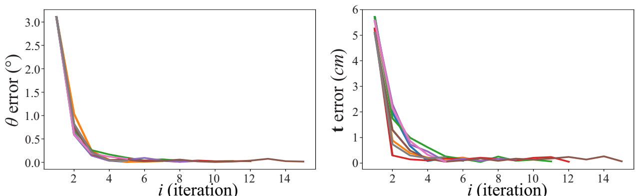
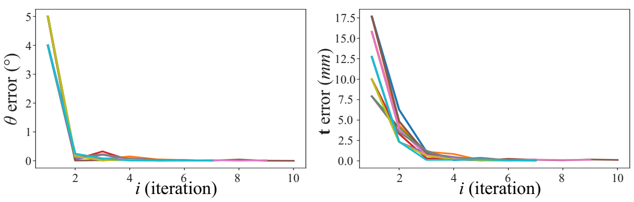


Fig. 3. Results of different ACR methods in three real-world scenes. Reference image, ACR image and difference image ($\times$ 5) with reference by each method is shown. Depth maps are shown in left-bottom of ACR-D images.



	1	2	3	4	5	6	7	8
θ error (°)	0.0387	0.0240	0.0263	0.0244	0.0284	0.0171	0.0786	0.0391
t error (*cm*)	0.0886	0.0761	0.0665	0.0533	0.1412	0.0701	0.0575	0.2326

(a) Relocalization error of ACR-D in virtual scenes.



	1	2	3	4	5	6	7	8	9	10
θ error (°)	0.0065	0.0249	0.0019	0.0121	0.0044	0.0000	0.0202	0.0011	0.0137	0.0037
t error (*mm*)	0.1645	0.0574	0.0693	0.0766	0.0621	0.1163	0.1236	0.0471	0.0540	0.0126

(b) Relocalization error of ACR-D in real world.

Fig. 4. Relocalization error of ACR-D in virtual scenes (a) and real world (b), different experiments are distinguished by colors. The table below shows the finally relocalization error.

ACR experiments. Following [1], we use AFD (average feature displacement) as criterion to evaluate ACR accuracy.

Table 1. Comparison of average AFD and variance of 3 methods.

Method	ACR-H [4]	ACR-B [1]	ACR-D
AFD (avg)	15.8317	0.7963	**0.7877**
Variance	43.3824	**0.0222**	0.0858

Table 2. Comparison of iteration number of 9 real experiments.

	Scene 1			Scene 2			Scene 3		
ACR-B	14	15	15	12	13	14	10	14	15
ACR-D	**6**	**7**	**5**	**6**	**5**	**4**	**4**	**4**	**5**

4.2. Accuracy and convergence validation

To evaluate ACR accuracy and convergence in real world, we set up 3 scenes and conduct 3 experiments for each scene (totally 9 experiments). The average AFD value and variance for each method are shown in Table. 1. Fig. 3 gives visual ACR results and difference image with reference from three different experiments. From Table. 1 and Fig. 3, we can see the proposed ACR-D has achieve sub-pixel accuracy (AFD < 1) and very robust (small variance) and its accuracy is comparable with ACR-B and much better than ACR-H.

Table. 2 shows iteration numbers used in ACR-D compared with ACR-B, from which we see the proposed ACR-D use only $\frac{1}{3.5}$ to $\frac{1}{2.1}$ iterations of that in ACR-B. Thus the proposed ACR-D methods has much faster convergence rate. From these accuracy and convergence comparison, we can draw a conclusion that the ACR-D speeds up the relocalization considerably while keeps equivalent high accuracy.

4.3. Physical relocalization accuracy evaluation

To further evaluate the physical relocalization accuracy, we conduct 8 experiments in one virtual scene and 10 experiments in one real-world scene. For virtual scene, we can get current and reference camera pose to compute relocalizaton error at each iteration. For real-world scene, since ground truth of camera pose is unknown, we use hand relocalization error as evaluation. We first move hand by $\mathbf{M}^{\mathrm{GT}}$, then conduct ACR and accumulate each hand motion till ith iteration as $\mathbf{M}^i$ and use the difference between $\mathbf{M}^{\mathrm{GT}}$ and $\mathbf{M}^i$ as hand relocalization error. Fig. 4 shows relocalization error change of different ACR process. Physical rotation errors are less than $0.04°$ in all experiments, and translation error is less than $2.40mm$ and $0.17mm$ in virtual and real-world scene. These experiments indicates that our ACR-D method is very accurate and robust in both virtual and real environments.

5. CONCLUSION

In this paper, we extend existing ACR algorithm to a more efficient one by utilizing additional depth information from RGBD camera. We rigidly prove convergence of the proposed method without hand-eye calibration and propose a bounding strategy for camera motion to reduce the estimated camera pose error. Extensive experiments verify the convergence and effectiveness of our approach. Our work can be naturally extended to benefit fine-grained change detection, visual robot manipulation and many other applications. We hope our work draws more attention of communities to the active computer vision with robotics rather than just static and passive vision.

Acknowledgement This work was supported by NSFC 61671325 and 61572354.

6. REFERENCES

[1] Wei Feng, Fei-Peng Tian, Qian Zhang, and Jizhou Sun, "6d dynamic camera relocalization from single reference image," in *CVPR*, 2016.

[2] Bernard Espiau, François Chaumette, and Patrick Rives, "A new approach to visual servoing in robotics," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 3, pp. 313–326, 1992.

[3] Yuke Zhu, Roozbeh Mottaghi, Eric Kolve, Joseph J Lim, Abhinav Gupta, Li Fei-Fei, and Ali Farhadi, "Target-driven visual navigation in indoor scenes using deep reinforcement learning," *arXiv preprint arXiv:1609.05143*, 2016.

[4] Wei Feng, Fei-Peng Tian, Qian Zhang, Nan Zhang, Liang Wan, and Jizhou Sun, "Fine-grained change detection of misaligned scenes with varied illuminations," in *ICCV*, 2015.

[5] Rui Huang, Wei Feng, Mingyuan Fan, Qing Guo, and Jizhou Sun, "Learning multi-path cnn for mural deterioration detection," *JAIHC*.

[6] Sameer Agarwal, Yasutaka Furukawa, Noah Snavely, Ian Simon, Brian Curless, Steven M Seitz, and Richard Szeliski, "Building rome in a day," *Communications of the ACM*, vol. 54, no. 10, pp. 105–112, 2011.

[7] Nianjuan Jiang, Zhaopeng Cui, and Ping Tan, "A global linear method for camera pose registration," in *ICCV*, 2013.

[8] Georg Klein and David Murray, "Parallel tracking and mapping for small ar workspaces," in *ISMAR*, 2007.

[9] Jakob Engel, Thomas Schöps, and Daniel Cremers, "Lsd-slam: Large-scale direct monocular slam," in *ECCV*, 2014, pp. 834–849.

[10] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[11] Amazon Robotics LLC, "2017 amazon robotics challenge official rules, version 3," 2017.

[12] C. Liu, J. Yuen, and A. Torralba, "SIFT flow: Dense correspondence across scenes and its applications," *IEEE TPAMI*, vol. 33, no. 5, pp. 978–994, 2011.

[13] Richard Szeliski, "Image alignment and stitching: A tutorial," *Foundations and Trends® in Computer Graphics and Vision*, vol. 2, no. 1, pp. 1–104, 2006.

[14] David Nistér, "An efficient solution to the five-point relative pose problem," *IEEE TPAMI*, vol. 26, no. 6, pp. 756–770, 2004.

[15] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua, "Epnp: An accurate o (n) solution to the pnp problem," *IJCV*, vol. 81, no. 2, pp. 155–166, 2009.

[16] Richard Hartley and Fredrik Kahl, "Global optimization through rotation space search," *IJCV*, vol. 82, no. 1, pp. 64–79, 2009.

[17] Jan Heller, Michal Havlena, and Tomas Pajdla, "A branch-and-bound algorithm for globally optimal hand-eye calibration," in *CVPR*, 2012.

[18] Jan Heller, Michal Havlena, and Tomas Pajdla, "Globally optimal hand-eye calibration using branch-and-bound," *IEEE TPAMI*, vol. 38, no. 5, pp. 1027–1033, 2016.

[19] Ruth West, Abby Halley, Daniel Gordon, Jarlath O'Neil-Dunne, and Robert Pless, "Collaborative rephotography," in *ACM SIGGRAPH Studio Talks*, 2013.

[20] Soonmin Bae, Aseem Agarwala, and Frédo Durand, "Computational rephotography," *ACM TOG*, vol. 29, no. 3, pp. 1–15, 2010.

[21] James Edward Wells II, *Western landscapes, western images: a rephotography of US Highway 89*, Ph.D. thesis, Kansas State University, 2012.

[22] Davis Guggenheim, Albert Gore, Lawrence Bender, Scott Z Burns, Laurie David, and Global Warming Documentary, *An inconvenient truth*, Paramount Classics Hollywood, CA, 2006.

[23] Gian Luca Mariottini, Giuseppe Oriolo, and Domenico Prattichizzo, "Image-based visual servoing for nonholonomic mobile robots using epipolar geometry," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 87–100, 2007.

[24] Xuebo Zhang, Yongchun Fang, and Xi Liu, "Motion-estimation-based visual servoing of nonholonomic mobile robots," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1167–1175, 2011.

[25] Weichao Qiu and Alan Yuille, "Unrealcv: Connecting computer vision to unreal engine," *arXiv preprint arXiv:1609.01326*, 2016.