Proceedings of 2017 IEEE
International Conference on Mechatronics and Automation
August 6 - 9, Takamatsu, Japan

RGB-D Camera-based Tracking System for an Amphibious Spherical Robot

Shaowu Pan[1, 2], Shuxiang Guo[1, 2, 3], Liwei Shi[1, 2, *], Ping Guo[1, 2], Huiming Xing[1, 2], Shuxiang Su[1, 2] and Zhan Chen[1, 2]
[1] Key Laboratory of Convergence Medical Engineering System and Healthcare Technology, the Ministry of Industry and Information Technology, School of Life Science, Beijing Institute of Technology, No.5, Zhongguancun South Street, Haidian District, 100081 Beijing, China.
[2] Key Laboratory of Biomimetic Robots and Systems, Ministry of Education, Beijing Institute of Technology, No.5, Zhongguancun South Street, Haidian District, 100081 Beijing, China.
[3] Faculty of Engineering, Kagawa University, 2217-20 Hayashi-cho, Takamatsu, Japan.
panshaowu@bit.edu.cn, guoshuxiang@bit.edu.cn, shiliwei@bit.edu.cn
*Corresponding author

***Abstract* – To execute vision-based tasks of the amphibious spherical robot, a visual tracking system was designed and constructed. A RGB-D camera was calibrated in the amphibious environments and was then used to perceive the surroundings of the robot. A RGB-D tracker, which was capable of handling occlusions and scale changes of the target, was built upon the KCF tracker to locate the target object. HoG and CN features in the color images were extracted to describe the target object. The scale and position models of the object was established using the Gaussian model and the depth histogram of the object. The occlusion event was recognized by segmenting the depth image and using an empirical formula. The online update process of the KCF tracker was temporarily stopped once an occlusion event was detected. Experimental results with various image sequences in amphibious environments demonstrated the effectiveness and robustness of the proposed tracking algorithm, which can meet the application requirements of the amphibious spherical robots.**

Index Terms* – *Kernelized Correlation Filtering. RGB-D Tracker. ToF Camera. Amphibious Spherical Robot. Visual Tracking.

I. Introduction

As a fundamental function of robotic vision system, visual tracking is critical important for mobile robots to realize intelligent functions such as multi-robot cooperation and autonomous navigation. Benefiting from the progresses in the pattern recognition theory and the image processing technology, numerous excellent tracking algorithms, such as TLD (Tracking-Learning-Detection) [1], STUCK (Structured output tracking with kernels) [2], Staple [3] have been proposed in recent years [4]. However, it still remains a very challenge work to handle disturbances like scale changes, illumination variations, and partial occlusion for these color-only trackers [5]. Compared with theoretical studies conducted on high performance computers, designing a practical visual tracking system for small-sized amphibious robots is an even more difficult work. On the one hand, common problems of visual tracking such as illumination variations and partial occlusions shall be especially considered in the amphibious environments. On the other hand, the carrying capacity and the information processing ability of an amphibious robot is usually very limited. Thus, the real-time performance of the system shall also be taken into considerations.

The emerging technology of RGB-D cameras provide an alternative solution to realizing robust visual tracking [6]. As a supplement to the color images, the depth images provide the position information of the object which may facilitate the tracking process. In 2013, Song *et al.* [7] proposed a baseline RGB-D tracker and release a unified benchmark captured with a Microsoft Kinect. Combing with detector built upon SVM (Support Vector Machine), the optical flow algorithm was adopted to track the object. In 2015, Awwad *et al.* [8] proposed a local depth pattern for tracking in depth videos, which was proved to be very effective in RGB-D tracking [9]. In 2015, Hannuna *et al.* [10] proposed a RGB-D tracker on the basis of the real-time KCF (Kernelized Correlation Filtering) tracker. The problems of scale changes and partial occlusions were handled using the method of image segmentation. But as far as we know, there were no relevant applications on small-sized amphibious robots.

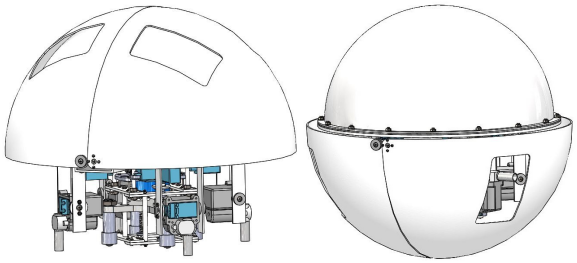
To execute vision-based tasks of our amphibious spherical robot, a RGB-D visual tracking system was designed and implemented. A Softkinetic ToF (Time-of-Flight) camera was calibrated in the amphibious environments and was then used to perceive the surroundings of the robot. A RGB-D tracker, which tracked the target object using the HoG (Histograms of Oriented Gradients) and CN (Color Name) features in the color images, was built upon the KCF tracker. The scale and position models of the object was established using the Gaussian model and the depth histogram of the object. The occlusion event was recognized by segmenting the depth image and using an empirical formula. The online update process of the KCF tracker was temporarily stopped once an occlusion event was detected. In the experimental section, the Princeton benchmark and the RGB-D images captured from the robotic platform was adopted to verify the validation of the tracker. Test results demonstrated the effectiveness and robustness of the proposed tracking algorithm, which can meet practical application requirements of the amphibious spherical robots.

The rest of this paper is organized as follows. An overview on our amphibious spherical robots and the RGB-D camera will be introduced in Section II. The global structure of the proposed

RGB-D tracker will be elaborated in Section III. The anti-occlusion method of the RGB-D tracker will be described in Section IV. Experimental results conducted on the computer and the robot platform will be provided in Section V. And Section VI will be conclusions and follow-up relevant research work.

II. Related Work and Application Requirement

A. Amphibious Spherical Robot



(a) Land mode (b) Underwater Mode

Fig. 1 Diagram of the improved amphibious spherical robot

In 2012, our research team proposed a novel amphibious spherical robot for precise and stealthy applications in littoral regions [11]. As shown in Fig. 1, the upper hemisphere was a waterproof cabin, in which the electronic devices and the sensors were installed. Four hybrid driving units or legs were symmetrically installed inside the lower hemisphere which consisted of two openable quarter-sphere shells [12]. In the terrestrial environment, the robot crawled with the four legs. In the underwater environment, it swam with water jets which were fixed to the legs. Unlike most existing amphibious spherical robots which moved by rolling, the robot was able to work in complex environments like coral reefs and pipelines [13].

In 2016, a latest version of the amphibious spherical robot with a larger diameter of 350 mm was designed and fabricated. As shown in Fig. 2, on the basis of the prototype robot, various sensors, including a global positioning system (GPS) module, a depth gauge, a micro-electromechanical system (MEMS) inertial measurement unit (IMU), an industrial camera, and a ToF camera, were added to sense the status of the robot and the surroundings. Considering that the load space of the robot was narrow and enclosed, the information processing system of the robot was fabricated using a Xilinx Zynq-7000 SoC (XC7Z020) and a Raspberry Pi 3 to reduce the power consumption and the heat generation [14]. On this basis, robotic visual algorithms including mixture Gaussian model-based foreground detection and compressive tracking were implemented on the embedded system. And a SoC-based heterogeneous computing architecture was designed to speed up image processing.

Due to the characteristics of the robot and the special application environments, designing a practical tracking system for the amphibious spherical robot remains a very challenge work. First, the adopted tracking algorithm should be robust to common disturbing factors like partial occlusions because the mobile robot would work under diverse amphibious scenarios [15]. Second, the real-time performance and the tracking accuracy of the system should be acceptable to meet the requirements of robotic applications [16]. Third, the problems of color distortion, under-exposure and fuzz, which are caused by light absorption and scattering in underwater environments, should be taken into account in the underwater tracking applications [17].

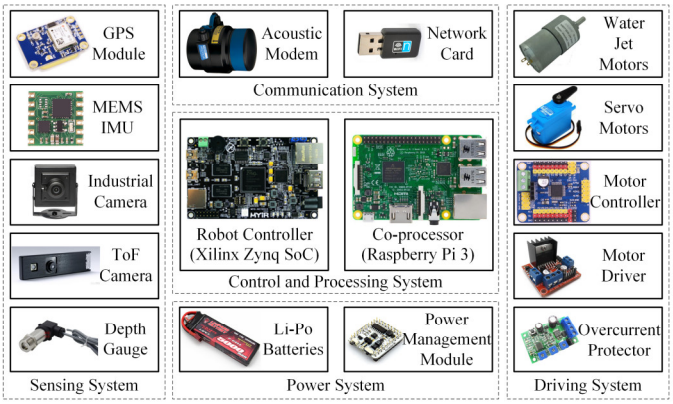
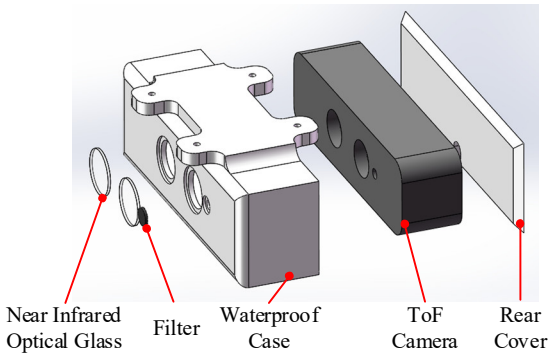


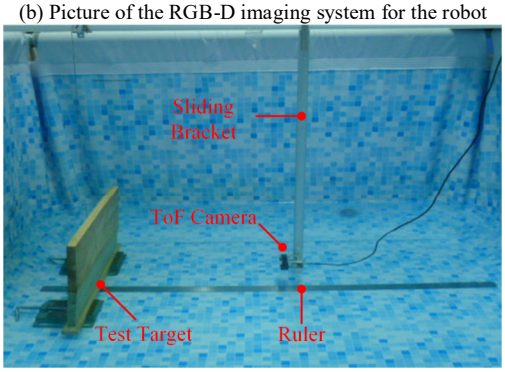
Fig. 2 Diagram of the electronic system of the amphibious spherical robot

B. RGB-D Camera and Calibration



(a) Waterproof cover for the Softkinetic DS525 ToF camera

(b) Picture of the RGB-D imaging system for the robot

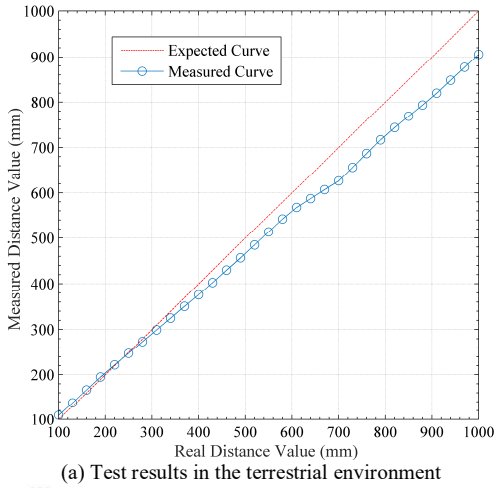


(c) Picture of the Calibration experiment

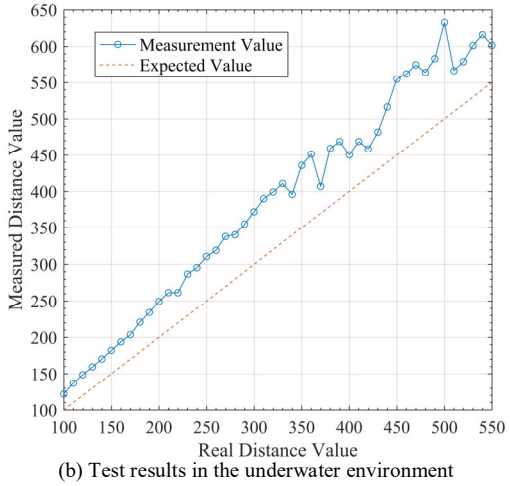
Fig. 3 Calibration experiment for the Softkinetic DS525 ToF camera

In recent years, the increasing low cost RGB-D cameras provided a novel solution to designing a robust tracking system for small-sized amphibious robots [6]. In addition to the conventional visual RGB information, the RGB-D cameras provide the complementary depth information of objects in the view field which could be used to boost the visual tracker [9]. In comparison with RGB-D cameras using structured light, the ToF

cameras acquired the three dimensional data by measuring the phase difference in the reflected light. Thus, a TOF camera was relatively robust to interference from ambient light, it has application potential in 3D reconstruction in outdoor and underwater environments [18]. Moreover, there is a local minimum in the optical absorption of water around the wavelength of 800 nm, and the working length of the Softkinetic ToF camera is 825 nm, making it suitable for underwater applications [19]. As shown in Fig. 3 (a) and (b), a Softkinetic DS525 RGB-D camera, which captures the 720p color images and 320×240 depth image at 25–60 frames per second (fps), was adopted as the detection equipment of the amphibious spherical robot. A customized waterproof cover was machined using the 3D printing technology, and three pieces of optical glasses were fixed in front of the camera lens and the laser emitter. Given that the parameters of the light path have been changed, a calibration experiment was conducted as shown in Fig. 3 (b). The ToF camera was fixed on a sliding bracket, and a lusterless slab was served as a target to be imaged. The distance between the camera and the target was adjusted according to a ruler and was then compared with the measured value in the depth image.



(a) Test results in the terrestrial environment



(b) Test results in the underwater environment

Fig. 4 Calibration results of the Softkinetic DS525 ToF camera

As shown in Fig. 4, the ToF camera could work normally at the range of 100–1000 mm and 100–550 mm in the terrestrial and underwater environment, respectively. Because the water absorbs the near infrared laser emitted by the camera, the effective working range of the depth sensor was shorten in the underwater environment. The response of the depth camera was appropriate linear, and the captured depth image should be calibrated according to the test results.

III. KCF-BASED RGB-D TRACKER

A. *Kernelized Correlation Filtering Tracking*

Considering the practical application requirements of the amphibious spherical robot, it is essential to make a compromise between the processing precision and the real-time performance when designing the robotic vision system. Recently, a lot of real-time tracking algorithms such as Staple [3], L1-tracker [4], and FCT [20] have been successively proposed. However, the robustness and long-term tracking performance of these famous algorithms were not so ideal due to drift probable [21]. In 2015, Henriques *et. al* [22] proposed the KCF algorithm which had high tracking precision and excellent real-time performance, laying the foundation on designing a RGB-D tracker for robotic applications. As shown in Fig. 5, the algorithm flows of KCF included three phases: training, detection, and updating.

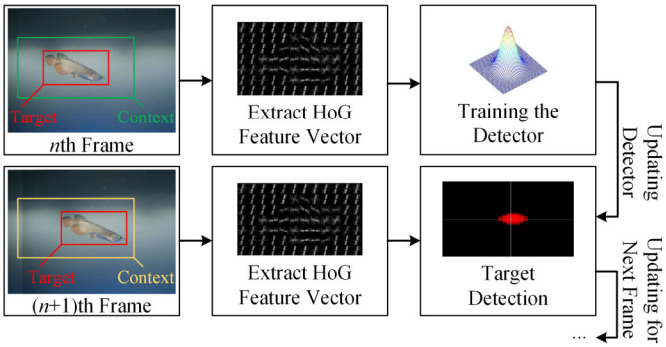


Fig. 5 Diagram of the KCF- tracker

KCF realized the process of tracking-by-detection using the rigid regression, in which a function $f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\mathbf{x}$ was tried to found that minimized the squared error over samples $\mathbf{x}_i$ and the corresponding regression target y_i:

$$\min_{\mathbf{w}} \sum_i (f(\mathbf{x}_i) - y_i)^2 + \lambda \|\mathbf{w}\|^2, \tag{1}$$

where λ is the regularization parameter. The solution of this optimization problem was:

$$\mathbf{w} = (\mathbf{X}^{\mathrm{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y}, \tag{2}$$

where $\mathbf{X}$ is a data matrix which contains all the samples. As to the tracking application, $\mathbf{x}_i$ represents the HoG feature vector of an image patch or a context sample, and y_i represents the degree of similarity between $\mathbf{x}_i$ and the target.

KCF accelerates the calculation process of $\mathbf{w}$ by utilizing the properties of circulant matrices. In the training phase, a $n \times n$ circulant matrix $\mathbf{X}$ was constructed on the basis of the $n \times 1$ target feature vector $\mathbf{x}$ using a cyclic shift operator:

$$\mathbf{X} = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ x_n & x_1 & \cdots & x_{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & \cdots & x_1 \end{bmatrix}, \tag{3}$$

Then the equation (2) could be reformulated as:

$$\mathcal{F}(\mathbf{w}) = \frac{\mathcal{F}^*(\mathbf{x}) \otimes \mathcal{F}(\mathbf{y})}{\mathcal{F}^*(\mathbf{x}) \otimes \mathcal{F}(\mathbf{x}) + \lambda}, \tag{4}$$

where $\otimes$ represents the element-wise multiplication, and $\mathcal{F}(\cdot)$ the Fourier transformation. Thus, the solving process of $\mathbf{w}$ could be efficiently realized using the fast Fourier transformation (FFT). And the computational complexity equals to $O(n\log n)$. KCF is a state-of-the-art high-speed tracking algorithm which successfully combines the characteristics of spatial domain and frequency domain. Several relevant improved algorithms have been proposed by combining multiple image features [23], designing part-based tracking mechanism [24], and adding a scale estimator [25].

B. *Structure of KCF-based RGB-D Tracker*

KCF is robust to interference factors such as background cluttering and pose changes. However, it is not capable of coping with scale changes and occlusions. It is a feasible solution to enhance the robustness of the KCF tracker by utilizing the depth data of the target without losing the real-time ability. The proposed RGB-D tracking algorithm was as shown in Fig. 6. The RGB images were input into an improved KCF tracker to realize conventional visual tracking. And the depth images were segmented to acquire the contour of the target and the average distance between the target and the camera which would be used for occlusion detection and scale estimation. A particle filter based on sequential importance sampling was adopted to predict the position of the target and to sample image patches to be detected [26], which would improve the adaptability of the tracker to fast moving targets.

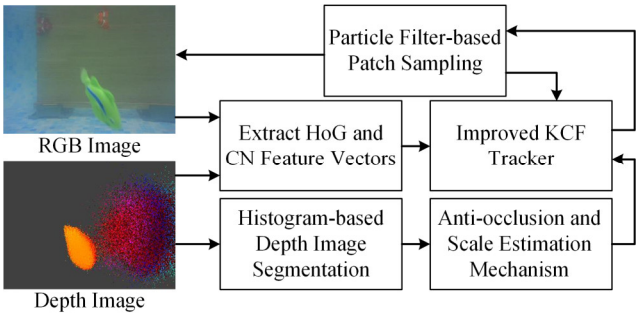


Fig. 6 Diagram of the KCF-based RGB-D tracker

On the basis of the original KCF tracker, two aspects of improvement were made to further increase the tracking precision. First, considering that the HoG feature only describe the texture of the target object, the CN feature was extracted to describe the color distribution using the same cell size as the HoG feature. These two feature vectored were simply combined to construct $\mathbf{x}_i$:

$$\mathbf{x}_i = \{\mathbf{x}_{i,\mathrm{HoG}}; \mathbf{x}_{i,\mathrm{CN}}\}. \tag{5}$$

Second, the original KCF tracker adopts a static updating mechanism, which may lead to the drift problem when the current tracking result is not so ideal. So an adaptive updating mechanism was adopted in the improved tracker for the better long-term tracking performance. The quality of the tracking result at the i th frame was described as [24]:

$$w_i = PSR_i + \frac{\eta}{SCCM_i}. \tag{6}$$

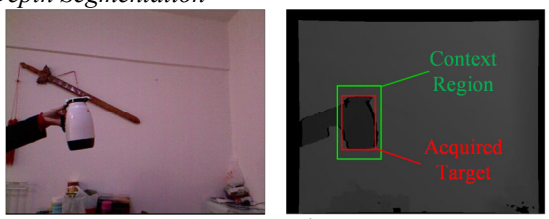
$$PSR_i = \frac{\max(\hat{f}_i) - \mu_i}{\sigma_i}. \tag{7}$$

$$SCCM_i = \left\| \hat{f}_i - \hat{f}_{i-1} \oplus \Delta \right\|_2^2. \tag{8}$$

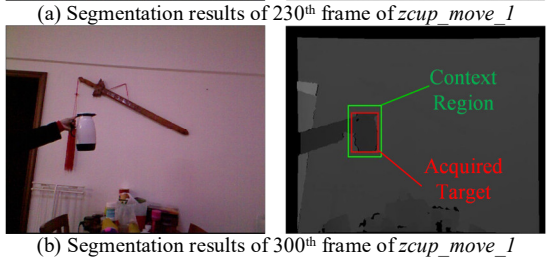
where $\hat{f}_i$ represents the confidence map at the i th frame, μ_i and σ_i the mean value and the standard deviation of $\hat{f}_i$, $\oplus$ the two dimensional shift operation of a matrix, and Δ the shift of maximum value in confidence maps from the (i–1) th frame to the i frame. PSR_i described the confidence level of the detection result, and $SCCM_i$ described the continuity of the target trajectory. Once the value of w_i was lower than the predefined threshold value, the tracker would stop updating to avoid polluting the parameters of the kenerlized correlation filter.

IV. Anti-occlusion and Scale Estimation Mechanism

A. *Depth Segmentation*



(a) Segmentation results of 230th frame of *zcup_move_1*



(b) Segmentation results of 300th frame of *zcup_move_1*

Fig. 7 Picture of the depth segmentation

The anti-occlusion and scale estimation mechanisms were designed on the assumptions that the position and the scale of the target change slowly and continuously. Because the RGB-D camera captured RGB and depth images at 30 fps, the assumptions should hold in most scenarios.

The process of depth segmentation is as follows. First, the histogram of the context region in the depth image was calculated. Second, the information entropy of the depth histogram was calculated to describe the richness of the details:

$$H = \sum_{i=1}^{L} h_i \log h_i, \tag{9}$$

where h_i represents an element of the depth histogram. Third, the depth histogram was clustered using the Meanshift algorithm, the bandwidth of which was set to be proportional to the information entropy. Fourth, the clustering results of the depth histogram was mapped to the context region. If there is no occlusion occurring, the target shall be in the foreground and have the minimum depth value on average. As shown in Fig. 7, the fraction of the target region inner the context region could be determined.

B. *Anti-occlusion and Scale Estimation Mechanisms*

On the basis of depth segmentation, the occlusion event could be detected by analyzing the depth distribution of the context region. The occlusion event was thought to have taken place if

$$\begin{cases} \varphi(\text{Area}_{\text{target}}^{i}) - \varphi(\text{Area}_{\text{target}}^{i-1}) < k\sigma^{i-1} \\ \max(\hat{f}_i) > \hat{f}_{\text{threshold}} \\ w_i > w_{\text{threshold}} \end{cases}, \tag{10}$$

where $\varphi(\text{Area}_{\text{target}}^{i})$ represents the average depth value of the target at the i th frame, and σ^{i-1} the standard deviation of the depth value at the $(i-1)$ th frame. The average depth value of target was modeled using a Gaussian model, the parameter of which was updated online. Once the average depth value suddenly changed, and the tracking result of the RGB KCF tracker deteriorated, the occlusion event was tend to happen. Then, the KCF tracker would stop updating parameters and try to redetect the target near the occluding object.

As to the scale estimation mechanism, the contour of the target object was located after the depth segmentation operation. To avoid instability of the tracker, the scale value of the object was adjusted with a small step value of 0.05. And the adjustment range was from 0.5 to 2.0. The adjusted scale value was adopted only when the values of w_i and $\max(\hat{f}_i)$ were relatively high.

V. Experimental Results

A. *Experiments with Benchmark Sequences*

(a) Test results of *bear_front*

(b) Test results of *face_occ5*

(c) Test results of *new_ex_occ4*

(d) Test results of *zcup_move_1*

Fig. 8 Evaluation results of the RGB-D benchmark sequences

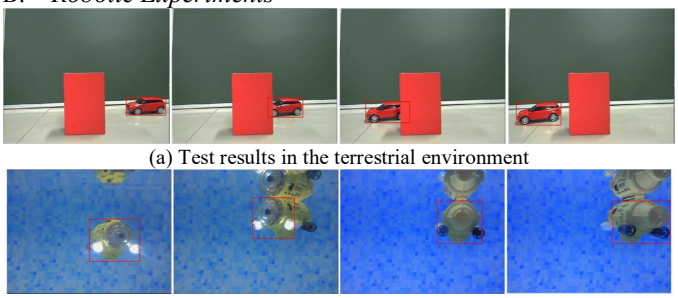
To verify the validation of the proposed RGB-D tracker, two phases of experiments were conducted. In the benchmark experiment phase, four RGB-D image sequences (*bear_front*, *face_occ5*, *new_ex_occ4*, and *zcup_move_1*) of Princeton Tracking Benchmark [7] were adopted. Fig. 8 shows the test results. TABLE I shows the quantitative evaluation results of the benchmark experiment. The results indicated that the robustness to drift, occlusion and background clutter of the proposed algorithm was reinforced by adopting the scale estimation and the occlusion detection mechanisms. As shown in Fig. 8 (a), (b) and (c), the proposed tracker was capable of precisely tracking the target object, while the KCF tracker and the CT tracker lost the target when the occlusion event occurred. But the proposed tracker could recover the tracking process using the adaptive updating and redetection mechanism. As shown in Fig. 8 (d), the proposed tracker was able to segment the target region using the depth information and then adjust the scale value. The tracking precision was improved using the scale estimation mechanism. In comparison, the CT tracker and the KCF tracker, which are not capable of adapting scale changes, could only provide the relative low tracking precision and high center errors.

TABLE I

Success Rate and Centre Location Error of the Proposed Algorithm

Sequences	Proposed		CT		KCF		Frames
	SR	CLE	SR	CLE	SR	CLE	
bear_front	83.1	12.4	16.6	149	20.4	192	281
face_occ5	99.1	4.7	52.2	65.5	99.1	4.7	330
new_ex_occ4	86.7	17.1	56.3	88.7	54.2	94.2	51
zcup_move_1	90.7	5.6	82.7	13.9	82.7	6.3	370
terrestrial	94.2	8.7	25.8	36.2	62.3	28.8	178
underwater	84.5	10.2	28.1	57.9	76.1	17.9	121

B. *Robotic Experiments*



(a) Test results in the terrestrial environment

(b) Test results in the underwater environment

Fig. 9 Evaluation results on the amphibious spherical robot

In the robotic experiment section, the water repellent ToF was installed at the amphibious spherical robot. A small car and a submarine model were adopted as the target object in the territorial and underwater environment, respectively. Fig. 9 shows the test results and TABLE I provided the quantitative evaluation results. The test results indicated that the proposed tracking system was capable of processing moving target in the amphibious environment. In the terrestrial environment, the target (a small car) was occluded by a shelter with the similar color, which resulted in the drift problem when adopting the CT and KCF tracker. In the underwater environment, the tracking process towards the target (a submarine model) was even more challenge due to the interference factors of the pose changes and the inverted reflection in water. But the proposed tracking system successfully neglected these disturbances and provided the precise trajectory of the submarine model.

VI. Conclusions and Future Works

Aiming at autonomous exploration and navigation tasks of the amphibious spherical robot, a RGB-D tracking system was

designed and implemented in this paper. A ToF camera was adopted and calibrated to perceive the surroundings of the robot. A RGB-D tracker was built on the basis of the improved KCF tracker which adopted the HoG-CN features and an adaptive updating mechanism. The depth images was segmented to realize occlusion detection and scale adjustment. Experimental results with various image sequences demonstrated the robustness of the proposed tracking system to environmental disturbances such as occlusions and scale changes, which can meet practical application requirements of the amphibious spherical robots.

The RGB-D tracking system also have some drawbacks. The proposed tracking algorithm was currently implemented using MATLAB R2014a. The average frame rate was only 2.6 fps, which was not suitable for real-time robotic applications. We will try to rewrite the algorithm using C/C++ and optimize the code using the heterogeneous computing technology to ensure the real-time performance. Moreover, a more delicate occlusion detection mechanism is also necessary to further improve the long-term tracking performance of the algorithm.

ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China (61503028, 61375094), and Excellent Young Scholars Research Fund of Beijing Institute of Technology (2014YG1611). This research project was also partly supported by National High Tech. Research and Development Program of China (No.2015AA043202).

REFERENCES

[1] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-Learning-Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.

[2] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: Structured output tracking with kernels," in *Proceedings of 2011 IEEE International Conference on Computer Vision (ICCV 2011)*, pp. 263–270, Barcelona, Spain, 2011.

[3] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: Complementary Learners for Real-Time Tracking," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016),* pp. 1401–1409, Las Vegas, Nevada,USA, 2016.

[4] B. Chenglong, W. Yi, L. Haibin, and J. Hui, "Real time robust L1 tracker using accelerated proximal gradient approach," in *Proceedings of 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012)*, pp. 1830–1837, Providence, USA, 2012.

[5] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. V. D. Hengel, "A survey of appearance models in visual object tracking," *ACM Trans. Intell. Syst. Technol.*, vol. 4, no. 4, pp. 1–48, 2013.

[6] M. Clark, D. Feldpausch, and G. S. Tewolde, "Microsoft kinect sensor for real-time color tracking robot," in *Proceedings of 2014 IEEE International Conference on Electro/Information Technology (EIT)*, pp. 416–421, Milwaukee, USA, 2014.

[7] S. Song and J. Xiao, "Tracking Revisited Using RGBD Camera: Unified Benchmark and Baselines," in *Proceedings of IEEE International Conference on Computer Vision*, pp. 233-240, Sydney, Australia, 2013.

[8] S. Awwad, F. Hussein, and M. Piccardi, "Local Depth Patterns for Tracking in Depth Videos," in *Proceedings of the 23rd ACM International Conference on Multimedia*, pp. 1115–1118, Brisbane, Australia, 2015.

[9] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced Computer Vision with Microsoft Kinect Sensor A Review," *IEEE Transactions on Cybernetics*, vol. 43, no. 5, pp. 1318–1334, 2013.

[10] M. Camplani, S. Hannuna, M. Mirmehdi, D. Damen, A. Paiement, L. Tao, and T. Burghardt, "Real-time rgb-d tracking with depth scaling kernelised correlation filters and occlusion handling," in *Proceedings of the British Machine Vision Conference (BMVC)*, pp. 145.1–145.11, Swansea, UK, 2015.

[11] S. Guo, S. Mao, L. Shi and M. Li, "Development of an amphibious mother spherical robot used as the carrier for underwater microrobots," in *Proceedings of 2012 ICME International Conference on Complex Medical Engineering (CME)*, pp. 758–762, Kobe, 2012.

[12] Y. He, L. Shi, S. Guo, S. Pan, and Z. Wang, "Preliminary mechanical analysis of an improved amphibious spherical father robot," *Microsystem Technologies*, vol. 22, no. 8, pp. 2051–2066, 2016.

[13] M. Li, S. Guo, H. Hirata and H. Ishihara, "Design and Performance Evaluation of an Amphibious Spherical Robot," *Robotics and Autonomous Systems*, vol. 64, pp. 21–34, 2014.

[14] S. Guo, S. Pan, L. Shi, P. Guo, Y. He and K. Tang. "A low-power SoC-based moving target detection system for amphibious spherical robots," *Sensors*, vol. 17, no. 4, pp. 1–21, 2017.

[15] Y. Wu, J. Lim and Yang, M, "Object tracking benchmark," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 9, pp. 1834–1848, 2015.

[16] A. W. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan and M. Shah, "Visual Tracking: An Experimental Survey," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 36, no. 7, pp. 1442–1468, 2014.

[17] X. Fu, P. Zhuang, Y. Huang and Y. Liao, "A retinex-based enhancing approach for single underwater image," in *Proceedings of IEEE International Conference on Image Processing*, pp. 4572–4576, Paris, France, 2014.

[18] G. Alenyà, S. Foix, and C. Torras, "Using ToF and RGBD cameras for 3D robot perception and manipulation in human environments," *Intelligent Service Robotics*, vol. 7, no. 4, pp. 211–220, 2014.

[19] C. L. Tsui, D. Schipf, K. R. Lin, J. Leang, F. J. Hsieh and W. C. Wang, "Using a Time of Flight method for underwater 3-dimensional depth measurements and point cloud imaging", in *Proceedings of OCEANS 2014 – TAIPEI*, pp. 1–6, Taipei, Taiwan, 2014.

[20] K. Zhang, L. Zhang and M.-H. Yang, "Fast compressive tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 10, pp. 2002–2015, 2014.

[21] Q. Liu, X. Zhao and Z. Hou, "Survey of single-target visual tracking methods based on online learning," *IEEE Transactions on Computer Vision*, vol. 8, no. 5, pp. 419–428, 2014.

[22] J. F. Henriques, R. Caseiro, P. Martins and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.

[23] Y. Ruan and Z. Wei, "Extended kernelised correlation filter tracking," *Electronics Letters*, vol. 52, no. 10, pp. 823–825, 2016.

[24] T. Liu, G. Wang and Q. Yang, "Real-time part-based visual tracking via adaptive correlation filters," in *Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4902–4912, Boston, MA, USA, 2015.

[25] C. Ma, X. Yang, C. Zhang and M. H. Yang, "Long-term correlation tracking," in *Proceedings of 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5388–5396, Boston, MA, USA, 2015.

[26] N. Wang, J. Shi, D.-Y. Yeung, and J. Jia, "Understanding and Diagnosing Visual Tracking Systems," in *Proceedings of 2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 3101–3109, Santiago, Chile, 2015.