CAMERA OSCILLATION PATTERN FOR VSLAM: TRANSLATIONAL VERSUS ROTATIONAL

Mohamed Heshmat[a], Mohamed Abdellatif[b], Kazuaki Nakamura[c], A.A.Abouelsoud[a,d] and Noboru Babaguchi[c]

[a] Egypt-Japan University of Science and Technology, Alexandria, Egypt.
[b] Future University, Cairo, Egypt.
[c] Osaka University, Osaka, Japan.
[d] Cairo University, Cairo, Egypt.

ABSTRACT

Visual SLAM algorithms exploit natural scene features to infer the camera motion and build a map of the environment landmarks. SLAM algorithm has two interrelated processes localization and mapping. For accurate localization, we need the features location estimates to converge quickly. On the other hand, to build an accurate map, we need accurate localization. Recently, a biologically inspired approach exploits deliberate camera oscillation to improve the convergence speed of depth estimate. In this paper, we explore the effect of camera oscillation pattern on the accuracy of VSLAM. Two main oscillation patterns are used for distance estimation: translational and rotational. Experiments, using static and moving robot, are made to explore the effect of these oscillation patterns on the VSLAM performance.

Index Terms— Visual SLAM, Deliberate Camera Oscillation, Translational Oscillations, Rotational Oscillations.

1. INTRODUCTION

For autonomous vehicle exploring unknown environments, the ability to localize itself and build a map of the surrounding environment is essential. The two tasks are coupled in such a way so as to benefit each other; a good localization is required to create good map, and a good map is necessary for localization. Visual SLAM is the Simultaneous Localization And Mapping using visual information obtained from a single [3, 4], stereo camera [9, 16], or multicamera [12, 17]. Visual SLAM is getting much popularity recently for its affordable cost compared to laser range finder. For better VSLAM performance, the robot need to know, quickly and efficiently, the distance to the landmarks around him, as a result the robot can know where he is. Landmarks distance estimation is not a naive problem and even it is more difficult using monocular camera. Monocular cameras, which work as time stereo, need

enough physical clues to get the landmarks distances, especially, in the forward and curved robot motion [15]. Camera oscillation was proposed as a solution for the lower physical clues case like the forward and the curved robot motion [5]. This can be regarded as an improvement in camera geometry for a time stereo, since we do not control main motion (robot motion), but we can control a secondary motion (camera oscillations).

In contrast to the current direction of image stabilization or anti-shaking camera systems, camera oscillations showed better performance for the VSLAM and improved the features convergence and the robot localization. Camera oscillation means to superimpose oscillations on the camera / robot main motion, in a lateral direction to give sufficient physical clues for the solution [5]. This idea stems from the well-known fact that human and several animals use oscillation to improve the distance estimation. Retinal image motion and motion parallax generated with the aid of head movements is used as a means of distance estimation. Camera oscillation had been used for edge detection, contrast detection and motion analysis [6, 11]. Recently, a camera which moves in a circular path about the optical axis was used for removing motion blur [1].

In nature, objects distance estimation is an important characteristic for humans and animals and the praying mantis is a good example [7]. Praying mantis peer its head left and right to obtain depth perception. Mantis uses a head-peering method during the hunt for static prey from ambush, or when presented with water or air gap barrier before leaping across the barrier [2]. This strategy is used instead of depth recovery using stereopsis, and likely gives a much better estimate of distance [10]. The velocity and maximum amplitude of peering movements executed immediately before a jump are related to the distance to a stationary target object. Before jumping a mantis normally increases the amplitude of peering movements with increasing distance to the target. In contrast, the peering velocity is generally kept constant, independently of the target distance. However, if objects are located beyond the jumping range, peering velocity may also increase
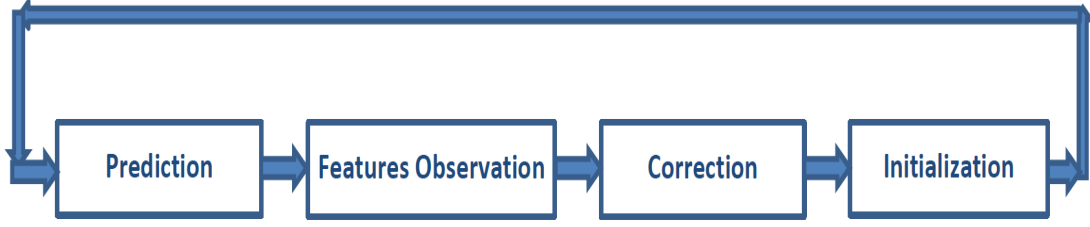

Fig. 1: SLAM block diagram

with distance, and the velocity and amplitude may then be decreased again for objects that are quite far away [8].

It has been observed that, during oscillation, the head is moved in either a translational or rotational manner in a horizontal plane. These two patterns used by several animals for distance estimation [7]. Fig. 2 shows the translatory and rotational patterns of the Mantis. Mantis has two mutually independent systems of motion, and can at any moment switch from one to the other [7]. Thus, the robot eye (the camera) could oscillate from side to side in translational or rotational pattern, like the Mantis, to estimate features distances. Our argument is to answer the question: which oscillation pattern is better for VSLAM for the fast convergence of the features and accurate localization.

This paper is organized as follows: section 2 outlines the basic probabilistic framework for visual SLAM algorithm. Then, in section 3, the camera oscillation models are presented. The simulation results and discussion are described in section 4. Conclusions are finally given in section 5.

2. VISUAL SLAM

Extended Kalman Filter-based SLAM (EKF-SLAM) is used as the core probabilistic framework. The EKF state vector generally contains robot and landmarks states, which are modeled as a Gaussian variable using the mean vector and the covariance matrix of the state vector. This state vector is maintained using EKF through a prediction/correction loop. In our case, the sensor state is added to the state vector therefore the system can track the camera motion including camera oscillation. Fig. 1 shows SLAM algorithm block diagram. First, SLAM Algorithm predicts of the next robot pose using the motion model and modifies the state vector and the covariance matrix according to that. Then using the sensor, in our case the camera, it can observe the features which had seen before. Using observed features, the SLAM algorithm corrects the state vector and the covariance matrix. At last, the algorithm initializes new features and starts all over again.

The state vector can be described as follows:

$$x = \begin{bmatrix} R & S & M \end{bmatrix}^T = \begin{bmatrix} R & S & L_1 & \dots & L_n \end{bmatrix}^T \tag{1}$$

where R is the robot state, S is the sensor state and M is the map landmarks state. The covariance matrix is defined as:

$$P = \begin{bmatrix} P_{rr} & P_{rs} & P_{rm} \\ P_{sr} & P_{ss} & P_{sm} \\ P_{mr} & P_{ms} & P_{mm} \end{bmatrix} \tag{2}$$

The robot position represented by X-Y-Z position and the robot orientation represented by quaternion q construct the robot state vector. The landmarks state vector represents all the landmark positions. The Euclidean parameterization or anchored homogeneous parameterization is used to represent the landmarks based on the landmark linearity check [14].

$$R = \begin{bmatrix} X_r & Y_r & Z_r & q_{r1} & q_{r2} & q_{r3} & q_{r4} \end{bmatrix}^T \tag{3}$$

$$S = \begin{bmatrix} X_s & Y_s & Z_s & q_{s1} & q_{s2} & q_{s3} & q_{s4} \end{bmatrix}^T \tag{4}$$

The EKF prediction step:

$$x \leftarrow f(x, \ u, \ n) \tag{5}$$

$$P \leftarrow F_x P F_x{}^T + F_n N F_n{}^T \tag{6}$$

where u is the control vector, $f()$ is the generic time update functions, F_x and F_n are the Jacobian matrices and N is the covariance matrix of the perturbation n.

The EKF correction step:

$$z = y - h(x) \tag{7}$$

$$Z = H_x P H_x{}^T + W \tag{8}$$

$$K = P H_x{}^T Z^{-1} \tag{9}$$

$$x \leftarrow x + Kz \tag{10}$$

$$P \leftarrow P - KZK^T \tag{11}$$

where y is the noisy measurement, x is the full state, $h()$ is the observation function, H_x is Jacobian matrix, W is the covariance matrix of the measurement noise and K is the Kalman gain. Equations 7 and 8 are the innovation mean and covariance matrix.

3. CAMERA OSCILLATION MODELS

Let (X_s, Y_s) be the camera position in the robot-centered coordinate system. As shown in Fig. 3 translational oscillation is modeled as

$$X_s = A_T.\sin(2\pi F_T t) \tag{12}$$

$$Y_s = d \tag{13}$$

where A_T is the oscillation amplitude, F_T is the oscillation frequency and t is the time.

On the other hand, as shown in Fig. 3 rotational oscillation is modeled as

$$X_s = d.\sin(\Theta.\sin(2\pi F_R t)) \tag{14}$$

$$Y_s = d.\cos(\Theta.\sin(2\pi F_R t)) \tag{15}$$

$$\Theta = \sin^{-1}(A_R/d) \tag{16}$$

where A_R is the oscillation amplitude, F_R is the oscillation frequency and t is the time.

4. SIMULATION RESULTS

The objective of these experiments is to explore the effect of both translational and rotational patterns on VSLAM performance. We made experiments in case of a robot in both static and moving cases. The motion types are forward, curved and lateral. The simulations are made using the public EKF-SLAM toolbox [13]. The robot is simulated in an environment of landmarks in an 16 m x 16 m area. This environment contains 162 landmarks configured into two layers of landmarks. The robot receives noisy control inputs which are used in the prediction stage, and one noisy image per control step. Table 1 summarizes the simulation parameters for the sensor, the robot and the landmarks.

In each experiment, we check the localization accuracy using the two oscillation pattern in case of changing the oscillation parameters (amplitude and frequency). Experiments are made using Monte Carlo method. We perform 25 runs using different random seed for the process noise and the measurement noise. At each run, we use the root mean square error (RMSE) of the localization parameters and use the Euclidean distance of errors in X and Y to represent the position
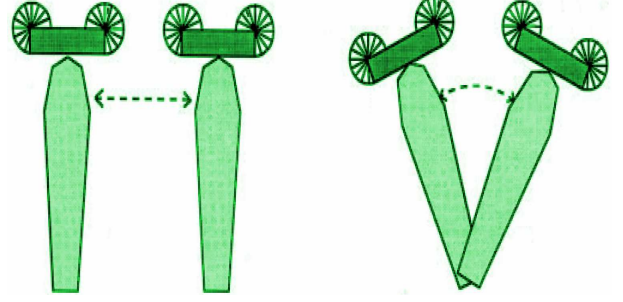

Fig. 2: Mantis oscillation patterns. Left: the translational one and right: the rotational one [7]
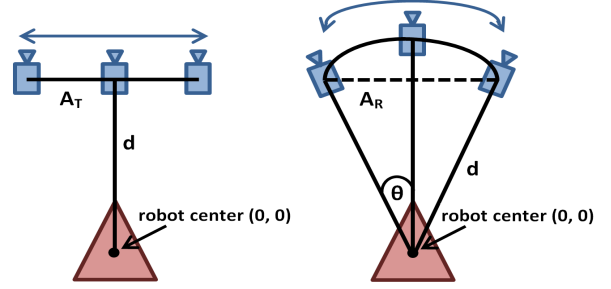

Fig. 3: Camera oscillation models. Left: the translational one and right: the rotational one

error. After all runs, the mean of all parameters is used for evaluation. Two parameters were checked through the experiments, oscillation amplitude ranging from 1 to 20 cm and oscillation frequency ranging from 1 to 4 Hz and the localization errors were recorded for all experiments.
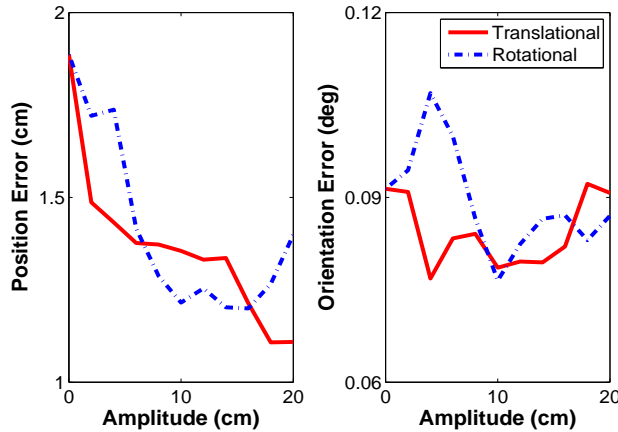
4.1. Static Robot Testing

At first, we wish to explore the oscillation patterns independently of the robot motion. Therefore, we examine the two camera oscillations patterns while the robot is static for a period of time and 80 frames were captured in each experiment. The results are shown in Fig. 4.

In the oscillation amplitude results, the translational oscillation pattern showed decreased localization errors than the rotational one. For the frequency experiments, the rotational oscillation showed better performance in the position error, but the translational oscillation showed better performance in the orientation error.
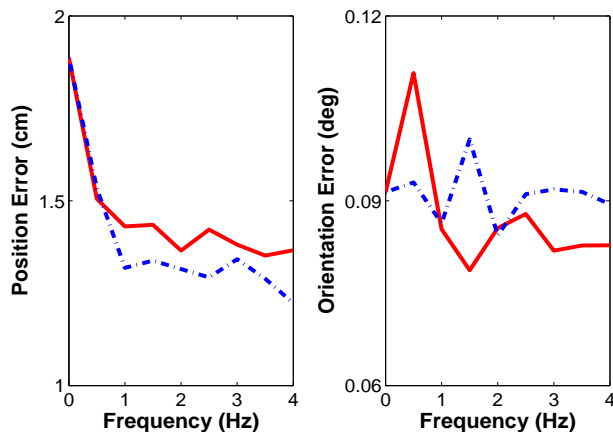
Using rotational oscillation, the camera can see a wide range of view, which means more features, which is a good advantage, but the overlap between the camera views will be so small. On the other hand, using translational oscillation, the camera can observe a limited field of view, but with high overlap between views. The overlap between views is useful for the features convergence and the localization accuracy.

Sensor	Image Size	640 x 480 pixel
	Pixel error std.	1.0
	Field of View	90°
	Frame Rate	10 fps
	Intrinsic Parameters	[u_0 v_o au av]
		[320 240 320 320]
Robot	Motion Increment	0.05 m
	Motion Error	[X Y θ]
		[0.005 0.005 0.05]
Features	Number of Features	162 features
	Features for Update	10 features
	Dynamic Features	6 features
	Features Initialization	[first frame other]
		[10 1]

Table 1: Simulation Parameters

(a) The amplitude effect using frequency = 3 Hz.

(b) The frequency effect using amplitude = 0.07 m.

Fig. 4: Effect of oscillation parameters on the localization errors in case of static robot.

4.2. Moving Robot Testing

In these experiments, we test the translational and rotational oscillation patterns for different robot motions: forward, curved, and lateral. The robot moves with a constant velocity = 0.5 (m/s) and 160 frames were captured in each experiment.

Fig. 5 shows the results of changing the amplitude for the three types of motion. We can observe that, in general by increasing the amplitude, the localization errors decrease. The translational oscillation showed better performance in most of experiments than the rotational one. The rotational oscillation showed better performance only in case of the orientation error of the robot lateral motion. Fig. 6 shows the results of changing the frequency for all types of motion. It is observed that, The translational oscillation showed better performance in most of experiments than the rotational one. The rotational oscillation showed better performance in case of the orientation error of the robot lateral motion.

Through the robot motion, another motion is added to the camera oscillation, and so the need for fast convergence of the features is increased. Using translational oscillation pattern, the camera can observe fewer number of features for more times and then the features can converge quickly. On the other hand, using rotational oscillation, the camera can see larger number of features for less times and then the features need more time to converge. For VLSAM, converged lower number of features is better than non converged larger number of features and this what translational oscillation can exactly offer. This in turn gives translational pattern advantage in the dynamic case.

5. CONCLUSIONS

In this paper, two biologically inspired delibrate camera oscillations patterns, namely: translational and rotational oscillation, were investigated to explore their effect on VSLAM algorithm performance. The two patterns were checked for static and moving robot using different amplitudes and frequencies. In general, the translational oscillation pattern showed better performance than the rotational one. This in the static robot is attributed to the overlap between the camera views. In the moving robot experiments, it can be attributed to the increasing number of observed features for long time. Therefore, we can conclude that the translational oscillation is better than the rotational oscillation for the VSLAM algorithms.
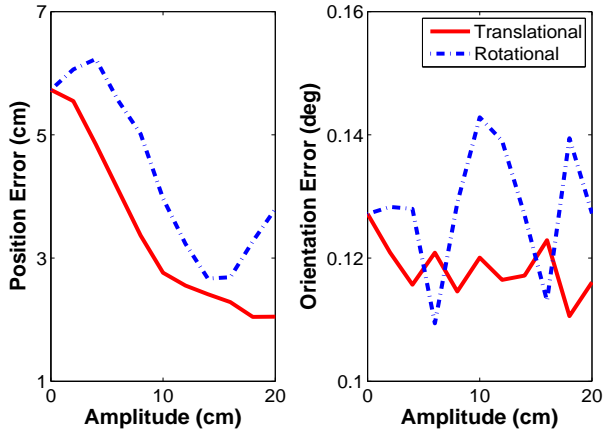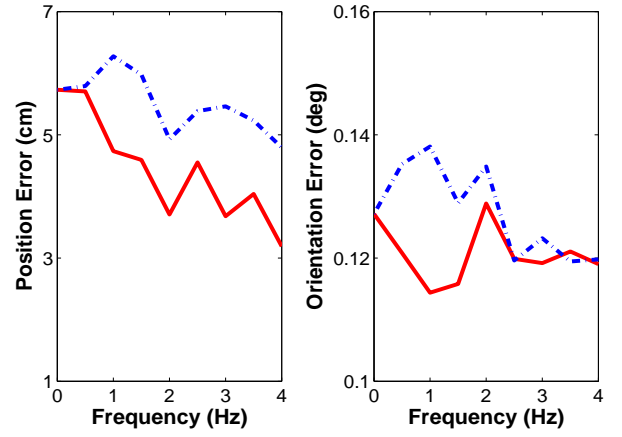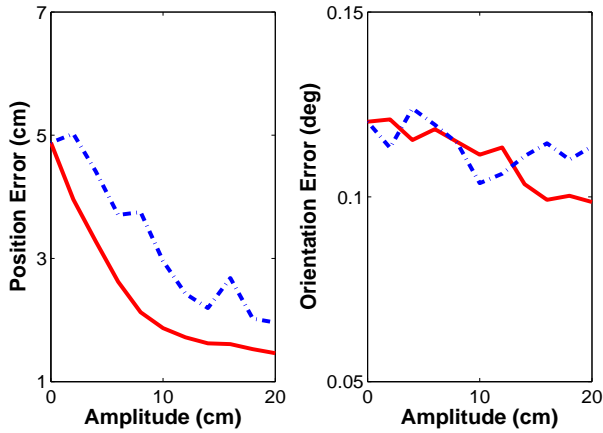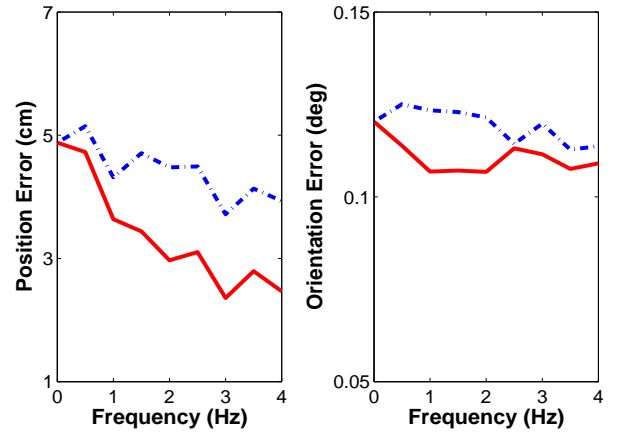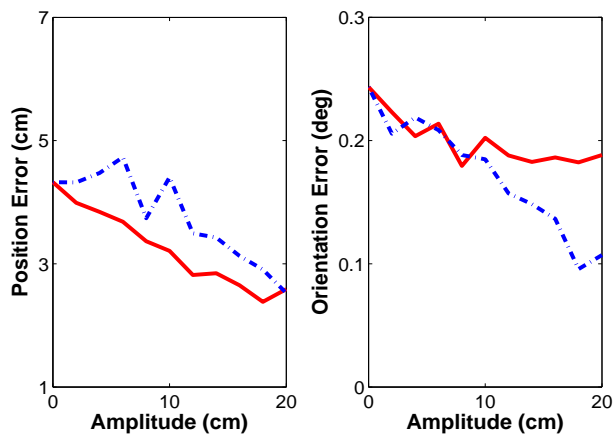
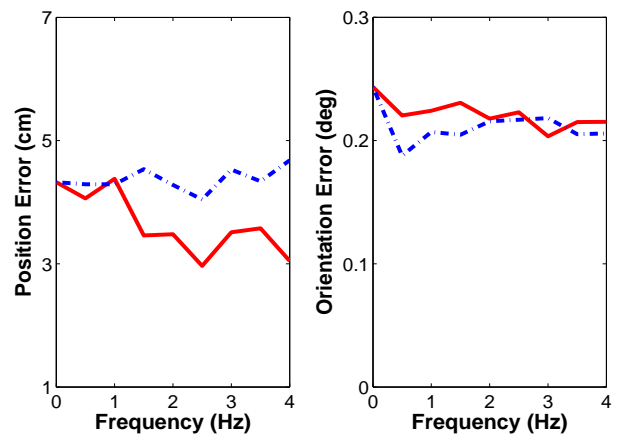Fig. 5: Effect of oscillation amplitude on the localization errors using frequency = 3 Hz for different robot motion.

Fig. 6: Effect of oscillation frequency on the localization errors using amplitude = 0.07 m for different robot motion.

6. REFERENCES

[1] Yosuke Bando. *Single-Shot Image Deblurring with Modified Camera Optics*. PhD thesis, University of Tokyo, 2009.

[2] Alfred Bruckstein, RobertJ. Holt, Igor Katsman, and Ehud Rivlin. Head movements for depth perception: Praying mantis versus pigeon. *Autonomous Robots*, 18(1):21–42, 2005.

[3] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, 2007.

[4] Ethan Eade and Tom Drummond. Scalable monocular SLAM. In *Computer Vision and Pattern Recognition, CVPR*, volume 1, pages 469–476, 2006.

[5] Mohamed Heshmat, Mohamed Abdellatif, and Hossam Abbas. Improving visual SLAM accuracy through deliberate camera oscillations. In *IEEE International Symposium on Robotic and Sensors Environments, ROSE*, pages 154–159, 2013.

[6] M O Hongler, Yuri L de Meneses, Antoine Beyeler, and Jacques Jacot. The resonant retina: exploiting vibration noise to optimally detect edges in an image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:1051–1062, 2003.

[7] Karl Kral. Side-to-side head movements to obtain motion depth cues:: A short review of research on the praying mantis. *Behavioural Processes*, 43(1):71–77, 1998.

[8] Karl Kral. The functional significance of mantis peering behaviour. *European Journal of Entomology*, 109:295–301, 2012.

[9] Christopher Mei, Gabe Sibley, Mark Cummins, Paul Newman, and Ian Reid. RSLAM: A system for large-scale mapping in constant-time using stereo. *International Journal of Computer Vision*, 94(2):198–214, 2011.

[10] M Anthony Lewis Mark E Nelson. Look before you leap: peering behavior for depth perception. In *the Fifth International Conference on Simulation of Adaptive Behavior*, volume 5, page 98, 1998.

[11] Peter N Prokopowicz and Paul R Cooper. The dynamic retina: Contrast and motion detection for active vision. *International journal of computer vision*, 16(3):191–204, 1995.

[12] J. Sola, A. Monin, M. Devy, and T. Vidal-Calleja. Fusing monocular information in multicamera SLAM. *IEEE Transactions on Robotics*, 24(5):958–968, 2008.

[13] Joan Solà, David Marquez, Jean-Marie Codol, and Teresa Vidal-Calleja. EKF-SLAM toolbox for matlab. *http://www.joansola.eu/JoanSola/eng/toolbox.html*, 2009.

[14] Joan Sola, Teresa Vidal-Calleja, Javier Civera, and José María Martínez Montiel. Impact of landmark parametrization on monocular EKF-SLAM with points and lines. *International journal of computer vision*, 97(3):339–368, 2012.

[15] Hauke Strasdat, José MM Montiel, and Andrew J Davison. Visual SLAM: why filter? *Image and Vision Computing*, 30(2):65–77, 2012.

[16] M. Tomono. Robust 3D slam with a stereo camera based on an edge-point ICP algorithm. In *IEEE International Conference on Robotics and Automation*, pages 4306–4311, 2009.

[17] Danping Zou and Ping Tan. CoSLAM: Collaborative visual SLAM in dynamic environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):354–366, 2013.