DYNAMIC FEATURE DETECTION USING VIRTUAL CORRECTION AND CAMERA OSCILLATIONS

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 a static environment. In this paper, we relax the severe assumption of a static scene to allow for the detection and deletion of dynamic points. A new "virtual correction" method is introduced which serves to detect the dynamic points by checking the re-projection error of the points before and after the virtual measurement update. It can also recover the erroneously excluded useful features, particularly the distant points which may be deleted because of the change in its position after new measurement observation. Deliberate camera oscillations are also used to improve the VSLAM accuracy and the camera observability. The simulation results showed the effectiveness of the virtual correction in improving the consistency of the VSLAM and the detection of dynamic points and in particular for difficult scenarios.

Index Terms— Visual SLAM, Dynamic Points Detection, Camera Oscillation, Virtual Correction.

1. INTRODUCTION

Visual SLAM is the Simultaneous Localization And Mapping using visual information obtained from a single [5, 6] or stereo camera [12, 14]. The SLAM algorithms have gained much popularity due to its importance for mobile robot applications into unstructured environment. Visual SLAM is attractive because it uses available on-board cameras to complete the SLAM objective. The SLAM objective is to localize a robot moving into an unknown environment and have the robot incrementally build a map of this environment. In feature-based SLAM approach, the map is described by a set of features (landmarks), where these landmarks are supposed to be static. The assumption of static scene landmarks is not valid for all scene points, since the robot has to work in a dynamic environment and select only static features [2].

In the literature, several approaches were proposed for handling the dynamic feature problem. A method for including information about camera pose from monocular visual SLAM into 3D object tracker, had been used to exclude these features from the SLAM computation in [21]. Stereo based SLAMMOT was proposed to solve the observability problem of monocular camera as well as to increase the accuracy of localization, mapping and tracking [12]. Bi cameras (using two independent monocular cameras) were claimed to detect and track the dynamic points through making observability analysis to check whether the problem is solvable from a mono vision equipped moving platform. It was concluded that the observer (camera) needs to perform more maneuvers than the target [18]. Multiple independently moving cameras were used to detect the dynamic points. The re-projection error was used to differentiate static points from dynamic points. If the point is uncertain, it will be checked using its appearance in another camera [22].

The re-projection error, RE is the distance between the measured point and the projection of the estimation of the same point and it is represented by the Mahalanobis Distance, MD. The RE was used to distinguish the dynamic point, DP or the inconsistent point. If the RE for this feature is less than a defined threshold, the point is classified as inlier and used for the correction. If the feature's RE is more than the threshold, the point is classified as inconsistent and deleted [20]. The RE method faces the problem of the static points, that may change its positions whenever a new observation is available, are misclassified and can not be recovered. The idea of the misclassified features recovery was shown before [16]. A combination of RANSAC plus extended Kalman filter (EKF) that uses the prior probabilistic information from the EKF in the RANSAC model hypothesize stage was used to solve the data association problem. First, a partial state and covariance update is done using only the reliable set of low-innovation inliers of the RANSAC model. Second, the high-innovation inliers are rescued and a second partial update is
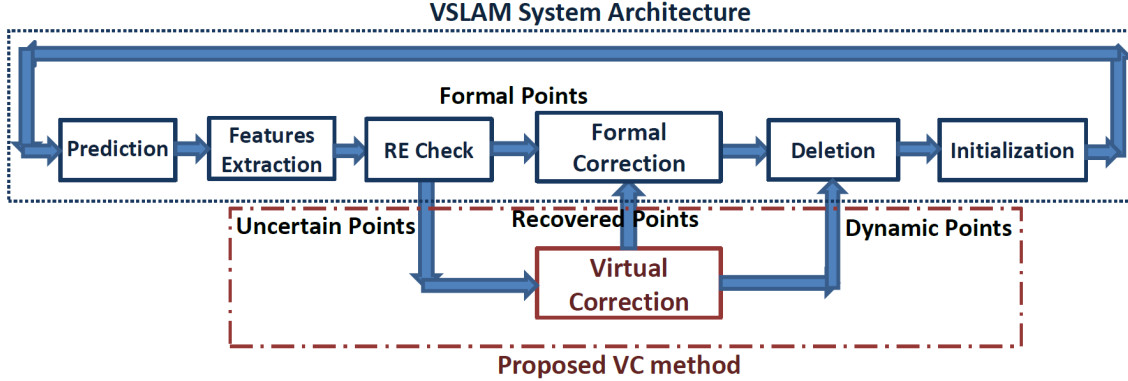
Fig. 1: VSLAM system architecture with the proposed Virtual Correction method.

performed with all the points classified as inliers [16].

To solve camera observability problem, we are using a camera with superimposed oscillations, which was shown to improve depth estimates of the features [9]. It is known that during the human walk the head moves in a horizontal plane from side to side [7, 13]. Head motion is observed not only in humans, but also for animals. Head movements generate retinal displacement which can be used for distance estimation. Praying mantis peer its head left and right to obtain depth perception. Praying 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 [3]. pigeons bob its head forth and back to obtain depth perception, since the pigeon eyes in the head side so the motion is in the lateral direction for pigeons [15]. Camera oscillation had been used for edge detection, contrast detection and motion analysis [10, 17].

In visual SLAM, the retinal image motion is always the result of relative camera-feature motion. This creates an ambiguity as to whether the robot (camera) is moving or the feature is moving or both are moving. This ambiguity is relaxed by the basic static feature assumption in VSLAM algorithms. In the real world, dynamic points violate this assumption.

RE is the main metric to judge if the feature is well classified by the VSLAM algorithm. Our core approach is that, if we can check the points RE, then we can isolate a set of points for which the RE is above a threshold and is believed to include the dynamic points together with the static points having irregular and out of model image motions.

The set of uncertain points are then checked through a technique we call as "virtual correction" for each point separately. This isolation is useful since it can recover misclassified static points and can even make use of dynamic points if they are changed again to static points. Therefore, this problem correlated to dynamic points and static points that were misclassified as dynamic points. Such points are informative for the SLAM system especially for distant points which are

good for robot orientation estimation [4].

The detection of dynamic points is difficult, especially using a monocular camera, because of the limited monocular observability [12, 18]. Camera oscillations are superimposed on the camera main motion (robot motion) to improve features depth estimation [9]. Besides the improvement of the depth estimation of the points, this biologically inspired idea solves the monocular camera observability problem.

This paper is organized as follows: section 2 outlines the basic probabilistic framework for visual SLAM algorithm. Then, in section 3, the virtual correction method is presented. The simulation experiments to verify the VC method effectiveness are described in section 4. The discussion is introduced in section 5, and conclusions are finally given in section 6.

2. VSLAM FRAMEWORK

Extended Kalman Filter based SLAM, EKF-SLAM is used as the core probabilistic framework. VSLAM system architecture with the proposed Virtual Correction, VC method is shown in Fig. 1. The EKF state vector 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 therefore the system can track the camera motion including camera oscillation. Camera oscillation is modeled as a sine wave in the lateral direction as follows:

$$X_s = A.\sin(2\pi f t) \tag{1}$$

where A is the oscillation amplitude, f is the oscillation frequency and t is the time.

The state vector can be described as follows:

$$x = \begin{bmatrix} R & S & M \end{bmatrix}^T = \begin{bmatrix} R & S & L1 & \dots & Ln \end{bmatrix}^T \tag{2}$$

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

$$P = \begin{bmatrix} P_{rr} & P_{rs} & P_{rm} \\ P_{sr} & P_{ss} & P_{sm} \\ P_{mr} & P_{ms} & P_{mm} \end{bmatrix} \quad (3)$$

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 [20].

$$R = \begin{bmatrix} X_r & Y_r & Z_r & q_{r1} & q_{r2} & q_{r3} & q_{r4} \end{bmatrix}^T \quad (4)$$

$$S = \begin{bmatrix} X_s & Y_s & Z_s & q_{s1} & q_{s2} & q_{s3} & q_{s4} \end{bmatrix}^T \quad (5)$$

The EKF prediction step:

$$x \leftarrow f(x, u, n) \quad (6)$$

$$P \leftarrow F_x P F_x^T + F_n N F_n^T \quad (7)$$

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

The EKF correction step:

$$z = y - h(x) \quad (8)$$

$$Z = H_x P H_x^T + W \quad (9)$$

$$K = P H_x^T Z^{-1} \quad (10)$$

$$x \leftarrow x + Kz \quad (11)$$

$$P \leftarrow P - KZK^T \quad (12)$$

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 8 and 9 are the innovation mean and covariance matrix.

3. VIRTUAL CORRECTION

In visual SLAM, based on the basic static feature assumption, the retinal image motion which is a result of relative camera-feature motion is used through the RE method to detect the dynamic points. The RE method has some shortcomings, that RE method can isolate a set of susceptible points. This set
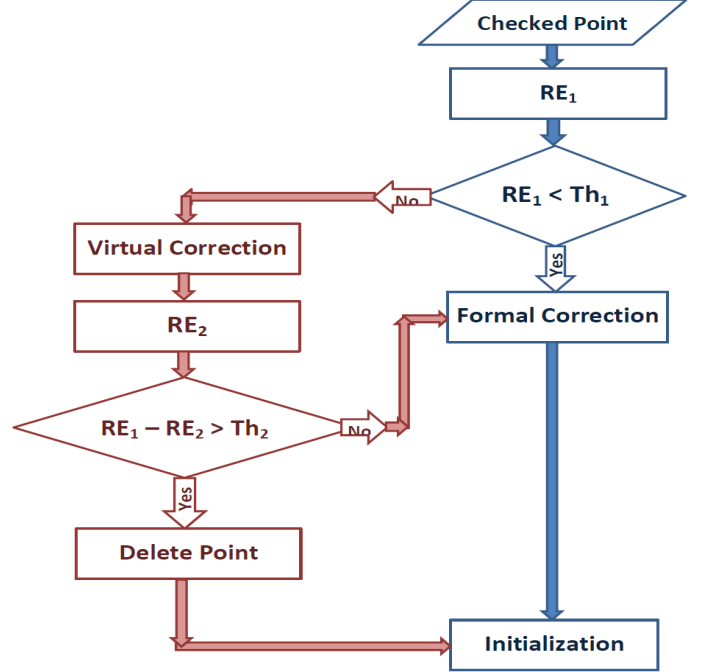

Fig. 2: Flowchart shows the steps of The VC method. The red parts represent the VC added parts over the RE method

includes the dynamic points together with the static points having irregular and out of model image motions. The distant point that may change its position after the measurement update is an example of these irregular points.

The virtual correction method can circumvent the shortcomings of the RE method. First, the RE is checked for each feature. If the RE is less than a defined threshold, the point will be used in the formal correction. If it is more than the threshold, the point will be isolated. After finishing all points inspections, this set of isolated points will be used in the virtual correction. Each point will be used to update the virtual state and covariance. Then the RE will be computed again and the condition will be checked. If the difference between the two REs is more than the defined threshold, the point will be classified as a dynamic point and deleted. If it is less, the point will be recovered and reused in the formal correction. The re-projection error is represented by the squared Mahalanobis distance. Figure 2 shows the flowchart for the VC method.

According to the flowchart, first we compute the RE_1 for the checked point:

$$RE_1 = (y - \hat{e})^T \hat{W}^{-1} (y - \hat{e}) \quad (13)$$

where y is the point measurement, $\hat{e}$ is the projection of point state after the prediction stage and $\hat{W}$ is the sum of the point measurement covariance and the estimation covariance after the prediction stage. Second, we compare RE_1 against de-

fined threshold Th_1:

$$RE_1 = \begin{cases} FormalCorrection & \text{if } RE_1 < Th_1 \\ VirtualCorrection & \text{if } RE_1 > Th_1 \end{cases}$$

The static point which RE_1 is less than the threshold, will be used in the formal correction directly, and the uncertain point which RE_1 is greater than the threshold, will be forwarded to the virtual correction. This uncertain point will be used to update a virtual copy of the state and the covariance. The virtual update equations are the same as the equations from 8 to 12. Third, the RE_2 for the uncertain point will be computed and compared against defined threshold Th_2:

$$RE_2 = (y - e)^T W^{-1} (y - e) \qquad (14)$$

where y is the point measurement, e is the projection of point state after the virtual correction stage and W is the sum of the point measurement covariance and the estimation covariance after the virtual correction stage. Fourth, the difference between the two REs will be checked:

$$(RE_1 - RE_2) = \begin{cases} RecoveredPt. & \text{if } (RE_1 - RE_2) < Th_2 \\ DynamicPt. & \text{if } (RE_1 - RE_2) > Th_2 \end{cases}$$

For points with $(RE_1 - RE_2)$ less than the threshold, it will be used in the formal correction, otherwise deleted. The condition after the virtual correction will pass only the dynamic points. The first RE includes the point motion in addition to the other types of errors, in the opposite the second RE which is computed after the virtual update with the measured point will not include the point motion, therefore the difference between the REs will indicate to the point motion. In this case, the static points that changed its positions due to the new observations will not pass.

4. SIMULATION RESULTS

The simulations are made using the public EKF-SLAM toolbox [19]. In the toolbox, the RE method is used for detecting the dynamic or the inconsistent points. The VC method results will be compared with the toolbox RE method results. In these experiments, we simulate a robot going through an environment of landmarks in an area of 16 m x 16 m. This environment contains 162 landmarks configured into two layers of landmarks. The same features configuration is used throughout all our experiments to avoid the scene features composition effect [1, 8]. 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, the oscillation, and the landmarks.

The 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

Sensor	Image Size	640 x 480 pixel
	Pixel error std.	1.0
	Field of View	90°
	Number of Frames	160 frame
	Frame Rate	10 fps
	Intrinsic Parameters	[u₀ vₒ au av]
		[320 240 320 320]
Robot	Motion Increment	0.05 m
	Motion Error	[X Y θ]
		[0.005 0.005 0.05]
Oscillation	Amplitude	0.05 m
	Frequency	3 Hz
	Sample Time	0.1 sec
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

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 error. After all runs, the mean of all parameters is used for evaluation.

4.1. Static Environment Testing

The objective of these experiments is to verify the ability of the VC method in the recovery of the misclassified features and the effect of the recovered features on the localization errors. Experiments were made for different types of robot motions: forward, curved, and lateral. Each experiment was made once using the RE method and another using the VC method for comparison. Each VC experiment was made once using the steady camera and another using the oscillating camera. The localization errors and the number of recovered points were registered.

Table 2 shows the results of the static environment testing. It can be observed that, the localization errors of all types of motions for the VC method are lower than the RE case. Another observation is the ability of the VC method for recovering the misclassified features in all experiments. The VC method experiments using oscillating camera, show the same number of the recovered features. On the other hand, the experiments show that localization errors are less than steady camera ones for all the motion types. Figure 3 shows the position and the orientation errors for forward robot motion. It is shown that, the VC method experiments using steady camera, show slightly better performance than the RE method. On the other hand the VC method experiments using oscillating camera, show clearly better performance than the RE method.

Robot Motion	Error Types	RE(steady Camera)	Recovered Features	VC(steady Camera)	Recovered Features	VC(Oscillating Camera)	Recovered Features
Forward Motion	Euclidean Error	5.7318 cm	0	5.6966 cm	17	4.2568 cm	16
	Orientation Error	0.1272 deg		0.1208 deg		0.1169 deg	
Curved Motion	Euclidean Error	4.8793 cm	0	4.8409 cm	16	2.7066 cm	17
	Orientation Error	0.1203 deg		0.1159 deg		0.1130 deg	
Lateral Motion	Euclidean Error	4.3247 cm	0	3.9828 cm	16	3.7290 cm	16
	Orientation Error	0.2435 deg		0.2287 deg		0.2144 deg	

Table 2: The Euclidean of position errors, the orientation error and the recovered features number for different motion types in static environment. The RE method results are done using steady camera and the VC method results are done using steady and oscillating cameras.

DP Motion	Error Types	RE(steady Camera)	Dynamic Points	VC(steady Camera)	Dynamic Points	VC(Oscillating Camera)	Dynamic Points
Lateral Motion	Euclidean Error	5.7167 cm	27	5.9133 cm	28	4.5283 cm	29
	Orientation Error	0.1393 deg		0.1429 deg		0.1284 deg	
Inverse Forward	Euclidean Error	5.8394 cm	0	5.7285 cm	0	9.3783 cm	24
	Orientation Error	0.1222 deg		0.1236 deg		0.1353 deg	

Table 3: The Euclidean of position errors, the orientation error and the number of DP. The RE method results are done using steady camera and the VC method results are done using steady and oscillating cameras. The DP velocity = 1 m/s.

DP Motion	Error Types	RE(steady Camera)	Dynamic Points	VC(steady Camera)	Dynamic Points	VC(Oscillating Camera)	Dynamic Points
Lateral Motion	Euclidean Error	6.1239 cm	3	5.7186 cm	2	4.6172 cm	43
	Orientation Error	0.1562 deg		0.1413 deg		0.3739 deg	
Inverse Forward	Euclidean Error	6.1498 cm	0	5.4650 cm	0	12.5170 cm	18
	Orientation Error	0.1170 deg		0.1120 deg		0.1251 deg	

Table 4: The Euclidean of position errors, the orientation error and the number of DP. The RE method results are done using steady camera and the VC method results are done using steady and oscillating cameras. The DP velocity = 0.5 m/s.

DP Motion	Error Types	RE(steady Camera)	Dynamic Points	VC(steady Camera)	Dynamic Points	VC(Oscillating Camera)	Dynamic Points
Lateral Motion	Euclidean Error	5.9099 cm	6	5.8257 cm	11	5.0702 cm	48
	Orientation Error	0.1729 deg		0.1582 deg		0.5228 deg	
Inverse Forward	Euclidean Error	5.6649 cm	0	5.6610 cm	0	15.1492 cm	9
	Orientation Error	0.1308 deg		0.1190 deg		0.1195 deg	

Table 5: The Euclidean of position errors, the orientation error and the number of DP. The RE method results are done using steady camera and the VC method results are done using steady and oscillating cameras. The DP velocity = 0.25 m/s.
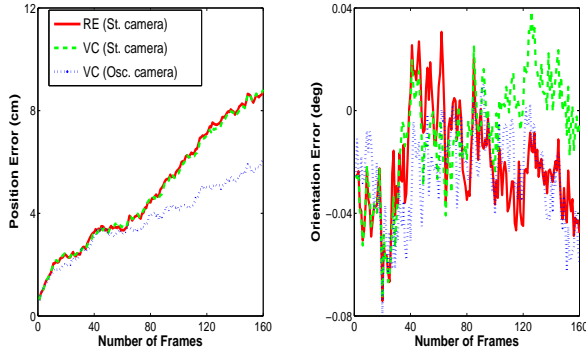
Fig. 3: The position and the orientation error against the number of frames for forward robot motion in static environment using RE method using steady camera and VC method using steady and oscillating camera.
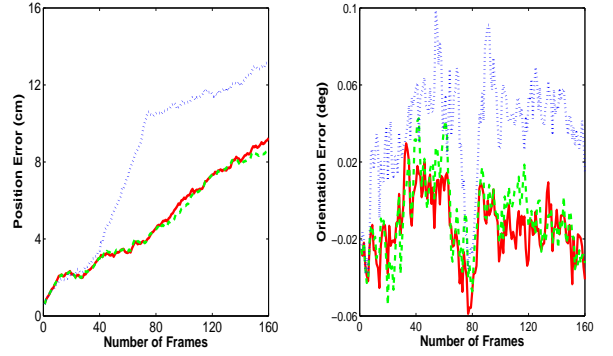

Fig. 4: The position and the orientation error for forward robot motion and lateral DP motion using RE method using steady camera and VC method using steady and oscillating camera. The DP velocity = 1 m/s

4.2. Dynamic Environment Testing

We hope to verify the capability of the VC method in the detection of the dynamic points and if the oscillating camera can extend the VC method dynamic point detection capability. Experiments were made for the basic robot motion type, the forward motion and with two cases of the dynamic points linear motion, the lateral motion and the most difficult motion, the inverse forward motion. Experiments were made using three velocities for the dynamic points: 1 m/s, 0.5 m/s, and 0.25 m/s. Figure 6 shows the motion of the robot and the dynamic points in the two cases of dynamic points motion. The experiments were made using both the RE and the VC methods. Each VC experiment was made one time using the steady camera and another using the oscillating camera. The localization errors and the number of detecting dynamic points were registered.

Fig. 5: The position and the orientation error for forward robot motion and inverse froward DP motion using RE method using steady camera and VC method using steady and oscillating camera. The DP velocity = 1 m/s
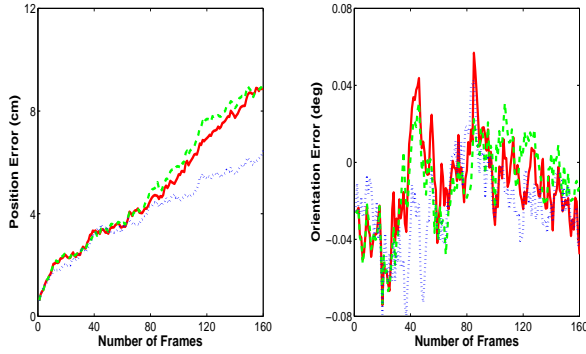
Tables 3, 4, and 5 show the results using the RE method and the VC method for different dynamic point velocities. In case of lateral motion of the dynamic points, the steady camera and the oscillating camera detected the dynamic points successfully. Figure 4 shows the position and the orientation errors for forward robot motion and lateral DP motion and the DP velocity = 1 m/s. It is shown that, the RE and the VC methods experiments using steady camera, show similar performance. On the other hand the VC method experiments using oscillating camera, show better performance than the RE method. In case of the low velocities lateral dynamic points motion, the steady camera ability of detecting the dynamic points is decreased for the two methods, especially, for the velocity which equals the robot velocity. On the opposite, the ability of detecting the dynamic points using oscillating camera is increased significantly. In the difficult inverse forward motion of the dynamic points, the steady camera could not detect any dynamic point, but the oscillating camera still can detect the dynamic points. For the low velocities inverse forward DP motion, the oscillating camera still can detect the dynamic point. These results show that the camera oscillation extends the VC method dynamic point detection capability even in the difficult situations. On the other hand the localization errors increase. Figure 5 shows the position and the orientation errors for forward robot motion and Inverse forward DP motion and the DP velocity = 1 m/s. It is shown that, the localization errors increased in the oscillating camera case.

5. DISCUSSION

In this paper, the VC method is investigated in static and dynamic environment and using fixed and oscillating camera. VC fixes the shortcomings of the RE method in detection the dynamic points and also allows recovering misclassified points.
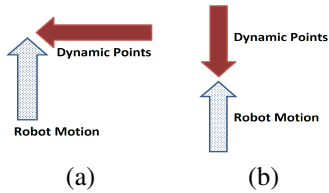
Fig. 6: The dotted blue arrow represents the robot motion and the solid red one represents the DP motion in case of (a) DP lateral motion, (b) DP inverse forward motion

In static environment experiments, the VC showed less localization errors than the RE. These errors decrease, especially the orientation, proves the more consistent VSLAM system using VC method. The decrease of the orientation error increases the consistency of the system significantly [11]. The results also showed that the VC method has added, about 16 points recovered in each experiment. The oscillating camera showed better accuracy than the fixed camera using the VC method.

In dynamic environment experiments, the VC method succeeded to detect dynamic points. By solving the problem of the observeability of the monocular camera, camera oscillations increase the VC method dynamic point detection ability significantly even in the difficult situation like the inverse forward motion. The lateral camera oscillations added a new physical clues for the camera to detect the dynamic points motion. The reason for the localization errors increase in case of inverse forward motion is the misclasified recovered points, especially, the first appearance of the dynamic points and the distant dynamic points. This issue will be studied in detail in our future research agenda.

Our results showed that the VC method can differentiate between static and dynamic points from knowledge of the relative motion between the camera and the points. However, if there is no relative motion, it will be difficult to identify dynamic features. It is very less probable that the feature will oscillate similar to the camera, and this is an advantage of camera oscillation. In the future, we will verify the VC method using real data sets and practical experiments.

6. CONCLUSIONS

In this paper, a new technique is introduced to detect dynamic features and to recover the static misclassified features. Camera imposed oscillations improve the observability and improve depth estimates. The VC method is investigated using fixed and oscillating camera. Following can be concluded for VC:

- detect dynamic points efficiently when combined with camera oscillation.
- recover misclassified features.

7. REFERENCES

[1] Mohamed Abdellatif. Monocular visual SLAM: Improvements from the object side. In *The 13th Mechatronics Forum International Conference*, pages 393–398, 2012.

[2] Tim Bailey and Hugh Durrant-Whyte. Simultaneous localization and mapping (SLAM): Part II. *IEEE Robotics & Automation Magazine*, 13(3):108–117, 2006.

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

[4] Javier Civera, Oscar G Grasa, Andrew J Davison, and JMM Montiel. 1-point ransac for extended kalman filtering: Application to real-time structure from motion and visual odometry. *Journal of Field Robotics*, 27(5):609–631, 2010.

[5] 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.

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

[7] Wolfgang Einhäuser, Frank Schumann, Stanislavs Bardins, Klaus Bartl, Guido Böning, Erich Schneider, and Peter König. Human eye-head co-ordination in natural exploration. *Network: Computation in Neural Systems*, 18(3):267–297, 2007.

[8] Mohamed Heshmat and Mohamed Abdellatif. The effect of feature composition on the localization accuracy of visual SLAM systems. In *the International Conference on Computer Vision Theory and Applications, VISAPP*, pages 419–424, 2012.

[9] 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.

[10] 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.

[11] Shoudong Huang and Gamini Dissanayake. Convergence and consistency analysis for extended Kalman filter based SLAM. *IEEE Transactions on Robotics*, 23(5):1036–1049, 2007.

[12] Kuen-Han Lin and Chieh-Chih Wang. Stereo-based simultaneous localization, mapping and moving object tracking. In *IEEE/RSJ International Conference on Intelligent Robots and Systems,IROS*, pages 3975–3980, 2010.

[13] Jack D Mcdonald, A Terry Bahill, and Mark B Friedman. An adaptive control model for human head and eye movements while walking. *IEEE Transactions on Systems, Man and Cybernetics*, (2):167–174, 1983.

[14] 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.

[15] Reinhold Necker. Head-bobbing of walking birds. *Journal of Comparative Physiology A*, 193(12):1177–1183, 2007.

[16] Lina M Paz, Pedro Piniés, Juan D Tardós, and José Neira. Large-scale 6-dof SLAM with stereo-in-hand. *IEEE Transactions on Robotics*, 24(5):946–957, 2008.

[17] 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.

[18] Joan Solà. *Towards Visual Localization, Mapping and Moving Objects Tracking by a Mobile Robot: a Geometric and Probabilistic Approach.* PhD thesis, Institut National Polytechnique de Toulouse, 2007.

[19] 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.

[20] 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.

[21] Somkiat Wangsiripitak and David W Murray. Avoiding moving outliers in visual SLAM by tracking moving objects. In *IEEE International Conference on Robotics and Automation, ICRA*, pages 375–380, 2009.

[22] 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.