Outdoor Robot Localization using Scale Ratio Adjustment of Triplet Frames

My-Ha Le
Faculty of Electrical and Electronic Engineering
Ho Chi Minh City University of Technology and Education
No.1 Vo Van Ngan Str., Thu Duc Dist., Ho Chi Minh City, Viet Nam

Abstract—**This paper proposes a method for camera motion estimation and outdoor scene reconstruction from multiple views of monocular system. Firstly, invariant feature of each triplet of frames are detected and matched in consecutive-overlap pair. Wrong correspondence points matching are rejected by RANSAC algorithm to find fundamental matrix. Second, the rotation and translation constrain are derived from essential matrix which is computed based on fundamental matrix and intrinsic camera parameters. The scale adjustment is use to estimate the ratio of translation and generate the motion trajectory. Thirdly, 3D points of scene are triangulated and refined because of non-coincident clouds generated from various triplets of frames. The simulation results will demonstrate the effectiveness of this method.**

Keywords- SIFT, RANSAC, multiple views geometry, motion estimation, 3D reconstruction

I. INTRODUCTION

Motion estimation and 3D modeling of urban scene is one of important process in various applications of visual SLAM, visual odometry for autonomous mobile robot navigation and advanced driver assistance systems. Other application also can be considered are virtual environment and scene planning. Some progress has been made in the trajectory estimation and 3D modeling obtained during the last few years but they needed a large amount of work done by hand or apparatus, such as laser radar, and airborne light detection and ranging. They are usually expensive and require much more time for data acquisition.

In recent years many algorithms have been developed for motion estimation, which can roughly be devised into several categories, namely methods using monoscopic [1], methods using stereoscopic [2] and camera-electromagnetic device combination [3-6]. In the first group, monoscopic usually require robust feature detection and tracking through a certain number of images. Using these tracked features, the motion trajectory could be estimated as well as scene structure by using well known structure from motion algorithm [7]. In the second group, the 3D structure of scene could be reconstructed as camera calibration is known by triangulation. Base on the point clouds of consecutive frame the motion of camera will be estimated. In this case, the scale ambiguity exist in the monoscopic case is eliminated. Most of experiment presents that stereoscopic yield a better performance [8]. In the third group, they combined visual sensor and other sensors to increase the accuracy rate and reduce drift problem. Some of them make use of internal measurement units (IMUs) while the others make use of GPS and wheel encoder. The further separation can be done base on the used method, for instance, feature matching between consecutive images [9-11] or feature tracking cover a sequence of frame [3], [6], [12].

Without using any addition device, e.g. laser sensor out of single camera, our proposed method overcomes some disadvantages mentioned above. It is much cheaper and compact. The flow chart of proposed method can be seen in figure 1. From monocular system, sequence image are acquired along the scene. The triplet of frames is extracted from sequence input frames. SIFT algorithm [13], [14] is applied to find invariant feature and matching of each consecutive-overlap pair of views in triplet. The estimation of fundamental matrix and intrinsic parameters of camera is computed base on 8-points algorithm [15] and Jean-Yves Bouguet [16] method respectively. Essential matrix is derived from computed fundamental matrix and above calibration information. The rotation and translation constrain will be obtained using the method from Horn [17]. Because the motion is only estimates up to scale so we need to estimate the scale ration of the third frame [18]. Linear triangulation is the next step to generate 3D point cloud of scene. However the 3D point clouds generated from different image pair of triplet will yield a non-coincident structure so that the refinement step is needed to optimize the camera motion. Finally, the texture mapping will be perform to map the true value R,G,B color from 2D image pixels to 3D point clouds.

This paper is organized into 5 sections. The next section describes motion estimation from correspondence points and scale adjustment method. Section III presents point clouds generation. We also explain refinement of non-coincident point clouds in this section. Experiments are showed in section IV. Finally, paper is finished with conclusions and point out future works discussed in section V.

II. MOTION ESTIMATION

In order to compute the frame-to-frame motion we first find the essential matrix which depends on the relative position and orientation of a pair of views, and can be estimated using point correspondences and intrinsic parameters of camera. In this

This project was supported by Ho Chi Minh City University of Technology and Education, 2014 (T2014-07TĐ)

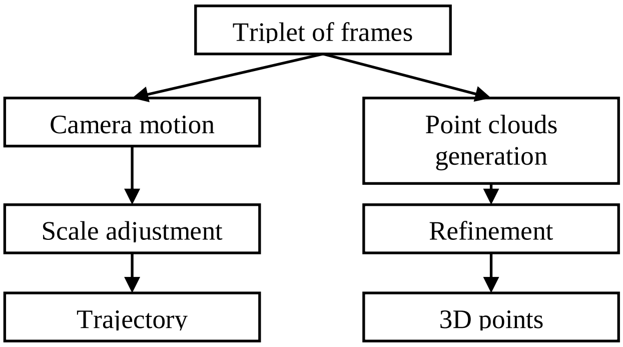


Figure 1. General proposed scheme

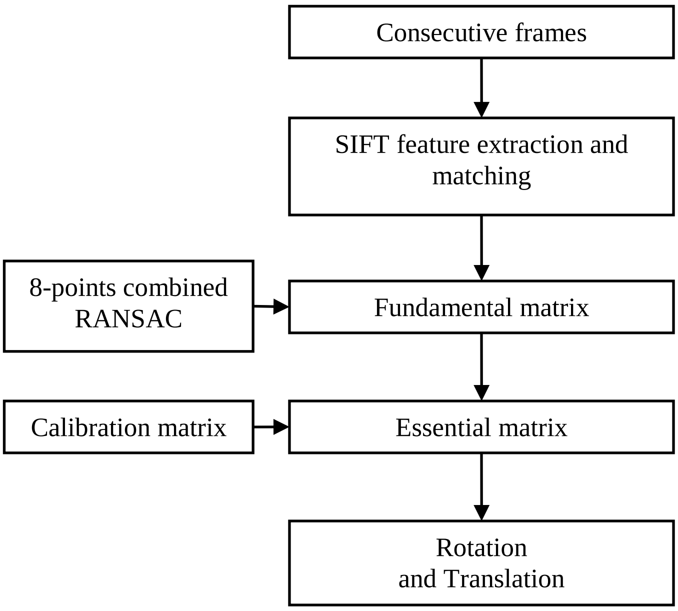


Figure 2. Motion estimation scheme

section, we explain what is camera model, how to extract and match salient features as well as how a essential matrix can be decomposed to recover the camera motion, and, thereby, camera projection matrices. The brief description of this step is showed in the over view of figure 2.

A. Camera model

We use the projective geometry throughout this paper to describe the perspective projection of the 3D scene onto 2D images [15]. This projection is described as follows:

$$x = PX \qquad (1)$$

where P is a 3×4 projection matrix that describes the perspective projection process, $X = [X, Y, Z, 1]^T$ and $x = [x, y, 1]^T$ are vectors containing the homogeneous coordinates of the 3D world coordinate, respectively, 2D image coordinate.

When the ambiguity on the geometry is metric, (i.e., Euclidean up to an unknown scale factor), the camera projection matrices can be put in the following form:

$$P = K[R|-RT] \qquad (2)$$

with T and R indicating the translation and rotation of the camera and K , an upper diagonal 3×3 matrix containing the intrinsic camera parameters.

$$K = \begin{bmatrix} f_x & s & u_x \\ & f_y & u_y \\ & & 1 \end{bmatrix} \qquad (3)$$

where f_x and f_y represent the focal length divided by the horizontal and vertical pixel dimensions, s is a measure of the skew, and (u_x , u_y) is the principal point. The check board used for calibration is present in figure 4.

B. Feature extraction and matching

There are many kind of features are considered in recent research in feature extraction and matching problem including Harris [19], SIFT, PCA-SIFT, SURF [20], [21], etc. SIFT is first presented by David G Lowe in 1999 and it is completely presented in 2004. As we know on experiments of his proposed algorithm is very invariant and robust for feature matching with scaling, rotation, or affine transformation. According to those conclusions, we utilize SIFT feature points to find correspondent points of image pairs. The SIFT algorithm are described through these main steps: scale-space extrema detection, accurate keypoint localization, orientation assignment and keypoint descriptor. SIFT features and matching is applied for one image pair as showed in Fig. 3. The result of correspondence point will be used to compute fundamental matrix described in the next step.

C. Camera motion

The result of correspondence point in previous step will be used to compute fundamental matrix. The epipolar constraint represented by a 3x3 matrix is called the fundamental matrix, F. This method based on two-view geometry theory which was studied completely [15].

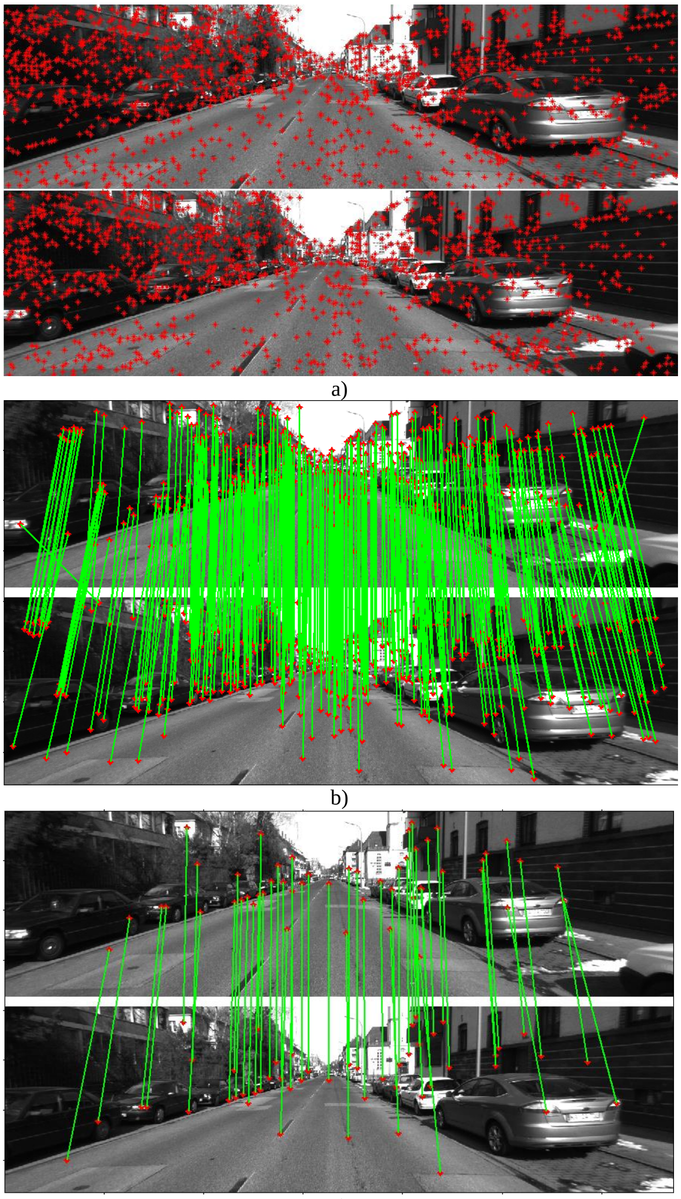
If the intrinsic parameters of the cameras are known, the fundamental epipolar constraint above can be represented algebraically by a 3x3 matrix, called the essential matrix. We have to do camera calibration to find these parameters. The good Matlab toolbox for doing camera calibration was provided by Jean-Yves Bouguet [16]. When we know camera intrinsic parameter, we can form the matrix K.

$$E = K'^T F K \qquad (4)$$

where E is essential matrix, K' and K are intrinsic parameters of frame 1 and 2. In the case of using the monocular camera, we have K' = K. The projection matrix of the first frame P is set follow this equation:

$$P = K[I|0] \qquad (5)$$

The second projection matrix is found from four possible choices: $P' = (UWV^T|+u_3)$ or $P' = (UWV^T|-u_3)$ or $P' = (UW^TV^T|+u_3)$ or $P' = (UW^TV^T|-u_3)$, where U and V are found from SVD decomposition of E, u_3 is the last column of U and W.



a)

b)

c)

Figure 3. SIFT feature extraction and matching. a) SIFT features, b) features matching before RANSAC, c) features matching after RANSAC

$$W = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (6)$$

Only one of these four choices is possible for the second camera. We can find it by testing whether a reconstructed point lies in front of both cameras.

D. Scale Adjustment

Until this step, only one global scale parameter remains unknown, it is scale ratio of translation between each pair of views. In order to obtain this ratio, there are a number of method were proposed. Bundle adjustment [22] is one of typical one. The expensive computation is a disadvantage of this method when the initial estimation far from the true value.

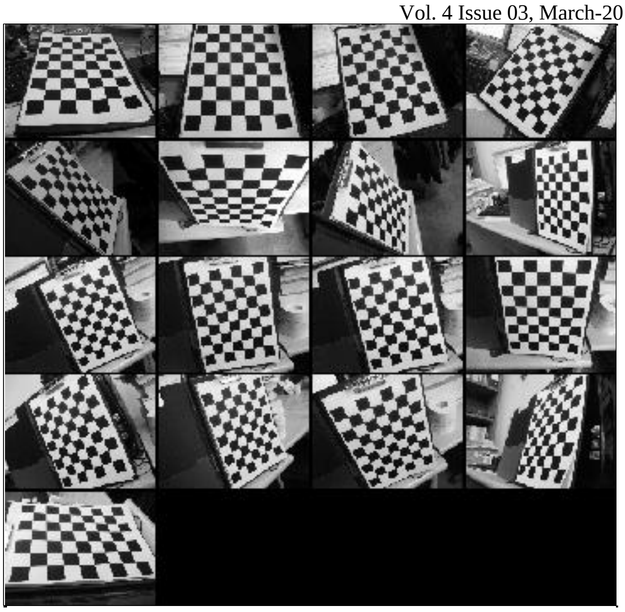


Figure 4. Camera calibration templates

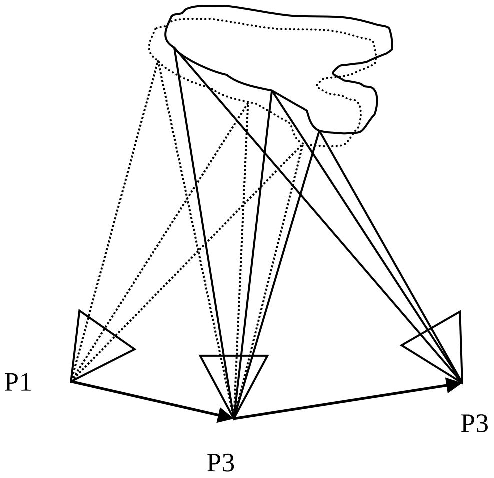


Figure 5. Camera motion optimization

Another common method use 3D-2D correspondence to estimate this parameter but most of solution generate the non-linear and interactive problem so it is also expensive. A linear solution is DLT (Direct Linear Transform) where a set of linear equation are solved by SVD (Singular Value Decomposition) method [23]. In this paper we apply a modification of DLT solution was presented in [18].

III. POINT CLOUD GENERATION

Having obtained projection matrices, 3D points can be computed from their measured image positions in two or more views. This step is called triangulation [24] in 3D space. Ideally, 3D points should lie at the point of intersection of the back-projected rays in all of consecutive pair views. However, because of measurement noise as well as inaccuracy of motion estimation, the reconstructed structures were non-coincident. Thus 3D points must be refined after generation. See figure 5 for illustration.

A. Linear Triangulation

Triangulation is the simplest but effective method to compute the 3D point X from the matching images points x

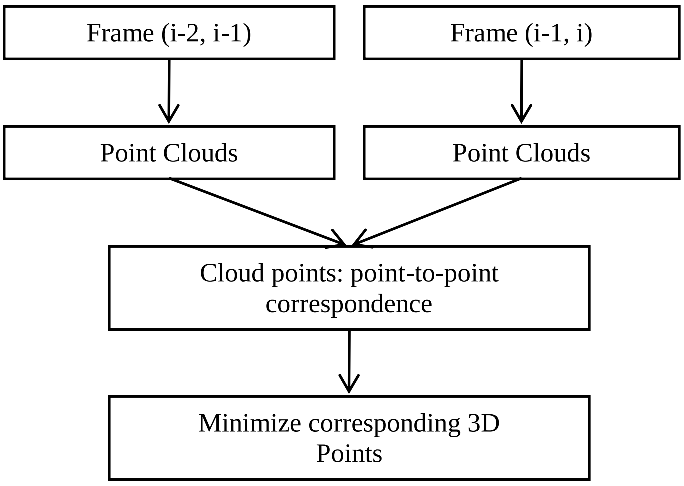

Frame (i-2, i-1)
Frame (i-1, i)
Point Clouds
Point Clouds
Cloud points: point-to-point correspondence
Minimize corresponding 3D Points

Figure 6. Point clouds refinement scheme

and x’ given two camera matrices. Difference with dense depth estimation or disparity map for image region, the linear triangulation is suitable for sparse point depth measurement. First, we have x = PX, but x is determined only up to scale in homogeneous coordinates. So we require that the vector x is collinear with vector PX by setting x(PX) = 0 which gives us two independent equations:

$$(P^{3T}X) - P^{1T}X = 0 \quad (7)$$

$$y(P^{3T}X) - P^{2T}X = 0 \quad (8)$$

where P^{iT} is the ith row of matrix P.

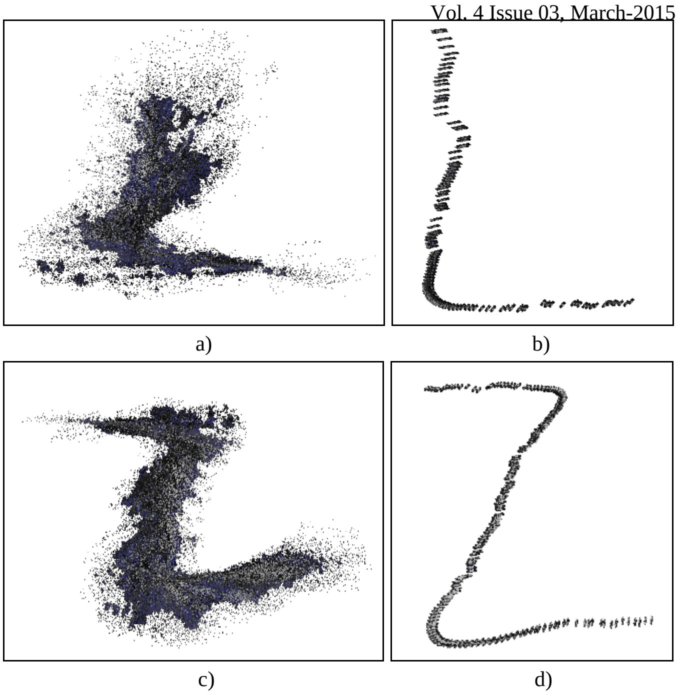
Similarly, we get another 2 equations from x’ and P’ and we establish an equation AX = 0. This equation is solved by SVD method to get X.

B. Point cloud refinement

The point clouds generated by the triplet frame of two overlap-consecutive pair of view are distinct. The 3D correspondence points of each triple of frames are known exactly according to correspondence points in 2D images. The refinement performs the interactive to minimize the distance of two point clouds. This performance will be extended for all point clouds which are reconstructed from each consecutive triplet of frame of sequence data images. Proposed scheme was presented in the figure 6.

IV. EXPERIMENTS

We experimented on outdoor images which are acquired from Karlsruhe dataset (www.cvlibs.net). All result were simulated on Intel(R) Core(TM) i5 CPU 750@2.67 GHz with 3GB RAM under Matlab environment. The point clouds and motion were visualized by Open Scene Graph tool. In the first and the second experiment, we run 120 and 193 images with 1344x372 sizes from 2010_03_09_drive_0019.zip dataset. Figure 7(a), 7(c) is the point clouds of scene and figure 7(b), 7(d) are the motion of camera mounted on vehicle. In the motion trajectory we keep both translation and rotation of camera poses. In the third and fourth experiment, see figure 8a and figure 8b, we combined both point clouds and motion trajectory in one graph. They were implemented on the 48 and 120 images with 1344x372 sizes from 2010_03_09_drive_0082.zip dataset.



a) b)

c) d)

Figure 7. Point clouds and motion trajectory. 7a, 7c are point clouds of scene. 7b, 7d are motion trajectories.

V. CONCLUSION

Motion estimation and scene reconstruction from multiple views are presented on this paper. Some advantage points can be realized through our explanation. First, we avoid using bundle adjustment which will expensive for computational time while the initial estimated far from the true value. We utilize minimization of point clouds distance instead. Second, in the scale estimation, we trend to use linear method to estimate this parameter. It needs less 2D-3D correspondence points. Our future woks focus on comparison of this method with stereoscopic based method. Also, we will improve and develop this method for Omni-directional camera by using its video data in outdoor scene. The last ambition is application of this method to real time systems.

ACKNOWLEDGMENT

This project was supported by Ho Chi Minh City University of Technology and Education, 2014 (T2014-07TĐ)

REFERENCES

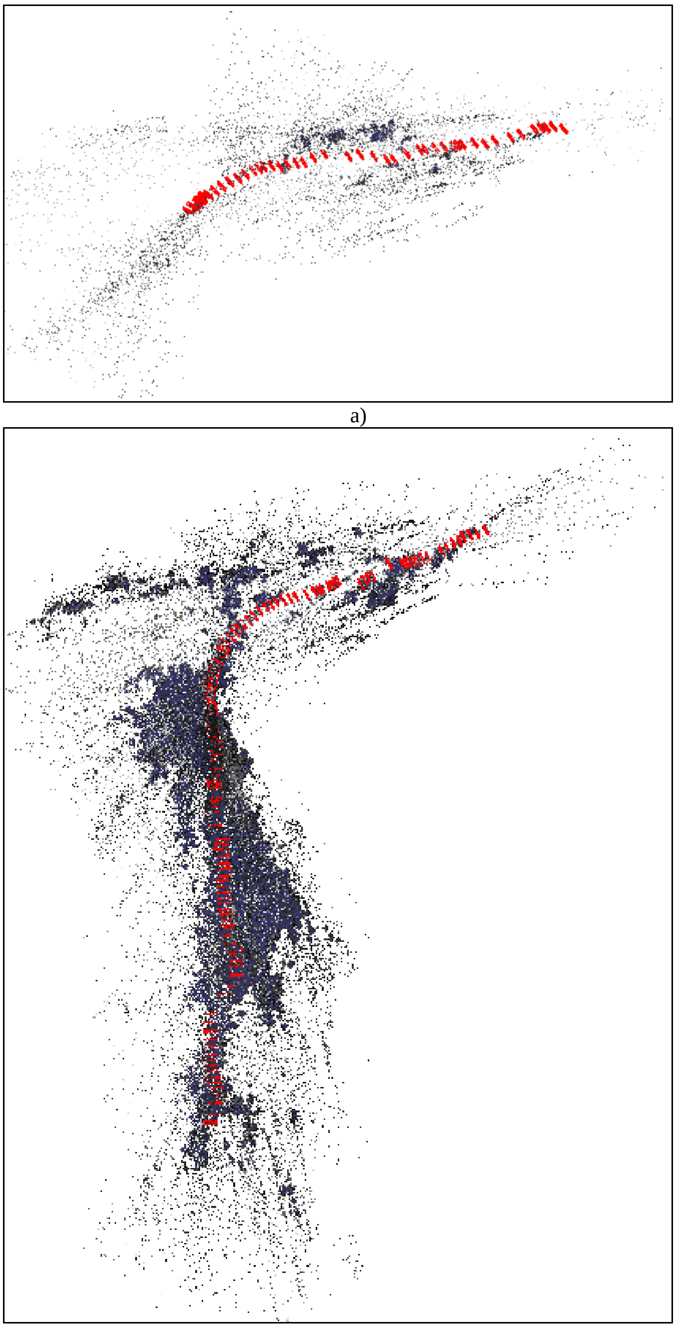
[1] K. Yamaguchi, T. Kato, and Y. Ninomiya, “Vehicle ego-motion estimation and moving object detection using a monocular camera,” in Proceedings of the 18th International Conference on Pattern Recognition, 2006, pp. 610 – 613.

[2] A. Milella and R. Siegwart, “Stereo-based ego-motion estimation using pixel-tracking and iterative closest point,” in Proceedings of the Fourth IEEE International Conference on Computer Vision Systems, 2006.

[3] C. Dornhege and A. Kleiner, “Visual odometry for tracked vehicles,” in Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics, 2006.

[4] M. Agrawal and K. Konolige, “Real-time localization in outdoor environments using stereo vision and inexpensive gps,” in Proceedings of the 18th International Conference on Pattern Recognition, 2006, pp. 1063 – 1068.

[5] M. Agrawal, K. Konolige, and R. C. Bolles, “Localization and mapping for autonomous navigation in outdoor terrains: A stereo vision approach,” in Proceedings of the Eighth IEEE Workshop on Applications of Computer Vision, 2007.



a)

b)

Figure 8. Combined point clouds and motion trajectory

[6] M. Agrawal and K. Konolige, “Rough terrain visual odometry,” in Proceedings of the International Conference on Advanced Robotics, August 2007.

[7] D. Nister, O. Naroditsky, and J. Bergen, “Visual odometry,” in IEEE Computer Society Conference Computer Vision and Pattern Recognition, vol. 1, 2004, pp. 652 – 659.

[8] H. Badino, “A robust approach for ego-motion estimation using a mobile stereo platform,” in First International Workshop on Complex Motion, 2004, pp. 198 – 208.

[9] A. Howard, “Real-time stereo visual odometry for autonomous ground vehicles,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, September 2008, pp. 3946 – 3952.

[10] A. Talukder, S. Goldberg, L. Matthies, and A. Ansar, “Real-time detection of moving objects in a dynamic scene from moving robotic vehicles,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 2, October 2003, pp. 1308 – 1313.

[11] A. Talukder and L. Matthies, “Real-time detection of moving objects from moving vehicles using dense stereo and optical flow,” in IEEE International Conference on Intelligent Robots and Systems, vol. 4, September 2004, pp. 3718 – 3725.

[12] A. E. Johnson, S. B. Goldberg, Y. Cheng, and L. H. Matthies, “Robust and efficient stereo feature tracking for visual odometry,” in IEEE International Conference on Robotics and Automation, May 2008, pp. 39 – 46.

[13] D. Lowe: Object recognition from local scale-invariant features. In Proc. of the International Conference on Computer Vision, pp. 1150-1157 1999

[14] D. Lowe, “Distinctive Image Features from Scale-Invariant Interest Points”, International Journal of Computer Vision, Vol. 60, pp. 91-110, 2004.

[15] R. I. Hartley and A. Zisserman: Multiple View Geometry in Computer Vision. Cambridge University Press, ISBN: 0521540518, 2nd edition, 2004.

[16] Jean-Yves and Bouguet: Camera Calibration Toolbox for Matlab.

[17] B. Horn, “Recovering baseline and orientation from essential matrix,” 1990. [Online]. Available: citeseer.ist.psu.edu/horn90recovering.html

[18] I. Esteban,J. Dijk, F. Groen, “Automatic 3D reconstruction of the urban landscape”**.** International Congress on Ultra Modern Telecomunications and Control Systems (ICUMT), 2010.pp. 421-428

[19] C. Harris, M. Stephens: A combined corner and edge detector, in Proceedings of the 4th Alvey Vision Conference, pp. 147-151, Manchester, UK 1998.

[20] Herbert Bay, Tinne Tuytelaars, Luc Van Gool: SURF: speeched up robust features, in proceeding of ECCV, Vol. 3951, pp. 404-417, 2006

[21] Luo Juan and Oubong Gwun: A Comparison of SIFT, PCA-SIFT and SURF, in International Journal of Image Processing, Volume 3, Issue 5, 2010

[22] B. Triggs, P. Mclauchlan, R. Hartley, and A. Fitzgibbon, “Bundle adjustment – a modern synthesis,” 2000.

[23] Y. Abdel-Aziz and H. Karara, “Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry,” Proceedings of the Symposium on Close-Range Photogrammetry (pp. 1-18), 1971.

[24] R. I. Hartley and P. Sturm, “Triangulation”, American Image Understanding Workshop, pages 957–966, 1994.