Real-Time Global Localization with A Pre-Built Visual Landmark Database

Zhiwei Zhu, Taragay Oskiper, Supun Samarasekera, Rakesh Kumar and Harpreet S. Sawhney
Sarnoff Corporation, 201 Washington Road, Princeton, NJ 08540, USA
{zzhu, toskiper, ssamarasekera, rkumar, hsawhney}@sarnoff.com

Abstract

In this paper, we study how to build a vision-based system for global localization with accuracies within 10cm. for robots and humans operating both indoors and outdoors over wide areas covering many square kilometers. In particular, we study the parameters of building a landmark database rapidly and utilizing that database online for real-time accurate global localization. Although the accuracy of traditional short-term motion based visual odometry systems has improved significantly in recent years, these systems alone cannot solve the drift problem over large areas. Landmark based localization combined with visual odometry is a viable solution to the large scale localization problem. However, a systematic study of the specification and use of such a landmark database has not been undertaken.

We propose techniques to build and optimize a landmark database systematically and efficiently using visual odometry. First, topology inference is utilized to find overlapping images in the database. Second, bundle adjustment is used to refine the accuracy of each 3D landmark. Finally, the database is optimized to balance the size of the database with achievable accuracy. Once the landmark database is obtained, a new real-time global localization methodology that works both indoors and outdoors is proposed. We present results of our study on both synthetic and real datasets that help us determine critical design parameters for the landmark database and the achievable accuracies of our proposed system.

1. Introduction

Real-time 6 degree-of-freedom (DOF) pose estimation of moving cameras has been studied for several decades with numerous applications in robotics, vehicle navigation and augmented reality. Most of the systems [7, 1, 9, 5, 12, 4, 10, 14] for camera pose estimation are based on detection and tracking a set of natural feature points in the scene. Assuming the scene feature points are stationary, methods based on this idea use them as reference points, thus the relative camera motion between two consecutive frames can be estimated. While contact-free and non-intrusive, these incremental-motion based methods work well only for a short period of time and the systems drift eventually as the error accumulates. This is a rather significant weakness of the existing visual odometry systems. Furthermore, uninterrupted operation over long periods of time may not be possible as even single frame drops or errors can be catastrophic. This is referred to as "robustness" issue in [13]. For navigation in large-scale areas continuously, such incremental-motion based visual odometry is not practically viable.

The latest research efforts [6, 5, 11, 9] are aimed at overcoming these limitations. A simple but effective method is to incorporate more measurements from either same or different sensors. For example, a rough map (low accuracy) of where the observer has been can be utilized to improve visual odometry [6]. In addition, different types of non-vision sensors including GPS [5], IMUs [11] and absolute orientation sensors [9] are incorporated into visual odometry systems. More recently, an extra pair of stereo-cameras facing backwards has been added to form a multiple-stereo-pairs rig [10], which demonstrates quite significant improvements both in accuracy as well as robustness. However, a rough map can not always be available; and the GPS measurements can not always be accurate since the satellite signals can dropout easily in urban canyons and indoors. Even with the use of multiple cameras and IMUs, errors still accumulate to grow the drift, although at a slower speed.

Due to recent advances in the image searching techniques, real-time landmark matching with a large landmark database has become possible. For example, in [13], a rapid recognition technique with randomized lists is utilized to perform landmark matching in real-time. The performance of this system is shown over a small landmark database (several hundred landmarks). In [14], an efficient indexing technique based on vocabulary trees is proposed to perform the real-time landmark matching over a large online-collected database (over tens of thousands landmarks). However, the main focus of both methods is how to do the landmark matching on-the-fly. No focus is given to building an accurate or compact landmark database efficiently (with minimum human efforts).

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

In this paper, we examine how to integrate landmark matching to a pre-built (with continuously updating) landmark database to improve overall performance of a visual odometry system. The goal being to obtain accurate global localization (within 10 cm.) indoors and outdoors over a large areas (e.g. multiple square kilometers) can be achieved. The first challenge is to build the landmark database efficiently using a robot or human worn system. Essentially, the human or robot would traverse the area the day before and from that we would build the landmark database. The second challenge is to use the automatically built large landmark database for global localization in real-time.

1.1. Overview

Before describing each component of our system in detail, the overall structure of the paper is first outlined.

We first describe an online pose estimation technique that integrates the traditional incremental-motion based visual odometry with visual landmarks to achieve both accurate and robust global localization. Next we discuss how to automatically build an accurate landmark database using data being collected by a robot or human. The position and orientation of the landmarks is computed using a combination of visual odometry, together with the use of topology inference to choose spatially neighboring frames and bundle adjustment. We also propose a pruning method to reduce the size of the landmark database. Finally we present results with our system over synthetic video data (generated by rendering a textured 3D model) and real video sequences respectively.

In summary, there are three main contributions in our paper: (1) a set of techniques to improve the accuracy of a landmark database; (2) a pruning method to reduce the size of the landmark database and represent it compactly; (3) a real-time methodology that integrates the traditional incremental-motion based visual odometry with visual landmarks to achieve both accurate and robust global localization.

2. Real-Time Localization with A Pre-Built Landmark Database

Figure 1 is a schematic of our proposed two-stage global localization system that combines visual odometry with landmark based localization.

When the system initializes, it locates itself by searching the whole landmark database. This is done via the fast indexing technique using vocabulary tree [8, 2]. Once it locates itself globally, it will update the current camera pose and its uncertainty to estimate a search region. The estimated search region will serve as a geo-spatial constraint to select a smaller set of landmarks for matching in the next

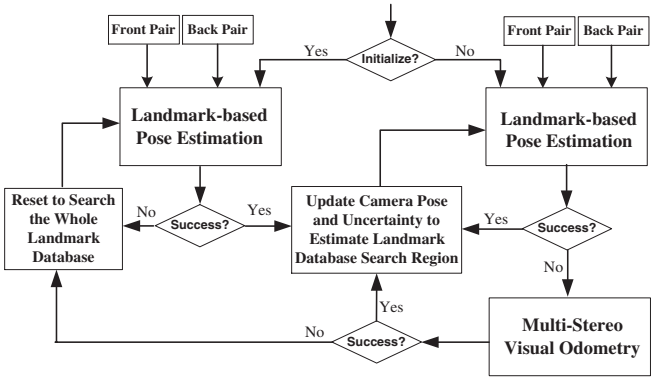


Figure 1. The flowchart of the combined global localization technique.

frame. As a result, both efficiency and accuracy can be increased.

If the system fails to locate via landmark-based localization, the visual odometry system takes over. The visual odometry system localizes by estimating the frame-to-frame relative poses and integrating them over time. The system will return to landmark-based localization as soon as an image is found in the landmark database.

Since the incremental-motion-based visual odometry alone will drift eventually as the error accumulates, it is only locally accurate. On the other hand, landmark-based navigation alone prevents drift, but it requires a huge landmark database to produce smooth and continuous trajectory. Therefore, after integrating them together, this two-stage localization technique employs complementary modules to increase the robustness of the combined system.

Before describing each of the modules in our system, the basic setup of our built system is introduced.

2.1. Multi-Stereo Visual Odometry System

We have developed a helmet-based visual odometry system that consists of two pairs of stereo-cameras mounted in a helmet, one facing forward while the other facing backward as shown in Figure 2(a). Both stereo pairs are synchronized via a developed synchronization circuitry so that they can be captured at the same time. A snapshot of the four synchronized images from both pairs is shown Figure 2 (b), with each camera capturing a gray-scale image with 640×480 pixel resolution.

Similar to the system proposed in [10], our system starts with the Harris corner detection and tracking individually for each stereo pair. Subsequently, the camera pose is estimated from the detected Harris corners. However, during the pose estimation, different from [10], a tightly coupled multi-stereo fusion algorithm as well as local bundle adjustment will be utilized to refine the final camera pose, which will be elaborated in Section 3.2 and 3.4.

In addition, in order to increase the robustness of the

(a) (b)

Figure 2. The helmet-based prototype system: (a) front and back views of the system; (b) captured images of both stereo pairs.

system, an inexpensive IMU unit is added, which is a $2K Crista IMU that drifts over 720^o per hour. Via the Kalman Filtering as proposed in [11], the system is able to probabilistically combine both the IMU and visual odometry measurements together so that it still can produce an accurate pose temporarily when both stereo pairs fail to perform accurate pose estimation due to the lack of features or bad illuminations.

3. How to Improve the Accuracy of the Landmark Database

3.1. Landmark Database

In our system, we define a landmark as a feature point in the scene. Specifically, it is extracted from the image using a Harris corner detector. For each landmark, it is associated with three elements: a 3D-coordinates vector representing its 3D location, a 2D-coordinates vector representing its 2D location in the image and a feature descriptor that characterizes its appearance. Here, the Histogram of Oriented Gradients (HOG) descriptor [12] is used.

In order to make each individual landmark uniquely, the spatial relationship with its neighboring landmarks is also utilized. Therefore, instead of adding each landmark into the landmark database individually, the database is represented as a collection of landmark shots, where a landmark shot is a set of landmarks captured at a specific camera location and view point (or camera pose), and each landmark shot works as a basic unit during landmark matching. For one landmark shot, besides storing all the location (2D+3D) and appearance (HOG) information of each landmark into the database, its camera pose at which they are taken is also stored.

When building a landmark database for a given area, a robot or person wearing our developed multi-stereo rig will move around to collect a set of video sequences. From the collected video sequences, the landmark shots and its camera pose will be estimated and stored into a landmark database. For a large area, we would like this database to be both accurate and compact in size. Therefore, the key tasks during database construction would be estimating the precise camera pose for each landmark shot, estimating the precise 3D location of the landmarks, as well as selecting the representative landmark shots.

First, a method to build an accurate landmark database is introduced. Specifically, we propose to build the landmark database by minimizing a global error measure defined over all the landmarks and the associated camera poses. Figure 3 depicts a flowchart for the approach. The approach uses standard techniques of image matching to establish neighborhood relationships (topology) amongst camera views and global bundle block adjustment for optimizing pose and 3D landmark estimates.

Figure 3. The flowchart of landmark database creation.

Given a video sequence, the algorithm starts with a pose estimate for each image frame using our multi-camera visual odometry algorithm.

3.2. Multi-Stereo Fusing

In [10], a two-stage multi-stereo fusion algorithm is proposed during pose estimation. In the first stage, camera poses are estimated by the front-pair and back-pair individually at each frame. Then during the second stage, a pose selection mechanism is utilized to select the one that produces the smallest cumulative error over both pairs as the final pose of the multi-camera system.

In order to improve the overall accuracy of pose estimation, we extend the above algorithm to perform a multi-stereo fusion that tightly couples both front and backward stereo pairs together to refine the estimated pose.

Specifically, for the front stereo pair (designated to be the master pair), given one consecutive stereo image pair at time frame t, the cost function e_t^m during the pose estimation is defined as a robust function f of the re-projection errors in both the left and right images of the stereo pair as follows:

$$e_t^m(P_t, X^{tm}) = \sum_{j=1}^{k_t^m} \left[f(x_{tj}^{ml}, P_t X_j^{tm}) + f(x_{tj}^{mr}, P_c^m P_t X_j^{tm}) \right] \tag{1}$$

where P_t is the pose of the left camera of the stereo pair at frame t, and k_t is the number of feature points, whose 3D coordinates are X_j and 2D coordinates in the left and right images are x_{tj}^l and x_{tj}^r respectively, P_c represents the fixed relative pose between left and right cameras or the extrinsic parameters of the stereo pair. To make it robust against the outliers, the Cauchy-based robust cost function $f(x, y) = log(1 + \|x - y\|^2/\sigma^2)$ is utilized, where σ is the standard deviation parameter.

Similarly, for the backwards stereo pair (designated to be the slave camera), given the consecutive stereo image pair

at time frame t, the cost function e_t^s can be represented as follows:

$$e_t^s(P_t, X^{ts}) = \sum_{j=1}^{k_t^s} \left[f(x_{tj}^{sl}, P_t^s X_j^{ts}) + f(x_{tj}^{sr}, P_c^s P_t^s X_j^{ts}) \right] \tag{2}$$

where $P_t^s = P_{ms} P_t P_{ms}^{-1}$ and P_{ms} is the fixed relative pose between the front stereo-pair and back stereo-pair. Specifically, P_{ms}, P_c^m and P_c^s are calibrated in advance using standard camera calibration methods.

Therefore, when both the front and back stereo-pairs are integrated together, for the consecutive stereo image pairs at frame t, the cost function e_t is expressed as

$$e_t(P_t, X^{tm}, X^{ts}) = e_t^m(P_t, X_{j=1,\ldots,k_t^m}^{tm}) + e_t^s(P_t, X_{j=1,\ldots,k_t^s}^{ts}) \tag{3}$$

3.3. Topology Inference

Global consistency of reconstructed landmarks is enforced by minimizing the total global error in the overlapped regions along a path with respect to the pose for each frame and 3D landmark parameters. However, a key issue is how to identify pairs of frames that overlap. We employ a topology inference algorithm to automatically find a set of spatially overlapped images in a large set of images. Figure 4 illustrates the flowchart of the topology inference algorithm.

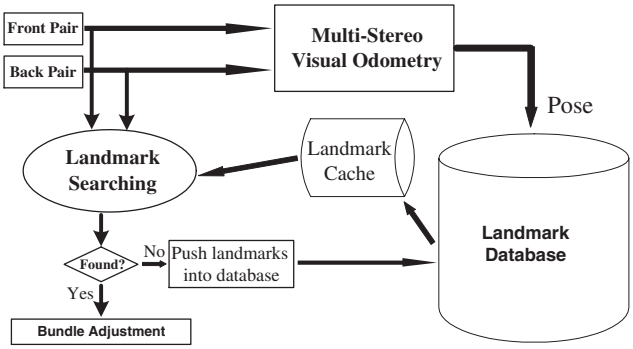


Figure 4. The flowchart of topology inference.

As shown in 4, at each frame t, a set of feature points are first extracted from the front and back stereo pairs individually to establish a set of 3D-2D feature point correspondences. From the set of 3D-2D feature point correspondences, we estimate pose, P_t, using the multi-stereo fusion technique described above. Subsequently, given the estimated camera pose P_t, a radius for a 3D search region is obtained automatically via the uncertainty based on the estimated drift rate.

Subsequently, all the landmarks acquired at the positions within the obtained search region are selected from the landmark database to form a landmark cache. For efficient landmark retrieval, the landmark database is indexed via a pre-built vocabulary tree [8]. Images from the front and back stereo pairs are matched to the vocabulary tree-indexed landmark cache to obtain a list of top-ranked landmarks. These are subsequently refined by imposing the epipolar constraint [3]. Finally, we utilize the number of matched landmarks to characterize the matching score for each image, and the image with the highest score that satisfies a predefined threshold is returned as a successful match.

For efficiency, we store the mapped node indices at each layer in the tree for each landmark in the landmark database so that the visual word quantization need be done only once.

With the topology inference technique, all the revisits to locations already stored in the landmark database are identified successfully, regardless of the complexity of the environment or the number of overlaps along the system's path during the process of landmark collection. Once a re-visit or an overlapped image pair is found, all the frames between the reference frame and the current frame will be used for optimization using bundle adjustment. This part is described next.

3.4. Bundle Adjustment

Bundle adjustment is used to find optimized estimates of poses for the landmark images and the corresponding 3D landmarks. Specially, given a set of N frames starting at time $t = 1$, the final cost function e is expressed as:

$$e(P_{t=1,\ldots,N}, X^m, X^s) = \sum_{t=1}^{N} e_t(P_t, X^{tm}, X^{ts}) \tag{4}$$

where $X^m = X^{1m} \bigcup \ldots \bigcup X^{Nm}$ and $X^s = X^{1s} \bigcup \ldots \bigcup X^{Ns}$.

Bundle adjustment minimizes the final cost function e over the set of N frames by solving the camera poses P_i and the 3D feature coordinates X_j:

$$\arg\min_{P_i, X_j} e(P_{i=1,\ldots,N}, X_{j=1,\ldots,K}) \tag{5}$$

where $X_{j=1,\ldots,K} = X^m \bigcup X^s$, and K is the total number of feature points.

Solving the above equation is a non-linear minimization problem, which is solved using the iterative Levenberg-Marquardt non-linear least-squares approach [3]. During the minimization, the initial values are taken from the estimated poses from the multi-stereo fusion described in Section 3.2 directly.

Bundle adjustment provides a globally optimized landmark. However, for a fixed large-scale site, the size of the built landmark database is usually very large. For example, in order to cover an area with one square kilometers, if we collect one landmark shot at every meter apart, there will be 10^6 landmark shots totally. In addition, if we assume that each landmark shot contains 300 landmarks, then as listed

Table 1. The size of landmarks on the disk .

Size	HOG	2D	3D	Indexes
a LM (bytes)	128	16	24	25
an image (300) (MB)	0.0366	0.0046	0.0069	0.0072
a database (10^6) (GB)	35.7422	4.4922	6.7383	7.0313

in Table 1, there will be at least 54.004Gigabytes in total, which is quite large.

On the other hand, given an image collected at a fixed location, all the images collected nearby may not need to be stored into the landmark database. Therefore, it is necessary to reduce the size of the landmark database and build a compact as well as efficient database.

3.5. How to Build A Compact and Efficient Database

The key questions to be addressed for a practical landmark based localization system covering many square kilometers include: (1) How many landmarks need to be collected? (2) What is the density of the landmarks? (3) How much translational and rotational sampling is adequate for a given level of localization performance?

In order to answer these design questions, we use the metric of accuracy of localization with landmark matching as an output parameter with respect to variation in the parameters governing the specified questions. We will systematically vary the density of landmarks and use the output metric to evaluate the quality of localization.

For our study, we define a 10×10 meter grid of locations as shown in Figure 5 (a) on the ground with precisely marked positions. At each location, the camera-rig is rotated 32 times along the pan-direction (horizontally) with a sampling of 11.5 degrees to collect 32 stereo image pairs. One pair of sample images taken at two consecutive angles for the same location is shown in Figure 5 (b). The figure shows a large overlapped region between these two images. In total 3200 images are collected.

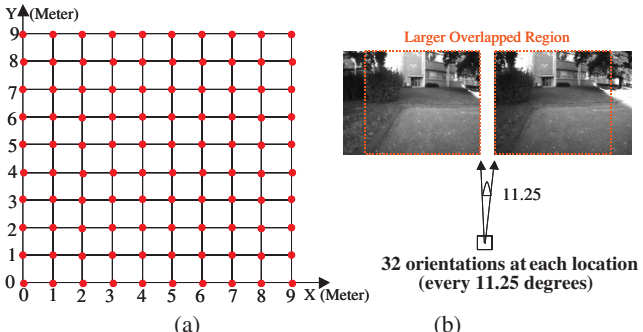


Figure 5. (a) The topology of the 10 by 10 grid points; (b) A pair of sample images taken at two consecutive angles for a same location.

Subsequently, each image is matched to all the 3200 images using the landmark matching algorithm. The relative pose is thus estimated for each image pair from the matched landmarks. Since the ground-truth of the relative pose between any image pair is known, an error is computed for each estimated relative pose as the Euclidean distance between the ground truth and the estimate. We define a successful match or landmark localization as a localization error less than a pre-defined threshold (we use 5cm in our experiments).

We summarize the results in Table 2. The independently varied parameters are the translational distance and the orientation difference between consecutive landmark images. Each entry of the table shows the percentage of images that were localized with an error less than 5cm. For instance, the table shows that for an angular distance of 22.5 degrees and displacement of 2m., about 71% of the database images were localized with less than 5 cm. localization error.

If in the operating area, our system can combine frame-to-frame visual odometry with landmark based localization done over about 30% of the area, then it is adequate to sample the landmark images at 6 meters translational and about 40 degrees orientational separation (within 3-meter displacement and 22.5 angular distance in Table 2). In other words, the systematic quantitative results provide us with an empirical basis for choosing an operating point in the design of a localization system.

Table 2. The successful rate of the landmark-based localization under different displacement and view-angle changes (under 5cm pose estimation accuracy) .

Angle Dev.	Distance Deviation (meter)					
(degree)	0	1	2	3	4	5
0	1	0.998	0.914	0.551	0.394	0.331
11.25	1	0.993	0.883	0.476	0.330	0.279
22.50	1	0.928	0.713	0.337	0.211	0.158
33.75	0.337	0.459	0.322	0.098	0.077	0.069
45.00	0	0.074	0.035	0.023	0.021	0.015
56.25	0	0.017	0.019	0.005	0.004	0.003

4. Experiment Results

4.1. Synthetic Video Generation

In reality, due to the difficulty of collecting the full set of ground-truth camera pose data (three translations and three angles) at each frame during experiments, how to evaluate the performance of the visual odometry system becomes an important issue. So far, most of the existing works either use loop closure [7, 10] or DGPS [7, 5] to evaluate the performance. However, DGPS itself may not be accurate enough to serve as the ground-truth, and loop closure may not be able to reflect the accuracy at those positions other than the

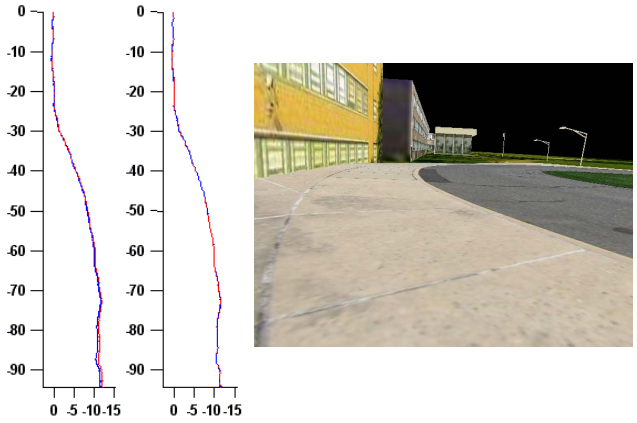
closure points. Therefore, in our experiments, a set of synthetic video sequences with the perfect camera pose data at each image frame are generated. The method to generate these synthetic videos is described briefly as follows.

Specifically, the first step is to build a 3D model of a real physical site that covers both indoors and outdoors. Then a person wearing our helmet-based system walks around the site and the trajectories (both translations and orientations) of the camera motion are estimated by the system. Once these real walking trajectories of the camera motion are obtained, they are used to control a virtual stereo-camera-rig to move through the built 3D site model exactly like the person, and all the images viewed by the virtual camera-rig will be captured simultaneously.

Via the above approach, different types of synthetic video sequences can be obtained, with the obtained real camera pose data serving as the ground-truth. Since these synthetic video sequences are very similar to the real collected video sequences, they are perfectly suitable for the performance evaluation of any visual odometry system.

4.2. Visual Odometry Improvements

In order to show the performance of our new set of proposed techniques, a synthetic video sequence of a multi-stereo-pair rig (it's configured exactly same to our helmet system) containing 1984 frames is created from a 106.23-meter long real trajectory of our system as shown Figure 6 (a). A snapshot of the left camera of the front pair is shown in Figure 6 (c).



(a) (b) (c)

Figure 6. (a) The ground-truth trajectory of the camera motion (blue), the estimated trajectory by visual odmetry with old camera fusion method only (red); (b) The estimated trajectory by visual odometry with new camera fusion + bundle adjustment (red);(c) A snapshot of the left-camera.

With the use of fusion method proposed in [10], from the synthetic video sequence, the visual odometry system outputs the estimated trajectory shown in Figure 6 (a). Apparently, small error accumulates gradually as the system moves over time and there is obvious deviation towards the end of the sequence. As listed in Table 3, the average distance deviation at each frame is 0.2754 meter, and the final deviation at the ending frame is 0.5312 meter after travelling 106.23 meters.

However, with the use of our proposed camera fusion method, the average deviation drops to 0.1437 meter and the final deviation at the ending frame drops to 0.2882 meter. Finally, if we further turn on the bundle adjustment, the average deviation drops to 0.0696 meter at each frame, and the final deviation at the ending frame is only 0.1246 meter, which is almost 5 times improvement.

Table 3. The deviations between the estimated trajectory and the ground-truth trajectory for different techniques.

Dev.	Old Fusion	New Fusion	New Fusion+Bundle
Avg.	0.2754	0.1437	0.0696
Final	0.5312	0.2882	0.1246

4.3. The Combination of Visual Odometry and Landmarks

In this experiment, the performance of the integrated pose estimation algorithm is reported on both synthetic and real video sequences.

4.3.1 Synthetic Video Sequences

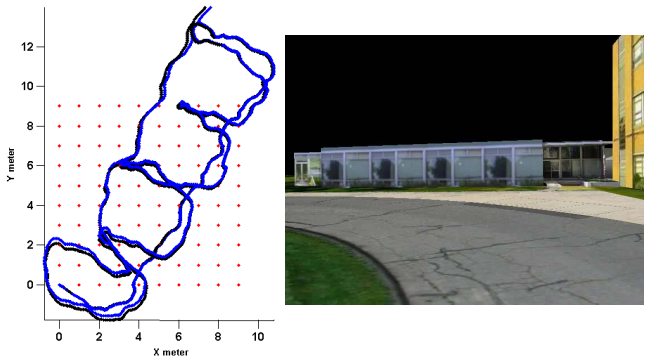
A synthetic video sequence (only front pair this time) that contains 1751 frames is generated from a 91.18-meter long real trajectory as shown in Figure 7 (a). Figure 7 (b) shows a left-camera snapshot of the virtual one-pair stereo-rig. In addition, a set of landmarks are collected at a set of grid positions that are 1-meter apart as shown as the "red" dots in Figure 7 (a). Specifically, at each position, 90 stereo-images are collected at every 4 degrees along the pan direction (horizontally).

Without the use of landmark database, the estimated trajectory by the visual odometry only is shown as the "blue" curve in Figure 7 (a), which clearly shows the growing drift as the camera moves over time. As listed in Table 4, the average of the computed distance deviation at each frame is 0.2239 meter, and the deviation at the ending frame is 0.4187 meter.

Table 4. The deviations between the estimated trajectory and the ground-truth trajectory under different landmark settings.

Dev.	front-pair	1m+4°	2m+4°	3m+4°	4m+180°
Avg.	0.2239	0.0200	0.0210	0.0251	0.0434

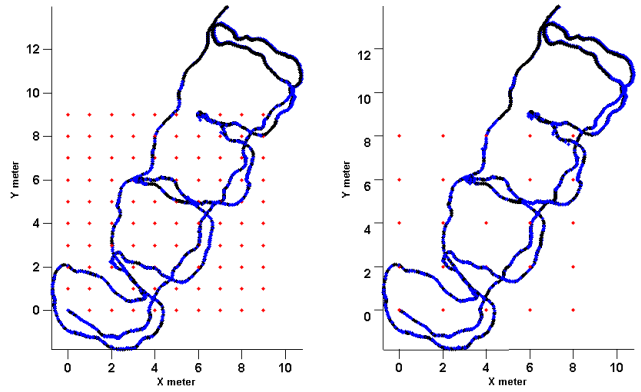
With the use of landmark database, the trajectory estimated by the visual odometry together with the visual land-



(a) (b)

Figure 7. (a) The ground-truth trajectory of the camera motion (dark), the estimated trajectory by visual odmetry only (blue) and the positions where landmarks are collected (red). (b) A snapshot of the left-camera.

marks is shown as the "blue" curves in Figure 8. Specifically, the average of the computed distance deviation at each frame is 0.0200 meter when using all the visual landmarks (9000 landmark shots) collected at 1-meter apart with 4-degree interval, and its deviation at the ending frame is 0.0542 meter. When using the visual landmarks (2250 landmark shots) collected at 2-meter apart with 4-degree interval, the average of the computed distance deviation at each frame is 0.0210 meter and its deviation at the ending frame is 0.0585 meter. Even when there are only 18 landmark shots collected at 4-meter apart with 180-degree interval, the average of the computed distance deviation at each frame is only 0.0434 meter. It clearly shows the significant improvement when the visual landmarks are integrated.

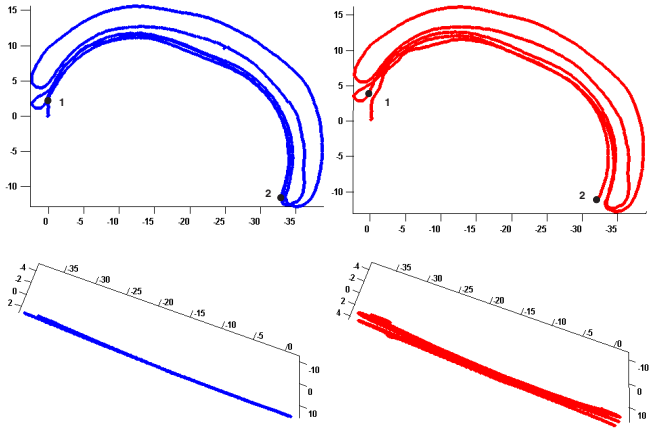


(a) (b)

Figure 8. (a) The estimated trajectory by visual odmetry+visual landmarks collected at 1-meter apart+4° interval (blue). (b) The estimated trajectory by visual odmetry+visual landmarks collected at 2-meter apart+4° interval (blue).

4.3.2 Real Video Sequences

Real video sequences were also collected to validate our proposed techniques. Specifically, during the experiments, the person wearing our developed system tried to collect the landmarks for a specific area. In addition, two locations are marked on the ground and the person had to travel through them twice. The total collected video sequences contain 3150 frames, and the total travelled path is around 311 meters. The estimated trajectory by the multi-stereo visual odometry without our proposed techniques is shown in Figure 9 (b). From the front view shown Figure 9 (b), you can see that there are large distance deviations at the revisited locations marked by a small dark dot. In addition, from the side view of its trajectories, we can see that there are obvious deviations along the vertical directions.



(a) (b)

Figure 9. (a) The front and side views of the estimated camera trajectory with our proposed techniques, where the positions marked by a small dot represent the revisited locations. (b) The front and side views of the estimated camera trajectory without our proposed techniques.

Database Accuracy Refinement. With the use of the proposed techniques, all the overlapped regions along the trajectories will be first detected. Then with the new camera fusion and bundle adjustment techniques, the pose of the camera will be refined off-line. Figure 9 (a) shows both the front and side view of the estimated trajectories after applying our proposed techniques. From its side view, you can see the trajectory becomes very flat and there is almost no deviation along the vertical direction (since the person was walking on the flat ground). In addition, from the front view of its trajectory, you can see that both revisited locations are aligned very well. Table 5 lists the distance deviations between both revisits in the estimated trajectories, and it tells that the camera pose accuracy is improved quite significantly with the proposed techniques.

Table 5. The distance deviations at two revisits (meters) .

Dev.	Without New Methods	With New Methods
Point 1	1.2074	0.1607
Point 2	1.6272	0.3028

Database Size Pruning. If we want to create a landmark

database for the above region, we do not need to put all the landmarks extracted from the front and back pairs at each location into it, especially when the region is over multiple square kilometers. In order to eliminate all the redundant landmarks and represent it compactly, the distance interval for the locations where the landmark shot is taken is set to be 2 meters and the angle interval is 15 degrees via Section 3.5. Therefore, instead of adding all 3150-shots of landmarks into the database, it only needs to add 79-shots as shown in Figure 10.

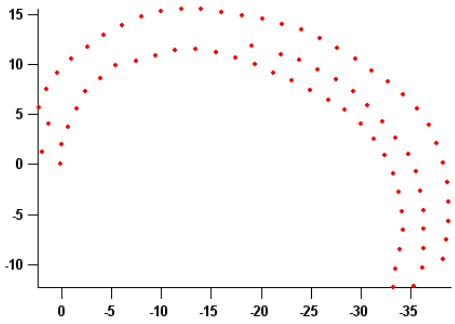
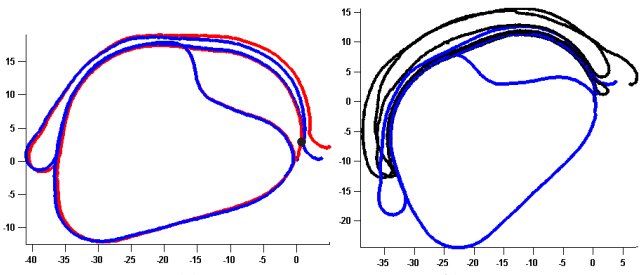


Figure 10. The locations where the landmark shots are added into the landmark database.

Localization Via Integrated Pose Estimation. During test, a new video sequence that contains 2376 frames was recorded while the person travelled around 245 meters near the region where the landmark database is collected. With no use of the above pruned landmark database, in the estimated trajectory by the visual odometry alone, the deviation at the revisit location marked by a dark dot is 1.9559 meters. However, once we integrate the above pruned landmark database, its deviation drops to only 0.2227 meter. The deviation difference can be easily observed from the Figure 11 (a). In addition, the trajectory can be automatically aligned to the coordinate system of the landmark database as shown in Figure 11 (b).



(a) (b)

Figure 11. (a) The estimated camera trajectories with visual odometry alone (blue) or with our integrated approach (red), where the position marked by a small dot represent the revisited location. (b) The estimated camera trajectory with our proposed techniques (blue) is aligned with the trajectory of landmark database (dark) perfectly.

5. Conclusion

In this paper, we have presented a set of techniques to reduce the global error during the process of landmark database building for an unknown but fixed environment using the visual odometry. The global error is reduced via the integration of a new multi-stereo fusion algorithm, an efficient topology inference algorithm as well as the dynamic bundle adjustment. Once an accurate landmark database is built for the unknown environment, it is further reduced to a realistic size and subsequently integrated into the visual odometry for navigation within it repeatedly. Experiments on both synthetic and real video sequences confirm the effectiveness of our proposed techniques.

References

[1] A. Davison. Real-time simultaneous localization and mapping with a single camera. In *IEEE Conference on ICCV'03*, 2003.

[2] F. Fraundorfer, H. Stewénius, and D. Nistér. A binning scheme for fast hard drive based image search. In *IEEE Conference on CVPR'07*, 2007.

[3] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[4] A. Johnson, J. Montgomery, and L. Matthies. Vision guided landing of an autonomous helicopter in hazardous terrain. In *IEEE Conference on ICRA'05*, 2005.

[5] K. Konolige, M. Agrawal, R. C. Bolles, C. Cowan, M. Fischler, and B. Gerkey. Outdoor mapping and navigation using stereo vision. In *International Symposium on Experimental Robotics*, 2006.

[6] A. Levin and R. Szeliski. Visual odometry and map correlation. In *IEEE Conference on CVPR'04*, 2004.

[7] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry. In *IEEE Conference on CVPR'04*, 2004.

[8] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *IEEE Conference on CVPR'06*, 2006.

[9] C. Olson, L. Matthies, M. Schoppers, and M. Maimone. Rover navigation using stereo ego-motion. *Robotics and Autonomous Systems*, 43(4):215–229, 2003.

[10] T. Oskiper, Z. Zhu, S. Samarasekera, and R. Kumar. Visual odometry system using multiple stereo cameras and inertial measurement unit. In *IEEE Conference on CVPR'07*, 2007.

[11] S. Roumeliotis, A. Johnson, and J. Montgomery. Augmenting inertial navigation with image-based motion estimation. In *IEEE Conference on ICRA'02*, 2002.

[12] S. Se, D. Lowe, and J. Little. Vision-based global localization and mapping for mobile robots. *IEEE Transactions on Robotics*, 21(3):364–375, 2006.

[13] B. Williams, G. Klein, and I. Reid. Real-time SLAM relocalisation. In *IEEE Conference on ICCV'07*, 2007.

[14] Z. Zhu, T. Oskiper, S. Samarasekera, H. Sawhney, and R. Kumar. Ten-fold improvement in visual odometry using landmark matching. In *IEEE Conference on ICCV'07*, 2007.