Estimating Camera Position and Posture by Using Feature Landmark Database

Motoko Oe[1], Tomokazu Sato[2], and Naokazu Yokoya[2]

[1] IBM Japan
[2] Nara Institute of Science and Technology, Japan

Abstract. Estimating camera position and posture can be applied to the fields of augmented reality and robot navigation. In these fields, to obtain absolute position and posture of the camera, sensor-based methods using GPS and magnetic sensors and vision-based methods using input images from the camera have been investigated. However, sensor-based methods are difficult to synchronize the camera and sensors accurately, and usable environments are limited according to selection of sensors. On the other hand, vision-based methods need to allocate many artificial markers otherwise an estimation error will accumulate. Thus, it is difficult to use such methods in large and natural environments. This paper proposes a vision-based camera position and posture estimation method for large environments, which does not require sensors and artificial markers by detecting natural feature points from image sequences taken beforehand and using them as landmarks.

1 Introduction

The recovery of camera position and posture is required in a number of different fields such as augmented reality and robot navigation. In these fields, to obtain absolute position and posture of the camera, sensor-based methods using GPS and magnetic sensors[1, 2, 3, 4, 5] and vision-based methods using input images from the camera[6, 7, 8, 9, 10, 11, 12, 13] have been investigated. However, sensor-based methods are difficult to synchronize the camera and sensors accurately, and usable environments are limited according to selection of sensors. Vision-based methods can be classified in two groups: Methods using markers and methods without markers. Methods using markers need to allocate many artificial markers in the environment. Thus, it is difficult to use such methods in large and natural environments. On the other hand, maker-less methods are also proposed. Most of these methods track natural features and estimate camera position and posture by concatenating transformations between adjacent frames. Thus, these methods are inappropriate for long sequences because estimation errors accumulate and causes drift. Therefore, methods using prior knowledge of the environment are recently proposed[12, 13]. Lepetit et al.[12] have proposed a method using the 3-D model of the environment. It is robust to large camera displacements, extreme aspect changes and partial occlusions, but

H. Kalviainen et al. (Eds.): SCIA 2005, LNCS 3540, pp. 171–181, 2005.

their method is limited to an environment that can be modeled manually, so it is difficult to use in an outdoor environment. Gordon et al.[13] have proposed a method which constructs a sparse metric model of the environment, and performs model-based camera tracking. It does not require camera pre-calibration nor prior knowledge of scene geometry, but it is difficult to use the method in large environments because the error will accumulate when reconstructing the 3-D model.

In this research, we propose a camera position and posture estimation method for large environments, which is based on detecting natural feature points from image sequence taken beforehand and using them as landmarks, and thus does not require sensors and artificial markers. Our method is composed of two stages. In the first offline stage, we reconstruct the environment from omni-directional image sequences. Then, a feature landmark database is created, and natural feature points extracted from the image sequences are registered as landmarks. The second stage is a sequential process, and camera position and posture which do not include significant cumulative errors are estimated by determining the correspondence between the input image and the landmarks. In Section 2, we describe the first stage of our method, which specifies the construction of the feature landmark database. Section 3 describes the position and posture estimation method using the feature landmark database created in Section 2. Section 4 shows the experiment result, and conclusion is shown in Section 5.

2 Constructing Feature Landmark Database

This section describes the first stage of our method, which specifies the construction of the feature landmark database. In our method, natural feature points detected from omni-directional image sequences are used as landmarks. We first take an omni-directional image sequence by walking through the environment with an omni-directional camera. Secondly, we obtain 3-D coordinates of landmarks and camera position and posture of the omni-directional camera from the image sequence. Lastly, the landmark database is created semi-automatically using the 3-D coordinates of the natural features, the omni-directional images, and its camera path. In the following sections, the elements of the feature landmark database are listed, and the way for constructing landmark database is detailed.

2.1 Landmark Information Acquisition by 3-D Reconstruction of Environment

Feature landmark database consists of a number of landmarks as shown in Figure 1. These landmarks are used to be matched to natural feature points from an input image in the second stage in order to estimate the camera position and posture of an input image. Each landmark retains the 3-D coordinate of itself(1), and several information for different camera positions(2). Information for different camera positions consists of four items: (A)camera position and posture of the omni-directional camera, (B)multi-scale image template of the

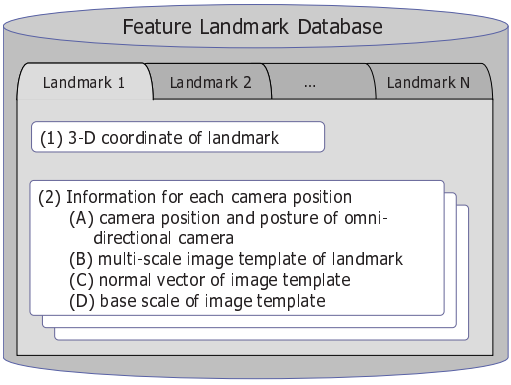


Fig. 1. Elements of feature landmark database

landmark, (C)normal vector of the image template, and (D)base scale of the image template.

To obtain the landmark information listed above (Figure 1), 3-D reconstruction of the environment is required. First, we reconstruct the environment from omni-directional image sequence and obtain (1)3-D coordinate of the landmark and (A)camera position and posture of the omni-directional camera. Next, we generate (C)normal vector of the image template, (D)base scale of the image template, and (B)multi-scale image template of the landmark.

3-D Reconstruction of the Environment from Omni-directional Image Sequence. Our extrinsic camera parameter estimation is based on structure-from-motion[14]. In this method, first, markers and natural features in the image sequences captured by an omni-directional multi-camera system are automatically tracked and then the reprojection errors are minimized throughout the sequences. Thus, we can obtain extrinsic camera parameter of the camera system and 3-D coordinates of natural features in absolute coordinate system based on the markers without accumulative estimation errors, even in a large and complex environment. Note that, in our method, intrinsic camera parameters of the camera system are assumed to be known.

Creating Landmarks. Landmark database is automatically created using the result of 3-D reconstruction. The elements of landmarks are created by the following procedures.

(1) 3-D coordinate of landmark
We use natural features detected by Harris operator[15] from the omni-directional image as feature landmarks. The 3-D coordinate of the landmark is estimated by the 3-D reconstruction of the environment, and is obtained by the world coordinate system. The X and Y axes of the world coordinate system are aligned to the ground and Z axis is vertical to the ground.

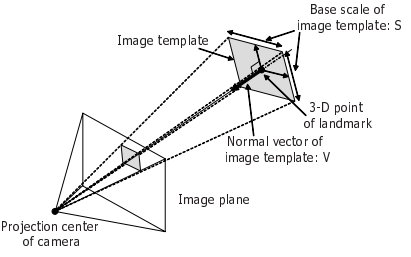


Fig. 2. Landmark and its image template

(2) Information for each camera position

Landmarks are used to determine the correspondence between feature points in an input image and 3-D coordinates of the landmarks. In this research, information from several different camera positions is obtained and used for a robust matching of the landmarks, considering the aspect changes of image patterns depending on the shooting position.

(A) Camera position and posture of omni-directional camera: Camera position and posture are retained by the world coordinate system, and are used to select landmarks from the database to match with the input image. We use the extrinsic camera parameter estimated in Section 2.1.

(B) Multi-scale image template of landmark: Image template is created by projecting the omni-directional image to a plane which is vertical to the line through the landmark's 3-D coordinate and the projection center of the camera, as shown in Figure 2. The lens distortion is removed from the image template. First, the normal vector V and the base scale S shown in Figure 2 are precalculated. Then, to create an image template of a base scale, a square plane which implements the following assumptions is configured.

- Landmark is allocated on the center of the plane
- The plane is vertical to the normal vector V
- The plane size is $S \times S$ in the world coordinate system
- The plane's X axis is parallel to the X-Y axis of the world coordinate system

Next, the previously defined plane is divided into an $N \times N$ grid where $N \times N$ is the resolution of image templates. Each center of the grid is projected to the omni-directional images by its 3-D coordinate, and the color value of the projected pixel is set as the template's pixel color value. In the same way, double and quadruple scale image templates are created for each camera position. We define single, double, and quadruple scale image templates as a set of multi-scale image templates.

(C) Normal vector of image template: As shown in Figure 2, the normal vector of the image template is defined as the normal vector of the plane which is vertical to the line through the landmark's 3-D coordinate and

the omni-directional camera's position. It is used to select an image template for matching from several image templates taken by different camera positions. Normal vector of the image template is simply acquired as a normalized vector from the landmark's 3-D coordinate to the omni-directional camera's position.

(D) Base scale of image template: As shown in Figure 2, the scale of the image template is the size of the plane used to create the image template. The scale size is retained in the world coordinate system, and the base scale is determined so that the resolution of the omni-directional image and the image template becomes nearly equal.

3 Camera Position and Posture Estimation Using Database

3.1 An Overview of Proposing Method

This section describes a camera position and posture estimation method based on the feature landmark database. The initial camera position and posture are estimated in the first frame. Then, using the previous camera position and posture, landmarks are selected from the landmark database(step 1). Detecting natural features from the input image and matching them with the landmark image templates, the correspondence between landmark and input image is established(step 2). Lastly, camera position and posture are estimated from the correspondences between landmarks and input image(step 3). In this paper, we assume that initial camera position and posture are given. The following sections describe these steps. Figure 3 shows the data flow of the estimation process.

3.2 Selecting Landmark from Landmark Database

To find a correspondence with the input image, several landmarks are selected from numerous landmarks in the landmark database. Furthermore, to handle partial occlusions and aspect changes, an image template with the nearest appearance to the input image is chosen from a number of image templates stored in the database. Considering the appearance, it is ideal if the image template and input image are taken in the same position. However, the camera position and posture of the input image are not yet estimated, so we use the camera

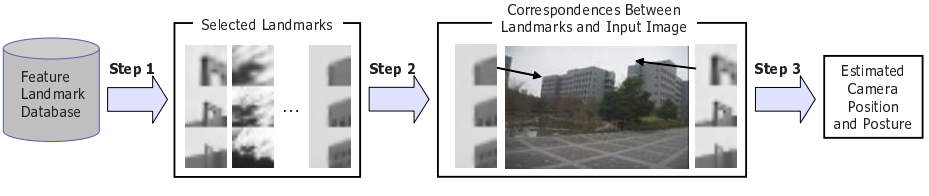


Fig. 3. Data Flow of Camera Position and Posture Estimation Process

position and posture of the previous frame as a replacement. Landmarks satisfying the following requirements are selected to make correspondence with the input image.

(requirement 1) Landmark has to be in the image when projecting its 3-D coordinate using the previous camera position and posture: We project the landmark's 3-D coordinate on the input image by using previous camera position and posture. Only the landmarks projected on the input image are selected.
(requirement 2) Distance between the camera position when the landmark was taken and the camera position when the input image was taken should be under a given threshold: We actually calculate the distance between the camera position when the landmark was taken and the camera position of the previous frame, and select landmarks under the threshold.
(requirement 3) Angle between the normal vector of the image template and the vector from landmark to camera position when the input image was taken should be under a given threshold and is the minimum for all the image templates of the landmark: We select the image template if angle θ between the normal vector of the image template and the vector from landmark to previous camera position is the minimum for all the image templates of the same landmark. If the angle θ of the selected image template is over the threshold, that landmark is not selected.
(requirement 4) Landmark must not be adjacent to already selected landmarks: First, the input image is divided into a grid. The landmarks on the input image are then projected to the image plane by using the previous camera position and posture, and only one landmark per each grid are selected.

Landmarks that implement the requirement 1 are selected first. Then, the selected landmarks are narrowed down to a fixed number of landmarks by the ascending order of the distance mentioned in the requirement 2. From the list of landmarks, landmarks with smaller angles in the requirement 3 are picked up one by one, and the selecting process is repeated until a fixed number of landmarks that implement the requirement 4 are chosen.

3.3 Determining Correspondence Between Landmark and Input Image Feature

In this step, the correspondence between selected landmarks and features in an input image are computed. First, natural features are detected from the input image using interest operator, and are then corresponded with the selected landmarks using template matching.

Detecting Natural Feature from Input Image. To find the correspondence between landmarks and input image, natural feature points are detected from the input image by Harris operator[15]. In this step, a landmark is projected to the input image, using previous camera position and posture. On the assumption that the corresponding point for the landmark exists near the projected point,

natural feature points are detected within a fixed window surrounding the projected point. The detected feature points are listed as correspondence candidates of the landmark.

Matching Between Landmark Image Template and Input Image. In this step, each landmark is compared with its correspondence candidates. First, an image pattern is created for each natural feature point listed as a correspondence candidate. Next, the landmark image template is compared with each image pattern by normalized cross correlation. Then, the feature point with the most correlative image pattern is selected, and its neighboring pixels are also compared with the landmark as correspondence candidates. Lastly, the most correlative feature point is corresponded with the landmark.

3.4 Camera Position and Posture Estimation Based on Established Correspondences

Camera position and posture are estimated from the list of 2-D and 3-D correspondences acquired from the matching between landmarks and input image. First, outliers are eliminated by RANSAC[16]. Next, camera position and posture are estimated using only the correspondences that are supposed to be correct. Finally, camera position and posture with the minimum reprojection error are computed by using non-linear least square minimization method.

4 Experiments

To verify the validity of the proposed method, we actually have created a landmark database of an outdoor environment and have carried out experiments of estimating camera position and posture from an outdoor image sequence.

4.1 Experiments in an Outdoor Environment

First, an outdoor image sequence is captured by an omni-directional multi-camera system(Point Grey Research Ladybug) as shown in Figure 4 for constructing a landmark database. In this experiment, intrinsic parameters of the camera system was calibrated by Ikeda's method in advance[17]. Captured image sequence consists of 1,250 frames long with 6 images per each frame(totally 7,500 images). Then, landmark database is created by estimating camera path and 3-D coordinates of natural features[14]. For every landmark, multi-scale image template with three different scales of 15 $\times$ 15 pixels each, is created per each camera position. The number of landmarks created in this experiment is about 12,400, and the number of image templates created per each landmark is 8 on average. Figure 5 shows a part of estimated camera path and 3-D coordinates of natural feature points in constructing the landmark database.

Next, we have captured a 1,000 frames long monocular video image sequence(720 $\times$ 480 pixels, progressive scan, 15fps) with a video camera(SONY

Fig. 4. Omni-directional camera system Ladybug and images taken by ladybug

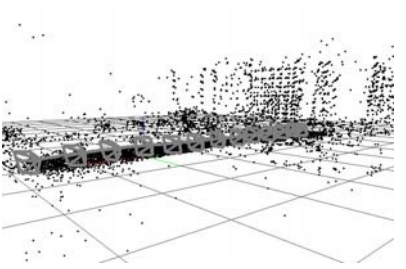
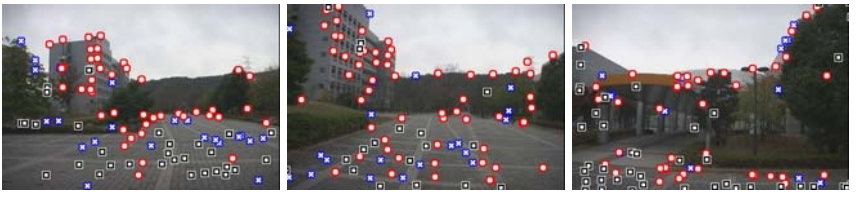


Fig. 5. Estimated camera path and 3-D coordinates of natural feature points



(a)1st frame (b)600th frame (c)1000th frame

Fig. 6. Landmarks used for camera position and posture estimation

(a)1st frame (b)200th frame (c)400th frame

Fig. 7. Match move based on estimated camera position and posture (*http* : *//yokoya.naist.jp/pub/movie/oe/outdoor.mpg*)

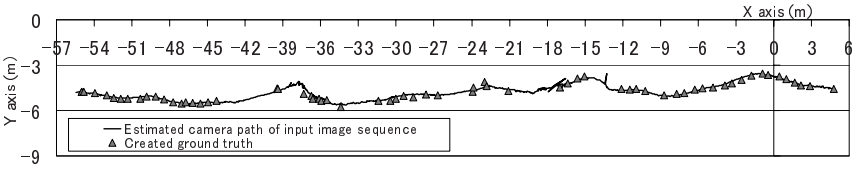


Fig. 8. Estimated camera path and the ground truth

DSR-PD-150) and camera position and posture are sequentially estimated using the landmark constructed earlier. In this experiment, initial position and posture of the camera is manually specified in the first frame of the input sequence. The

maximum number of landmarks selected from the database to correspond with input image is 100 per frame, with the window size for detecting natural features from input image is 120 × 60 pixels, the number of RANSAC iterations is 500. As a result, processing time for a frame was about 4 seconds with a PC(CPU Pentium4 3GHz, Memory 1.5GB).

Figure 6 shows the landmarks used for camera position and posture estimation. In this figure, squares indicate feature landmarks rejected by similarity measure, crosses are also rejected by RANSAC, and circles are inliers of feature landmarks. The inliers are used for camera position and posture estimation. Figure 7 shows the result of match move; matching virtual 3-D objects to the camera movements using the estimated camera position and posture. It can be observed that the CG person drawn in geometrically correct positions throughout the sequence(http://yokoya.naist.jp/pub/movie/oe/outdoor.mpg).

4.2 Quantitative Evaluation

We have evaluated the estimation accuracy by comparing the estimated camera position and posture with the ground truth. The ground truth is created by measuring 3-D position of feature points using a 3-D laser measure named "Total Station" and manually specifying their correspondence with each input image, and solving PnP(Perspective n-Point) problem from the correspondence. The ground truth is created for every 10 frames, except for the following frames: frames which could not obtain enough measured points because the scene is interspaced with natural objects, and frames in which the reprojection error of the obtained ground truth is over 1.5 pixels.

As a result, camera position estimation error was 220mm on average, and estimation error of the optical axis was approximately 0.37 degrees. Figure 8 shows the result of estimated camera parameter and the ground truth. Camera path is estimated from 1,000 frames long image sequence, and the X and Y axes of the figure corresponds to the X and Y axes of the world coordinate system. It shows that the estimated camera path is generally smooth and the estimation error does not accumulate during the whole sequence. However, there were some frames with larger estimation errors than other frames. In these frames, landmarks used for camera position and posture estimation are tended to be aligned lopsidedly in the input image, or only the landmarks far from the camera position are used. Therefore, it is necessary to investigate a method for selecting landmarks from the landmark database to raise the accuracy of our method.

5 Conclusion

In this paper, we have proposed a camera position and posture estimation method for large environments by detecting natural feature points from image sequence taken beforehand and using them as landmarks. The proposed method provides image-based localization. We create a feature landmark database by reconstructing the environment from image sequences in advance. Camera posi-

tion and posture are estimated by determining the correspondence between the input image and the landmarks. In experiments, we have successfully demonstrated camera position and posture estimation from an image sequence of an outdoor environment, and have confirmed that the estimation result does not include cumulative errors. As a future work, camera position and posture estimation needs to be performed in real-time for use in augmented reality applications and robot navigations. It is also desirable that the camera's initial position and posture are estimated automatically.

References

1. T. Höllerer, S. Feiner and J. Pavlik: "Situated documentaries: Embedding multimedia presentations in the real world," Proc. Int. Symp. on Wearable Computers, pp. 79–86, 1999.
2. T. H. S. Feiner, B. MacIntyre and A. Webster: "A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment," Proc. Int. Symp. on Wearable Computers, pp. 74–81, 1997.
3. R. Tenmoku, M. Kanbara and N. Yokoya: "A wearable augmented reality system using positioning infrastructures and a pedometer," Proc. IEEE Int. Symp. on Wearable Computers, pp. 110–117, 2003.
4. D. Hallaway, T. Höllerer and S. Feiner: "Coarse, inexpensive, infrared tracking for wearable computing," Proc. IEEE Int. Symp. on Wearable Computers, pp. 69–78, 2003.
5. G. Welch, G. Bishop, L. Vicci, S. Brumback, K. Keller and D. Colucci: "High-performance wide-area optical tracking -the hiball tracking system," Presence: Teleoperators and Virtual Environments, Vol. 10, No. 1, pp. 1–21, 2001.
6. H. Kato and H. Billinghurst: "Marker tracking and hmd calibration for a video-based augmented reality conferencing system," Proc. IEEE/ACM Int. Workshop on Augmented Reality, pp. 85–94, 1999.
7. U. Neumann and S. You: "Natural feature tracking for augmented-reality," IEEE Transactions on Multimedia, Vol. 1, No. 1, pp. 53–64, 1999.
8. A. J. Davison, Y. G. Cid and N. Kita: "Real-time 3d slam with wide-angle vision," Proc. IFAC Symp. on Intelligent Autonomous Vehicles, 2004.
9. L. Naimark and E. Foxlin: "Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker," Proc. IEEE/ACM Int. Symp. on Mixed and Augmented Reality, pp. 27–36, 2002.
10. C. Tomasi and T. Kanade: "Shape and motion from image streams under orthography: A factorization method," Int. J. of Computer Vision, Vol. 9, No. 2, pp. 137–154, 1992.
11. P. Beardsley, A. Zisserman and D. Murray: "Sequential updating of projective and affine structure from motion," Int. J. of Computer Vision, Vol. 23, No. 3, pp. 235–259, 1997.
12. V. Lepetit, L. Vacchetti, D. Thalmann and P. Fua: "Fully automated and stable registration for augmented reality applications," Proc. Int. Symp. on Mixed and Augmented Reality, pp. 93–102, 2003.
13. I. Gordon and D. G. Lowe: "Scene modelling, recognition and tracking with invariant image features," Proc. Int. Symp. on Mixed and Augmented Reality , pp. 110–119, 2004.

14. T. Sato, S. Ikeda and N. Yokoya: "Extrinsic camera parameter recovery from multiple image sequences captured by an omni-directional multi-camera system," Proc. European Conf. on Computer Vision, Vol. 2, pp. 326–340, 2004.
15. C. Harris and M. Stephens: "A combined corner and edge detector," Proc. Alvey Vision Conf., pp. 147–151, 1988.
16. M. A. Fischler and R. C. Bolles: "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," Comm. of the ACM, Vol. 24, pp. 381–395, 1981.
17. S. Ikeda, T. Sato and N. Yokoya: "A calibration method for an omnidirectional multi-camera system," Proc. SPIE Electronic Imaging, Vol. 5006, pp. 499–507, 2003.