Vehicle Ego-localization by Matching In-vehicle Camera Images to an Aerial Image

Masafumi NODA[1,*], Tomokazu TAKAHASHI[1,2], Daisuke DEGUCHI[1], Ichiro IDE[1], Hiroshi MURASE[1], Yoshiko KOJIMA[3] and Takashi NAITO[3]
[1] *Nagoya University, Furo-cho, Chikusa-ku, Nagoya, Aichi, 464–8601, Japan*
[2] *Gifu Shotoku Gakuen University, Nakauzura 1–38, Gifu, 500–8288, Japan*
[3] *Toyota Central Research & Development Laboratories, Inc., 41–1 Aza Yokomichi, Oaza Nagakute, Nagakute, Aichi, 480–1192, Japan*
[*] *mnoda@murase.m.is.nagoya-u.ac.jp*

Abstract. Obtaining an accurate vehicle position is important for intelligent vehicles in supporting driver safety and comfort. This paper proposes an accurate ego-localization method by matching in-vehicle camera images to an aerial image. There are two major problems in performing an accurate matching: (1) image difference between the aerial image and the in-vehicle camera image due to view-point and illumination conditions, and (2) occlusions in the in-vehicle camera image. To solve the first problem, we use the SURF image descriptor, which achieves robust feature-point matching for the various image differences. Additionally, we extract appropriate feature-points from each road-marking region on the road plane in both images. For the second problem, we utilize sequential multiple in-vehicle camera frames in the matching. The experimental results demonstrate that the proposed method improves both ego-localization accuracy and stability.

1 Introduction

The vehicle ego-localization task is one of the most important technologies for Intelligent Transport Systems (ITS). Obtaining an accurate vehicle position is the first-step to supporting driver safety and comfort. In particular, ego-localization near intersections is important for avoiding traffic accidents. Recently, in-vehicle cameras for the ego-localization have been put to practical use. Meanwhile, aerial images have become readily available, for example from Google Maps [1]. In light of the above, we propose a method for accurate ego-localization by matching the shared region taken in in-vehicle camera images to an aerial image.

A global positioning system (GPS) is generally used to estimate a global vehicle position. However, standard GPSs for a vehicle navigation system have an estimation error within about 30–100 meters in an urban area. Therefore, a relatively accurate position is estimated by matching information, such as a geo-location and an image taken from a vehicle, to a map. Among them, map-matching [2] is one of the most prevalent methods. This method estimates

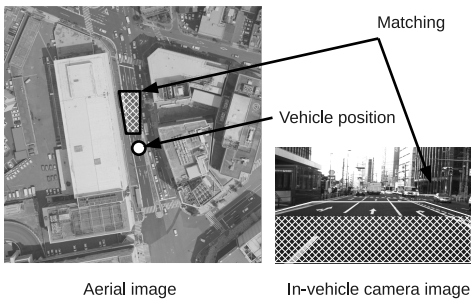


Fig. 1. Vehicle ego-localization by matching in-vehicle camera image to an aerial image: Shaded regions in both images correspond.

a vehicle position by matching a vehicle's driving trajectory calculated from rough estimations using GPS to a topological road map. Recently, in-vehicle cameras have been widely used; therefore, vehicle ego-localization using cameras has been proposed [3–5]. This camera-based vehicle ego-localization matches in-vehicle camera images to a map, which is also constructed from in-vehicle camera images. In many cases, the map is constructed by averaging in-vehicle camera images with less-accurate geo-locations. Therefore, it is difficult to construct a globally consistent map.

In contrast, aerial images that covers a wide region and with a highly accurate geo-location have also become easily available, and we can collect them at low-cost. There are some methods that ego-localize an aircraft by matching aerial images [6, 7]. However, the proposed method estimates a vehicle position. The proposed method matching the shared road-region of in-vehicle camera images and an aerial image is shown in Figure 1. Pink et al. [8] have also proposed an ego-localization method based on this idea. They estimate a vehicle position by matching feature-points extracted from an aerial image and an in-vehicle camera image. An Iterative Closest Point (ICP) method is used for this matching. As feature-points, the centroids of road markings, which are traffic symbols printed on roads, are used. This method, however, has a weakness in that a matching error occurs in the case where the images differ due to illumination conditions and/or occlusion. This decreases ego-localization accuracy.

There are two main problems to be solved to achieve accurate ego-localization using in-vehicle camera images and an aerial image. We describe these problems and our approaches to solve them.

1) **Image difference between the aerial image and the in-vehicle camera image**: The aerial image and the in-vehicle camera image have large difference due to viewpoints, illumination conditions and so on. This causes difficulty in feature-point matching. Therefore, we use the Speed Up Robust Feature (SURF) image descriptor [9]. The SURF image descriptor is robust for such differences of view and illumination. Additionally, since the road-plane region in the images has a simple texture, the feature-points extracted by a general method tend to be too few and inappropriate for the matching.

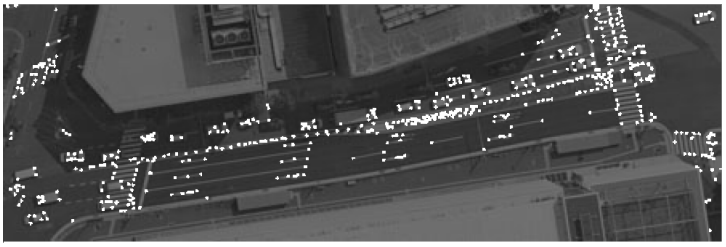


Fig. 2. Feature-point map: White dots represent feature-points.

Therefore, we extract feature-points appropriate for the matching from each road-marking region.

2) **Occlusion in the in-vehicle camera image**: In a real traffic environment, forward vehicles often exist. They occlude the road-markings in the in-vehicle camera image, and thus matching to an aerial image fails. However, even if the feature-points are occluded in some frames, they may be visible in other frames. Therefore, we integrate multiple in-vehicle camera frames to extract feature-points, including even those occluded in specific frames.

Based on the above approaches, we propose a method for vehicle ego-localization by matching in-vehicle camera images to an aerial image. The proposed method consists of two stages. The first stage constructs a map by extracting feature-points from an aerial image, which is performed offline. The second stage ego-localizes by matching in-vehicle camera images to the map.

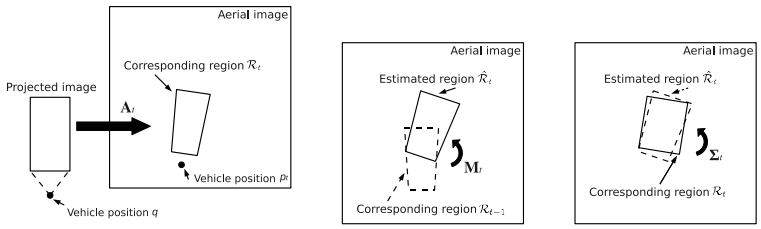
This paper is organized as follows: Section 2 proposes a method of map construction from an aerial image, and Section 3 proposes a method of ego-localization by matching in-vehicle camera images to the map, in real time. Experimental results are presented in Section 4, and discussed in Section 5. Section 6 summarizes this paper.

2 Construction of Feature-points Map for Ego-localization

A feature-points map is constructed from an aerial image for the ego-localization. To adequately extract the applicable feature-points, we first extract road-marking regions and then extract the unique feature-points from each region. We then construct a map for the ego-localization using SURF descriptors [9], which are robust against the image difference between the aerial image and the in-vehicle camera image. Figure 2 shows a feature-point map constructed from the aerial image. In this paper, the road region of the intended sequences is manually extracted in advance to evaluate the proposed method. We will automatically extract the region by a segmentation method in future work.

The map construction process is divided into the following steps:

1. Emphasize road markings by binarizing an aerial image, then split it into multiple regions by a labeling method.
2. Eliminate the regions considering appropriate road-marking size.
3. Extract feature-points $\mathbf{x}_n (n = 1, \ldots, N)$ from the road-marking regions in the binary image by Harris corner detector.



(a) (b) (c)

Fig. 3. Overview of the proposed method: (a) Correspondence of a projected image and the region in aerial image. (b) Estimation of the current corresponding region. (c) Estimation of an accurate corresponding region.

4. Calculate the SURF descriptor $\mathbf{f}_n$ around $\mathbf{x}_n$ from the aerial image.

The feature-point map is represented as the pairs of the position and the SURF descriptor $\{(\mathbf{x}_1, \mathbf{f}_1), \ldots, (\mathbf{x}_N, \mathbf{f}_N)\}$. In this paper, we treat objects on the road such as vehicles and trees as well as road markings, though the detection of these objects is required in a fully developed system.

3 Ego-localization by Matching the In-vehicle Camera Images to the Map

3.1 Overview

Vehicle ego-localization is achieved by sequentially matching in-vehicle camera images to a map constructed from an aerial image. The proposed method ego-localizes a vehicle at time step t (frame) by the following steps:

1. Transformation of an in-vehicle camera image to a projected image
2. Sequential matching between projected images
3. Matching of the projected image to the map using multiple frames
4. Estimation of the vehicle position

The proposed method first transforms the in-vehicle camera image to a projected image to simplify the matching process. Then, the proposed method finds a region $\mathcal{R}_t$ in the map that corresponds to the in-vehicle camera image as shown in Figure 3(a). The homography matrix $\mathbf{A}_t$ in this figure transforms the projected image on $\mathcal{R}_t$. Then, we estimate the vehicle position $\mathbf{p}_t$ as

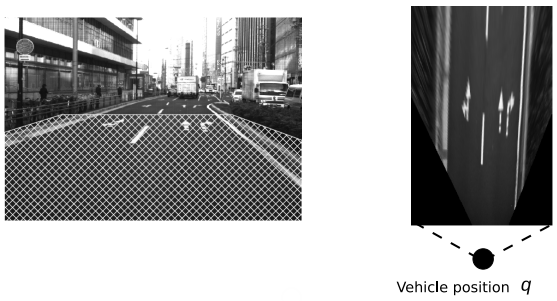
$$\mathbf{p}_t = \mathbf{A}_t \mathbf{q}, \tag{1}$$

where $\mathbf{q}$ is the vehicle position in the projected image, as shown in Figure 4(b) and Figure 3(a), obtained from the in-vehicle camera parameters.

The proposed method updates $\mathbf{A}_t$ by the two-step estimation shown in Figure 3(b) and Figure 3(c). $\mathbf{A}_t$ is then updated as

$$\mathbf{A}_t = \boldsymbol{\Sigma}_t \mathbf{A}_{t-1} \mathbf{M}_t. \tag{2}$$

$\mathbf{M}_t$ and $\boldsymbol{\Sigma}$ are the homography matrices. $\mathbf{M}_t$ transforms the projected image to the estimated corresponding region $\hat{\mathcal{R}}_t$ from the previous frame as shown in



(a) In-vehicle camera image. (b) Projected image.

Fig. 4. Transformation of an in-vehicle camera image to a projected image: the shaded region in (a) is transformed to the projected image (b).

Figure 3(b). Then, $\mathbf{M}_t$ is estimated by the sequential matching between projected images. The estimated region, however, contains some error due to the matching error $\boldsymbol{\Sigma}_t$, which transforms the estimated region to an accurate corresponding region $\mathcal{R}_t$ as shown in Figure 3(c). Therefore, $\boldsymbol{\Sigma}_t$ is estimated by the matching of the projected image to the map. In this matching, multiple in-vehicle camera frames are used to improve the matching accuracy. This aims to increase the number of feature-points and to perform accurate matching in a situation where part of the road markings are occluded in the in-vehicle camera images. We detail the ego-localization process below.

3.2 Transformation of an In-vehicle Camera Image to a Projected Image

An in-vehicle camera image is transformed to a projected image as shown in Figure 4. To transform the projected image, a 3×3 homography matrix is used. The matrix is calculated in advance from the in-vehicle camera parameters: installed position, depression angle and focal length. The vehicle position $\mathbf{q}$ in a projected image is also obtained using the matrix.

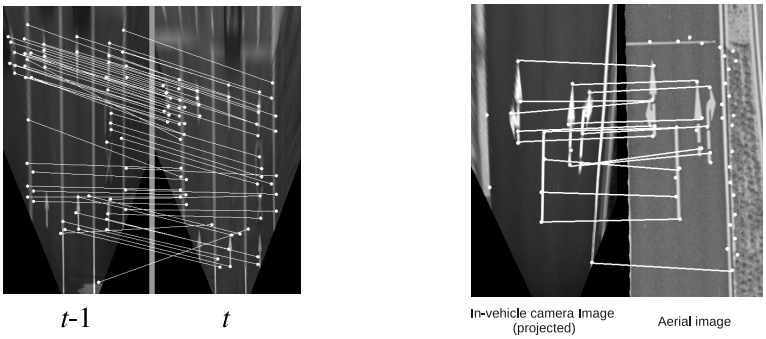
3.3 Sequential Matching between Projected Images

To estimate $\hat{\mathcal{R}}_t$, the proposed method performs the matching between sequential projected images. The projected image at t is represented as I_t. $\mathbf{M}_t$, shown in Figure 3(b), is obtained by matching between the feature-points in I_{t-1} and I_t.

The feature-points are extracted by Harris corner detector, then matched by Lucas-Kanade's method. Figure 5(a) shows the initial correspondence between the feature-points. $\mathbf{M}_t$ is calculated by minimizing the LMedS criterion by selecting the correspondences. $\hat{\mathcal{R}}_t$ is calculated from $\mathbf{M}_t$ and $\mathbf{A}_{t-1}$.

3.4 Matching of the Projected Image to the Feature-points Map using Multiple Frames

$\hat{\mathcal{R}}_t$ contains some error, which is represented as a homography matrix $\boldsymbol{\Sigma}_t$ shown in Figure 3(c). We calculate $\boldsymbol{\Sigma}_t$ by matching the projected image to the map



(a) Sequential matching between projected images.

(b) Matching the projected image to the feature-points map.

Fig. 5. Two step matching (Corresponding feature-point pairs in the projected images: The dots represent the feature-point in each image and the lines show their correspondence).

to obtain the accurate corresponding region $\mathcal{R}_t$. In this matching, in order to improve the accuracy and stability in a situation where occlusions occur in the in-vehicle camera image, multiple in-vehicle camera frames are used. We first explain a matching method the only uses a single frame, and then how to extend it to that uses multiple frames.

Matching using a Single Frame We extract the feature-points from the projected images in the same manner as described in Section 2. The position of a feature-point extracted from I_t is represented as $\mathbf{y}_{t,l_t}(l_t = \{1, \ldots, L_t\})$, where L_t is the number of feature-points. The SURF descriptor of $\mathbf{y}_{t,l_t}$ is represented as $\mathbf{g}_{t,l_t}$. Thus, the feature-points could be represented as $\{(\mathbf{y}_{t,1}, \mathbf{g}_{t,1}), \ldots, (\mathbf{y}_{t,L_t}, \mathbf{g}_{t,L_t})\}$.

For the matching, each feature-point position $\mathbf{y}_{t,l_t}$ is transformed to $\mathbf{y}'_{t,l_t}$ in the map as

$$\mathbf{y}'_{t,l_t} = \mathbf{A}_{t-1}\mathbf{M}_t\mathbf{y}_{t,l_t}. \tag{3}$$

Feature-point pairs are chosen so that they meet the following conditions:

$$\begin{cases} ||\mathbf{y}'_{t,l_t} - \mathbf{x}_n|| < r \\ \min\limits_{l_t} ||\mathbf{g}_{t,l_t} - \mathbf{f}_n|| \end{cases}, \tag{4}$$

where r is the detection radius. Figure 5(b) shows the feature-point pairs. Then, $\boldsymbol{\Sigma}_t$ is obtained by minimizing the LMedS criterion by selecting the correspondences.

Matching using Multiple Frames To achieve accurate matching in a situation where occlusions occur in some in-vehicle camera images, we integrate the feature-points in the multiple in-vehicle camera frames. The feature-points at t' are represented as $\mathcal{Y}_{t'} = \{\mathbf{y}_{t',1}, \ldots, \mathbf{y}_{t',L_{t'}}\}$. They are transformed to $\mathcal{Y}'_{t'} =$

Table 1. Dataset

		Aerial image	In-vehicle camera image	
Set No.	Length (m)	Occlusion	Occlusion	Time
1	85	small	small	day
2	100	small	small	night
3	100	small	large	day
4	75	large	large	day

$\{\mathbf{y}'_{t',1}, \ldots, \mathbf{y}'_{t',L_{t'}}\}$ in the map coordinate. $\mathbf{y}'_{t',1}$ is transformed as

$$\mathbf{y}'_{t',l_{t'}} = \begin{cases} \mathbf{A}_{t'-1}\mathbf{M}_{t'}\mathbf{y}_{t',l_{t'}} & t' \text{ is current frame} \\ \mathbf{A}_{t'}\mathbf{y}_{t',l_{t'}} & \text{otherwise} \end{cases}. \tag{5}$$

Then, the feature-points in the F multiple frames including the current frame are used for the matching. Then, we obtain $\boldsymbol{\Sigma}_t$ in the same manner as in the case of a single frame.

3.5 Estimation of the Vehicle Position

Finally, $\mathbf{A}_t$ is calculated by Equation 2, and the vehicle position $\mathbf{p}_t$ is estimated by Equation 1. As for the matrix $\mathbf{A}_0$ at the initial frame, it is obtained by a global matching method in the map without the estimation of $\hat{\mathcal{R}}_0$

4 Experiment

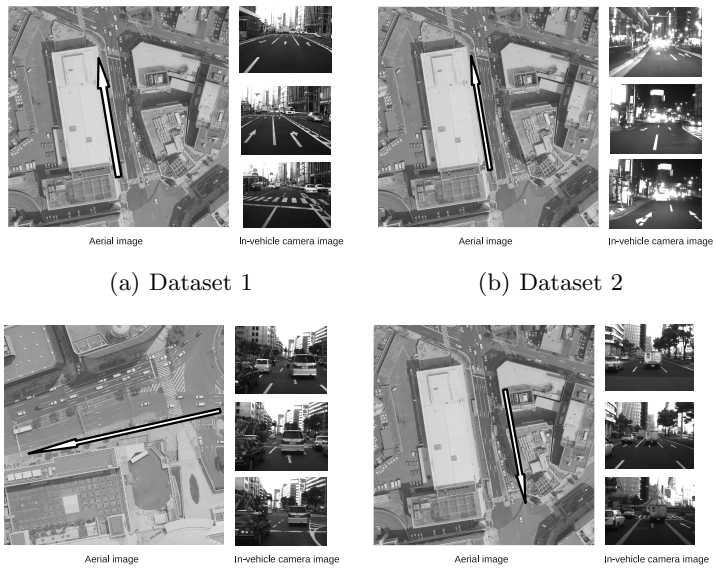
4.1 Setup

We mounted a camera, a standard GPS and a high accurate positioning system (Applanix, POSLV) [10] on a vehicle. The standard GPS contains an error of about 5–30 meters, which was used for the initial frame matching. The high-accuracy positioning system was used to obtain the reference values of vehicle positions. We used four sets of an aerial image and an in-vehicle camera image sequence with different capturing conditions. Table 1 shows the specification of the datasets and Figure 6 shows examples. The resolution of the aerial image was 0.15 meters per pixel. The resolution of the in-vehicle camera image was 640×480 pixels, and its frame-rate was 10 fps. Occlusions in the aerial image occurred due to vehicles, trees and so on. Occlusions in the road regions in an aerial image occurred due to vehicles, trees and so on. We defined a road segment in an aerial image which was occluded less than 10% as a small occlusion, and that occluded more than 50% as a large occlusion by visual judgment. Occlusions in the in-vehicle camera images were due to forward vehicles.

4.2 Evaluation

We evaluated the ego-localization accuracy by the Estimation Error and the Possible Ratio defined by the following equations:

$$\text{Estimation error} = \frac{\text{The sum of estimation errors in available frames}}{\text{The number of available frames}}, \tag{6}$$



(a) Dataset 1 (b) Dataset 2

(c) Dataset 3 (d) Dataset 4

Fig. 6. Datasets: Four sets of an aerial image and an in-vehicle camera image sequences.

$$\text{Possible ratio} = \frac{\text{The number of available frames}}{\text{The number of all frames}}. \tag{7}$$

The Estimation Error is the average error between the estimated vehicle position and the reference value. On the other hand, the Possible Ratio represents the stability of the estimation. So, we use available frames in which the estimation was achieved successfully to calculate the Estimation Error. The available frames were checked by the size and twisting of the corresponding region, which was transformed from the projected image to the aerial image. When the Possible Ratio was less than 0.50, we did not calculate the Estimation Error.

In this experiment, we compared the ego-localization accuracy between the proposed method and a method based on [8]. The comparative method used only the center position of road markings as the feature-point, then performed the matching of these feature-points to the map using the ICP method. In this matching, the comparative method used only a single in-vehicle camera frame. On the other hand, the proposed method used five frames selected from frames for the previous five seconds with the same interval.

4.3 Initial Estimation

For the initial estimation, we performed matching between a projected image and a circular region in an aerial image with the radius of 30 meters around the location measured by a standard GPS. In cases where the estimation failed in the frame, we also performed this initial estimation in the next frame.

Table 2. Experimental result.

	Proposed		Compared	
Set No.	Error (m)	Possible Ratio	Error (m)	Possible Ratio
1	0.60	1.00	0.72	0.83
2	0.70	1.00	0.75	0.90
3	0.98	0.73	N/A	0.30
4	N/A	0.12	N/A	0.04

4.4 Experimental Result

Table 2 shows the ego-localization accuracy. Each row shows the Estimation Error and the Possible Ratio of each dataset. We confirmed from this result that the proposed method improved the accuracy for all datasets compared with the comparative method. In the case of Dataset 1 with small occlusion in both the in-vehicle camera image sequence and the aerial image, the Estimation Error was 0.60 meters by the proposed method. Furthermore, the Possible Ratio 1.00 was achieved by the proposed method, compared to 0.83 by the comparative method. Thus, we also confirmed the high stability of the proposed method. In the case of Dataset 2 with the in-vehicle camera image sequence taken at night, the Estimation Error and the Possible Ratio also improved.

In the case of Dataset 3 with a large occlusion in the in-vehicle camera image sequence, an Estimation Error of 0.98 and Possible Ratio of 0.73 were achieved by the proposed method. In contrast, a Possible Ratio of only 0.30 was achieved by the comparative method, and the Estimation Error was not available because the possible rate was less than 0.50. Finally, in the case of Dataset 4, there was a large occlusion in the aerial image, and ego-localization by both methods was not available in most frames due to mismatching of the feature-points.

The estimation of the proposed method consumed about 0.6 (sec) per frame when we used a computer whose CPU was Intel(R) Core(TM) i7 860 2.80GHz.

5 Discussion

1) **Image Difference between the Aerial Image and the In-vehicle Camera Image**: For matching the in-vehicle camera image to the aerial image, we extracted unique feature-points from road markings, and used the SURF descriptor. From the results of Datasets 1 and 2, the proposed method improved the Estimation Error and the Possible Ratio. The results demonstrated that the proposed method could make the matching robust for the image difference between the images.
2) **Occlusion in the In-vehicle Camera Image**: The feature-points extracted from the in-vehicle camera image were occluded in some frames. However, they were not occluded in other frames. From the result of Dataset 3, we confirmed that the matching using the multiple frames in the proposed method worked well in such situations. In this experiment, we fixed the number of frames used for the matching. We consider that adapting the number to the changes of occlusions could further improve the performance.

3) **Limitation of the Proposed Method**: From the result of Dataset 4, the proposed method could not estimate accurately the vehicle position when a large occlusion existed in the aerial image. To solve this problem, we need to construct a map without occlusions. In future work, we will detect the occluded regions and interpolate them by using in-vehicle camera images.

6 Conclusion

We proposed a vehicle ego-localization method using in-vehicle camera images and an aerial image. There are two major problems in performing accurate matching: the image difference between the aerial image and the in-vehicle camera image due to view-points and illumination conditions; and occlusions in the in-vehicle camera image. To solve these problems, we improved the feature-point detector and the image descriptor. Additionally, we extracted appropriate feature-points from each road marking region on the road plane in both images, and utilized sequential multiple in-vehicle camera frames in the matching. The experimental results demonstrated that the proposed method improves both the ego-localization accuracy and the stability. Future work includes construction of a feature-points map without occlusions by using in-vehicle camera images.

Acknowledgement

Parts of this research were supported by JST CREST and MEXT, Grant-in-Aid for Scientific Research. This work was developed based on the MIST library (`http://mist.murase.m.is.nagoya-u.ac.jp/`).

References

1. Google Inc.: Google Maps (http://maps.google.com/) (2005)
2. Brakatsoulas, S., Pfoser, D., Salas, R., Wenk, C.: On map-maching vehicle tracking data. In: Proc. 32nd Conf. on Very Large Data Bases. (2005) 853–864
3. Kawasaki, H., Miyamoto, A., Ohsawa, Y., Ono, S., Ikeuchi, K.: Multiple video camera calibration using EPI for city modeling. In: Proc. 6th Asian Conf. on Computer Vision. Volume Vol. 1. (2004) 569–574
4. Ono, S., Mikami, T., Kawasaki, H., Ikeuchi, K.: Space-time analysis of spherical projection image. In: Proc. 18th Int. Conf. on Pattern Recognition. (2006) 975–979
5. Uchiyama, H., Deguchi, D., Takahashi, T., Ide, I., Murase, H.: Ego-localization using streetscape image sequences from in-vehicle cameras. In: Proc. Intelligent Vehicle Symp. 2009. (2009) 185–190
6. Lin, Y., Yu, Q., Medioni, G.: Map-enhanced UAV image sequence registraton. In: Proc. 8th Workshop on Applications of Computer Vision. (2007) 15–20
7. Caballero, F., Merino, L., Ferruz, J., Ollero, A.: Homography based Kalman filter for mosaic building. Applications to UAV position estimation. In: Proc. Int. Conf. on Robotics and Automation. (2007) 2004–2009
8. Pink, O., Moosmann, F., Bachmann, A.: Visual features for vehicle localization and ego-motion estimation. In: Proc. Intelligent Vehicle Symp. 2009. (2009) 254–260
9. Bay, H., Ess, A., Tuytelaars, T., Gool, L.V.: SURF: Speeded up robust features. Computer Vision and Image Understanding **110** (2008) 346–359
10. Applanix Corp.: POS LV (http://www.applanix.com/products/land/pos-lv.html) (2009)