Proceedings of the 2006 IEEE International Conference on Robotics and Automation
Orlando, Florida - May 2006

Ego-Motion Estimation by Matching Dewarped Road Regions Using Stereo Images

Akihito Seki and Masatoshi Okutomi
Graduate School of Science and Engineering
Tokyo Institute of Technology
Email: a-seki@ok.ctrl.titech.ac.jp, mxo@ctrl.titech.ac.jp

Abstract— **This paper proposes a method for vehicle ego-motion estimation using vehicle-mounted stereo cameras. Estimating ego-motion using cameras requires extraction of static regions from the images. We first use stereo images to estimate regions which correspond to the road plane and can be considered as static areas. Subsequently, we propose a virtual projection plane (VPP) image that is equivalent to the top view of the road scene. A vehicle's ego-motion is obtained by matching the sequential VPP images of the road patterns in the extracted regions. We use a vehicle-motion model and consider a matching method to easily and accurately determine the ego-motion. Finally, we present experimental results obtained using our method.**

I. Introduction

Estimating ego-motion using vehicle-mounted cameras is useful for visual navigation, assistance for safe driving, and converting the relative velocities of objects, which contain the vehicle's velocity, to absolute velocity.

Commonly, estimation of the vehicle ego-motion requires velocity sensor, rudder angle sensor, yaw rate sensor, and so on. However, if the wheels slip, the sensors obtain incorrect data.

In contrast, the matter is resolved if a road surface is observed directly, as is done by using a camera. Ego-motion estimation method using a camera is divisible into two categories. One uses a single camera. For example, reading line patterns with a standardized length on a highway yields the velocity of a vehicle [1]. Reading special patterns using a slit camera indicates the velocity and relative position [2]. Nevertheless, these methods can only be used under specific environments with standardized road patterns.

First, we describe methods without specific road patterns using single camera. It is possible to estimate ego-motion from optical flows on the road plane from images [3], [4]. However, it is difficult to distinguish optical flows which belong only to a static area, i.e. a road plane. Another method is estimation of a homography matrix for the road plane of sequential images [5]. These methods are incapable of estimating the distance to the road plane, so we are unable to obtain an absolute translation.

Next, Another method uses multiple cameras. This method corresponds stationary target points among images to obtain three-dimensional (3D) positions. Then the method checks changes of 3D positions at different times. The ego-motion derives from the changes [6]. Nevertheless, this method requires some successful conditions. Those conditions are that the correspondence of the points must be correct between the two times; the tracking must be correct and the target points must be stationary. Because of these conditions, we believe that generation of a method to obtain ego-motion under a complicated urban environment is extremely difficult.

Our proposed method is different from these methods. We explicitly extract static areas and dynamically estimate vehicle's pose. Then we can obtain vehicle ego-motion including an absolute translation by matching dewarped road regions. Our proposed method does not use stereo correspondence and tracking of feature points, but it uses road patterns in static areas and adapts to vehicle's pose change. These features enable our method to be stable even in an urban environment.

II. Proposed Method Overview

An overview of our proposed method is shown in Fig. 1. First, we employ the Road-Region Extraction method [7] which searches for a planar region in 3-D space using sequential stereo images. In this method, by dynamically estimating the homography matrix of the planar region, we can deal with vehicle vibration.

We can also determine the planar pose parameters, which are the normal vector of the planar region, and the distance between the optical center of the camera to the plane by decomposition of the homography matrix. This planar-pose parameter is estimated dynamically and is responsive to the vehicle's pose change. Next, we propose the use of a virtual projection plane (VPP) image, which is equivalent to the top view of the road scene. We produce sequential VPP images and VPP images of the road region. Ego-motion can be determined using these images.

We obtain the appropriate ego-motion by matching the sequential VPP images in common road regions of these frames. The matching method is the following. First, we estimate ego-motion based on a vehicle-motion model for increasing stability and reducing computational cost. Second, we apply an update-motion-parameter stage whereby the motion parameters are iteratively updated by using the parameters obtained at the previous stage as the initial estimates.

Here, we first explain the road-region extraction method proposed by Okutomi [7]. Figure 2 shows the process of the road-region extraction method. The process begins with inputs of stereo images. Next, the images are convoluted using a LoG filter. Then, we dynamically estimate the road-plane

0-7803-9505-0/06/$20.00 ©2006 IEEE

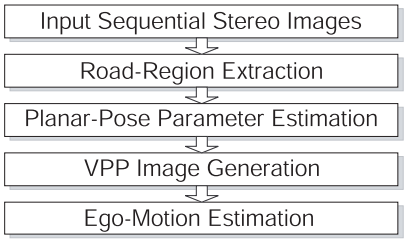

Fig. 1. Flow chart of our proposed method.

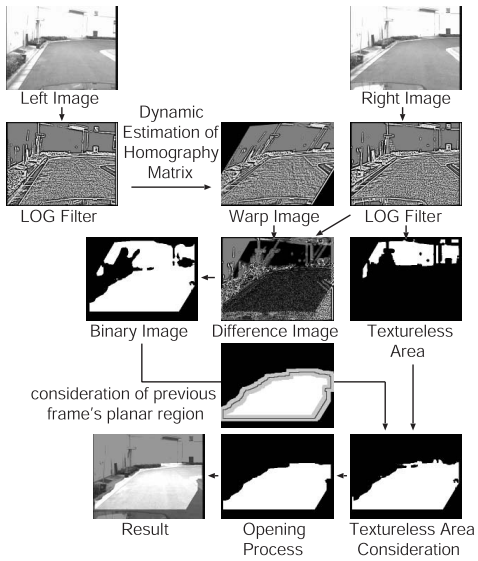

Fig. 2. Flow chart of the road-region extraction method.

homography matrix (this reacts to vehicle vibration) and warp the left image using the matrix. We obtain a difference image between the warped left image and the right image. If a point projected to the left image is on the plane, the transformed left image with the homography matrix will match the right image. We set the threshold of the difference image and extract the planar region corresponding to the road plane. However, the pixels in textureless areas will also have a small value, even if those points are either over or under the plane. To solve this problem, we examine the pixels in this area as one block and evaluate all blocks whether they are on the plane or not. Finally, we make an opening process and delete the small area to thereby obtain the road region.

III. VPP Image Generation

In this section, we propose a method to generate an image that is parallel to the road plane. The geometry of this image is shown in Fig. 3. We name this image the Virtual Projection Plane (VPP) image.

The VPP image has some distinguishing properties:

- The VPP image coordinates correspond to real-space coordinates. For that reason, a point on the VPP image is uniquely equivalent to a real 3D coordinate.
- The VPP image vertical axis is equivalent to the projection of the optical axis of the right camera to the VPP image plane.

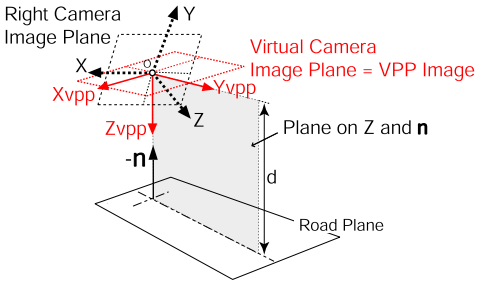

Fig. 3. VPP image geometry.

Planar pose parameters are needed to produce VPP images. First, we explain how to derive planar pose parameters and produce the VPP image.

A. Planar-Pose Parameter Estimation

Planar-pose parameter comprises two elements. One is the normal vector of the target plane (road plane) $\mathbf{n}$. Another is the distance d between the optical center of the right camera to the plane.

As known well, a plane (herein, the road plane) that is projected between two images is represented by a unique homography matrix $\mathbf{H}$:

$$\tilde{\mathbf{m}}_l \sim \mathbf{H}\tilde{\mathbf{m}}_r = k\mathbf{A}_l\left(\mathbf{R} + \frac{\mathbf{t}\mathbf{n}^T}{d}\right)\mathbf{A}_r^{-1}\tilde{\mathbf{m}}_r \quad (k \neq 0), \quad (1)$$

where $\tilde{\mathbf{m}}_*$ is a homogeneous coordinate of an image, $\mathbf{R}$ is the camera rotation, $\mathbf{A}_r, \mathbf{A}_l$ are intrinsic right and left camera parameters, $\mathbf{t}$ is the baseline, $\mathbf{n}$ is the normal vector of the plane, and d is the distance between the right camera and the plane.

As shown in Eq. (1), $\mathbf{n}$ and d are contained within the homography matrix $\mathbf{H}$. These parameters can be given by decomposing $\mathbf{H}$ [8]. If the intrinsic camera parameters are known, we can obtain $\mathbf{n}$ and $\mathbf{t}/d$. To obtain d, we use the baseline length $|\mathbf{t}|$ of a stereo camera.

B. VPP Image Generation

The VPP image feature was described before. These conditions are represented as the following:

- The optical axis of the virtual camera (Z_{vpp}) is equal to the normal vector ($\mathbf{n}$).
- The axis of Y_{vpp} is on the plane produced by the normal vector $\mathbf{n}$ and the optical axis of the right camera (Z).

The coordinate transformation of the right camera and virtual camera are represented by a rotation matrix $\mathbf{R}$ because they have the same optical center.

The rotation matrix $\mathbf{R}$ is represented as

$$\mathbf{R} = \begin{pmatrix} c\theta c\phi c\psi - s\phi s\psi & c\theta s\phi c\psi + c\phi s\psi & -s\theta c\psi \\ -c\theta c\phi s\psi - s\theta c\psi & -c\theta s\phi s\psi + c\phi c\psi & s\theta s\psi \\ s\theta c\psi & s\theta s\phi & c\theta \end{pmatrix}, \quad (2)$$

where $c\phi$ and $s\phi$ denote $\cos\phi$ and $\sin\phi$, respectively. In this matrix, unknown parameters are (θ, ϕ, ψ). Subsequently, using

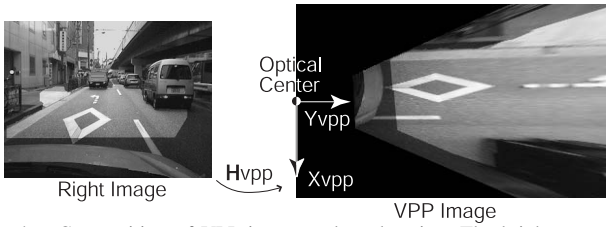

Fig. 4. Composition of VPP image and road region. The brighter areas in the images are the extracted road regions.

the rotation matrix, we consider the coordinate transformation as

$$\mathbf{X}_r = \mathbf{R}_{vpp\to r}\mathbf{X}_{vpp}, \tag{3}$$

where $\mathbf{X}_r$ is right camera coordinate and $\mathbf{X}_{vpp}$ is the virtual one. By the first condition, the axis of Z_{vpp} is $\mathbf{X}_{vpp} = (0,0,1)^T$. Therefore, we get

$$\mathbf{n} = (n_x, n_y, n_z)^T = \mathbf{R}_{vpp\to r}(0,0,1)^T. \tag{4}$$

The equation gives

$$\begin{cases} \theta &= \arccos(n_z) \\ \psi &= \arctan\left(\frac{-n_y}{n_x}\right) \end{cases} \tag{5}$$

Then, the axis of Y_{vpp} is on the plane made by that of Z and $Z_{vpp} = \mathbf{n}$ for the second condition.

$$\mathbf{R}_{vpp\to r}(0,1,0)^T = a(0,0,1)^T + (1-a)\mathbf{n}, \tag{6}$$

where a is an arbitrary constant number. This equation gives $\phi = \pm\pi/2$. The axes of Y_{vpp} and Z are the same direction, so the sign of ϕ is determined by the condition that these dot products are positive.

$$(0,0,1)\cdot\mathbf{R}_{vpp\to r}(0,1,0)^T > 0 \tag{7}$$

We have determined $\mathbf{R}_{vpp\to r}$ at this point. The transformation of the plane is described by the homography matrix. In this case, the homography between the right image plane and the virtual camera image plane is $\mathbf{H}_{vpp}$

$$\tilde{\mathbf{m}}_{vpp} \sim \mathbf{H}_{vpp}\tilde{\mathbf{m}}_r = \mathbf{A}_{vpp}\mathbf{R}^{-1}_{vpp\to r}\mathbf{A}_r^{-1}\tilde{\mathbf{m}}_r, \tag{8}$$

where $\tilde{\mathbf{m}}_{vpp}$ is the homogeneous coordinate of the VPP image, $\tilde{\mathbf{m}}_r$ is that of the right image and $\mathbf{A}_{vpp}$ represents the virtual camera's intrinsic parameter.

Figure 4 shows the VPP image and the composition of the planar region corresponding to the image. The brighter areas in these images are the extracted road regions.

Generation results of VPP images are shown in Sec. V-A.

IV. Ego-Motion Estimation

This section presents the vehicle ego-motion estimation method.

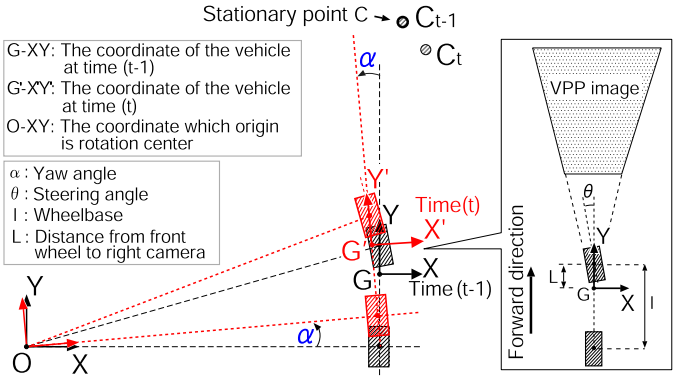

Fig. 5. Vehicle motion model.

A. Motion Estimation Using Vehicle-Motion Model

Assuming the target point (x, y) turns α and moves (t_x, t_y), we get the following.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \tag{9}$$

The shift vector$(\Delta x = x' - x, \Delta y = y' - y)$ is expressed below.

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} x(-1+\cos\alpha) - y\sin\alpha + t_x \\ x\sin\alpha + y(-1+\cos\alpha) + t_y \end{bmatrix} \tag{10}$$

The number of our unknown motion parameters is three, i.e. α, t_x, t_y. If we get one target point (x, y) and the shift vector $(\Delta x, \Delta y)$, the issue requires two equations. Thereby, if two target points and shift vectors are known, we can obtain the motion parameters. But it is difficult to obtain the shift vector of each point because of using small window in general for estimating, and is not easy to select accurately estimated points from these ones.

Next, we consider that motion is based on a vehicle-motion model. In this condition, one target point and the shift vector can uniquely produce the motion parameters of a vehicle.

The vehicle-motion model assumes an Ackermann-Steering-Geometry model (four-wheeled vehicle transformed to a two-wheeled vehicle) [9]. We set that the center axis of the vehicle through the right camera. This situation is shown in Fig. 5. In this figure, l is the vehicle wheelbase, L is the distance from the front wheel to the right camera, θ is the steering angle, and α is the rotation angle, which is equivalent to the yaw angle.

In the coordinate of $G-XY$, which origin is point G on the vehicle, the stationary point C locates $\mathbf{C}_{t-1,G} = (x, y)$. The rotation center O in the coordinate of $G-XY$ is

$$\mathbf{O}_G = \left(-\frac{l}{\tan\theta}\ ,\ -l+L\right). \tag{11}$$

Next, the point C transformed to the coordinate of $O-XY$ is

$$\mathbf{C}_{t-1,O} = \left(\frac{l}{\tan\theta} + x\ ,\ l-L+y\right). \tag{12}$$

The vehicle goes around point O by α at time t and the coordinate becomes $G'-X'Y'$. The stationary point C is the

same as the coordinate of $O-XY$, but $G-XY$ and $G'-X'Y'$ on the vehicle are not identical to these coordinates. The point C at $G'-X'Y'$ is transformed to C_t at $G-XY$, which is shown in Fig. 5. Point C_t is described as the following.

$$\begin{aligned}\mathbf{C}_{t,O} &= \begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} \frac{l}{\tan\theta}+x \\ l-L+y \end{bmatrix} \\ &= \begin{bmatrix} (\frac{l}{\tan\theta}+x)\cos\alpha+(l-L+y)\sin\alpha \\ -(\frac{l}{\tan\theta}+x)\sin\alpha+(l-L+y)\cos\alpha \end{bmatrix} \end{aligned} \quad (13)$$

The distance resulting from the movement is $\mathbf{C}_{t,O}-\mathbf{C}_{t-1,O}$. If the α is approximately zero, as ($\cos\alpha \fallingdotseq 1$, $\sin\alpha \fallingdotseq \alpha$), then the distance is shown as the following.

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \mathbf{C}_{t,O}-\mathbf{C}_{t-1,O} \fallingdotseq \begin{bmatrix} (l-L+y)\alpha \\ -(\frac{l}{\tan\theta}+x)\,\alpha \end{bmatrix} \quad (14)$$

Thereby, we obtain

$$\begin{cases} \alpha &= \frac{\Delta x}{l-L+y} \\ \theta &= \arctan\left(\frac{l}{-\frac{\Delta y}{\Delta x}(l-L+y)-x}\right) \end{cases} \quad (15)$$

The translation vector is described below.

$$\begin{cases} t_x &= \frac{l}{\tan\theta}(\cos\alpha-1)-(l-L)\sin\alpha \\ t_y &= \frac{l}{\tan\theta}\sin\alpha+(l-L)(\cos\alpha-1) \end{cases} \quad (16)$$

This issue illustrates that we are able to get ego-motion if the one target point and the shift vector are given.

B. Ego-Motion Estimation Using VPP Images

The relation between vehicle motion and the stationary coordinate is equal to the estimated camera ego-motion based on the stationary area in images. As already mentioned, it is reasonable that the road-region is a stationary coordinate. To address this issue, we use the VPP image and the road region and match these time series of VPP images within the road region.

For vehicle motion, swing speed has an upper bound. Speed $V_{\max}$[m/s] is represented empirically [10], If the camera frame rate is 30 [fps], then the yaw rate $\alpha_{\max}$[rad/frame] of the speed is the following.

$$\alpha_{\max} = \frac{V_{\max}}{30R_{ad}} \ , \ where \ \ V_{\max} = \sqrt{\mu R_{ad} g} \quad (17)$$

In that equation, μ is the coefficient of tire side-slip friction, g is the gravitational acceleration, and R_{ad} is the turning radius. If we consider $\mu = 0.7$ and $R_{ad} = 10$, $\alpha_{\max}$ is about 0.028 [rad/frame]. This issue shows that the yaw rate is very small. Therefore, we assume that the shift is uniform within the road-region on the VPP image. Consequently, we use 2D (translation only) matching and get the shift vector $\mathbf{m} = (\Delta x, \Delta y)$.

Figure 6 shows the matching flow of the VPP image. First, the VPP image of the road region at time t (R_t) and VPP image are shifted $\mathbf{m}$ ($R_{t,\mathbf{m}}$). Then we perform an AND operation between the region at time t to that of time $t-1$ (R_{t-1}) and obtain the common road-region area $R_s(\mathbf{m})$ two times.

$$R_s(\mathbf{m}) = R_{t,\mathbf{m}} \cap R_{t-1} \quad (18)$$

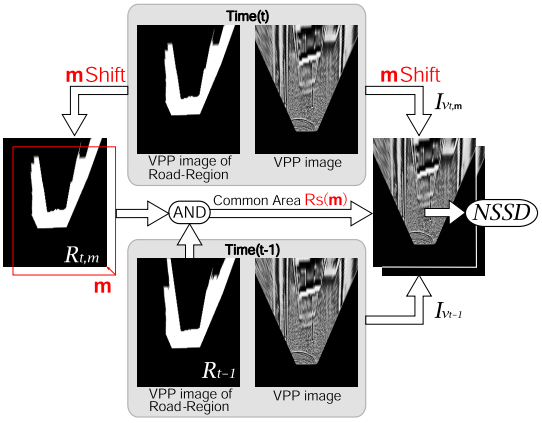

Fig. 6. Shift vector estimation flow.

Next, the original images, convolved with a LoG filter, are transformed to VPP images. The VPP image of the time t shifted $\mathbf{m}$ is $I_{V_{t,\mathbf{m}}}$. That of time $t-1$ is $I_{V_{t-1}}$.

To evaluate the similarity of the VPP images, we use NSSD (Normalized Sum of Squared Difference) within the common road region $R_s(\mathbf{m})$ in Eq. (19).

$$NSSD(\mathbf{m}) = \frac{1}{N}\sum_{\mathbf{x}\in R_s(\mathbf{m})}\{I_{V_{t,\mathbf{m}}}(\mathbf{x})-I_{V_{t-1}}(\mathbf{x})\}^2 \quad (19)$$

In that equation, N is the area of R_s. We want to get the minimum NSSD of $\mathbf{m}$, as

$$\hat{\mathbf{m}} = \arg\min_{\mathbf{m}\in\mathbf{M}_0}\{NSSD(\mathbf{m})\}, \quad (20)$$

where $\mathbf{M}_0$ is the search area of displacement per frame and $\hat{\mathbf{m}}$ is the shift vector.

Finally, vehicle ego-motion is obtained by converting the shift vector. The actual measure of the shift vector is calculated using internal camera parameters and the distance between the right camera and the road plane. Next, we obtain the ego-motion using Eq. (15), which is substituted for the real measure of the shift vector and the gravity center of $R_s(\hat{\mathbf{m}})$.

C. Updating Motion Parameters

This section presents the method to obtain more accurate motion by updating motion parameters.

Vehicle motion on the road plane is represented by only rotation and translation (Euclidean translation) on the VPP image. We derive a Euclidean translation matrix estimation formula by extending Shum's method [11]. This method requires good initial values; therefore, the motion parameters obtained in the previous section are set as the initial state of the matrix.

First, the Euclidean matrix is the following.

$$\mathbf{x}' \simeq \mathbf{M}\mathbf{x} = \begin{bmatrix} \cos\alpha & \sin\alpha & t_x \\ -\sin\alpha & \cos\alpha & t_y \\ 0 & 0 & 1 \end{bmatrix}\mathbf{x} \quad (21)$$

To recover the parameters, we iteratively update the transformation matrix using

$$\mathbf{M} \leftarrow \mathbf{M}(\mathbf{I}+\mathbf{D}). \quad (22)$$

In this equation, $\mathbf{D}$ is the matrix of incremental quantity. Resampling image I_{V_t} with the new transformation $\mathbf{x}' \simeq \mathbf{M}(\mathbf{I}+\mathbf{D})\mathbf{x}$ is equivalent to warping the resampled image $\tilde{I}_{V_t}$ by $\mathbf{x}'' = (\mathbf{I}+\mathbf{D})\mathbf{x}$. We wish to minimize the squared error metric of two images as the following.

$$E(\mathbf{d}) = \frac{1}{N}\sum_{i\in R_s}\left[\tilde{I}_{V_t}(\mathbf{x}_i'') - I_{V_{t-1}}(\mathbf{x}_i)\right]^2 \tag{23}$$

We use a Taylor series expansion to obtain

$$E(\mathbf{\Theta}) \approx \frac{1}{N}\sum_{i\in R_s}\left[\mathbf{g}_i{}^T\mathbf{J}_{\mathbf{\Theta}i}{}^T\mathbf{\Theta} + e_i\right]^2, \tag{24}$$

where

$$\mathbf{\Theta} = (\omega, dt_x, dt_y),\ \mathbf{g}_i^T = \nabla\tilde{I}_{V_t}(\mathbf{x}_i),\ e_i = \tilde{I}_{V_t}(\mathbf{x}_i) - I_{V_{t-1}}(\mathbf{x}_i)\ ,$$

$$\mathbf{J}_{\mathbf{\Theta}i} = \begin{bmatrix} -y & 1 & 0 \\ x & 0 & 1 \end{bmatrix}^T.$$

We use the Rodriguez formula and obtain an update matrix:

$$\mathbf{I}+\mathbf{D} = \begin{bmatrix} 0 & -\omega & dt_x \\ \omega & 0 & dt_y \\ 0 & 0 & 1 \end{bmatrix}. \tag{25}$$

At this moment, we want to minimize Eq. (24). $\mathbf{\Theta}$ at minimum E is equivalent to the derivative of E by $\mathbf{\Theta}$ equal to zero. By this condition, we get

$$\mathbf{A\Theta} = -\mathbf{b}, \tag{26}$$

where

$$\mathbf{A} = \sum_{i\in R_s}\mathbf{J}_{\mathbf{\Theta}i}\mathbf{g}_i\mathbf{g}_i{}^T\mathbf{J}_{\mathbf{\Theta}i}{}^T\ ,\ \ \mathbf{b} = \sum_{i\in R_s} e_i\mathbf{J}_{\mathbf{\Theta}i}\mathbf{g}_i\ .$$

These equations contain approximations. For that reason, we need an iteration estimation of $\mathbf{\Theta}$. Next, we update $\mathbf{\Theta}$, which are given in Eq. (26). The precise $\mathbf{M}$ is given in this iteration step. The converged matrix $\mathbf{M}$ gives vehicle motion parameters.

D. Kalman Filtering

For reducing errors in estimation, we apply Kalman filtering [12]. A state equation of the filter is shown as

$$\mathbf{x}_{t+1} = \mathbf{F}\mathbf{x}_t + noise(\mathbf{Q})\ ,\ y_t = \mathbf{H}\mathbf{x}_t + noise(R), \tag{27}$$

where

$$\mathbf{x} = [x\ \ \dot{x}]\ \ ,\ \ \mathbf{H} = [1\ \ 0]\ \ ,\ \ \mathbf{F} = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix}.$$

We use the measurement noise covariance R such that

$$R = \frac{K}{a}\ , \tag{28}$$

where, a represents the "roughness" of an image and K is a constant value. "Roughness" is the average slope of the error surface.

The similarity of matching VPP images is shown in Fig. 7. Road pattern matching often appears in the high-similarity direction (the direction of movement in this case). "Roughness" is the 2nd coefficient of quadratic function. We want to evaluate minimum roughness, so we consider four directions (in this figure, (1)–(4)) of roughness and determine the minimum roughness.

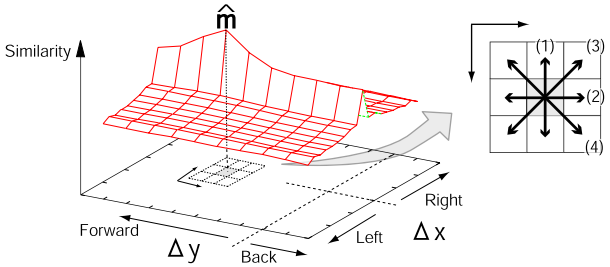

Fig. 7. Kalman filter and similarity.

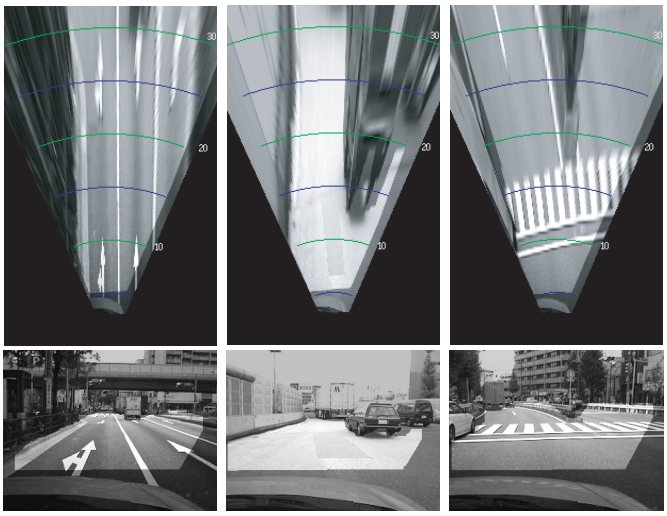
Fig. 8. Real scene results of road region extraction and VPP images (The upper image is the VPP image and the lower image is the right camera image. The brighter area in the images indicates the extracted road region.)

V. Experimental Results

This section first presents VPP image generation results. Then, we show the ego-motion estimation result using synthetic and real images.

A. VPP Image Generation Results

Figure 8 shows generated VPP images. The image size is 320×240 [pixel]. The VPP images are made up to 32 [m] distance. In these figures, the center of the circle arcs is the point at which the camera exists. The circle arcs are drawn at 5 [m] distance intervals. This generation method requires no specific road patterns such as parallel lines and maintains the direction to the plane because of the dynamical estimation of planar pose parameter.

B. Ego-Motion Estimation Results

1) Synthetic Images: Synthetic images using OpenGL are shown in Fig. 9. The images are assumed as captured at 30 fps and 320×240 [pixel].

The vehicle assumed herein has a 2.7 [m] wheelbase; a stereo camera is on the front wheel axis. The distance from the camera to the road surface is constant but unknown, the camera

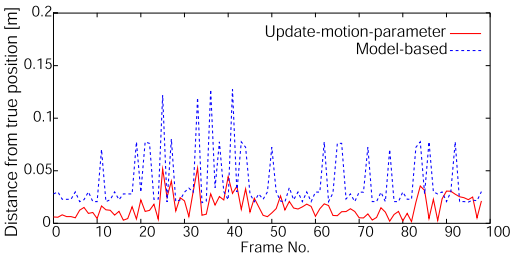

Fig. 10. Translation error.

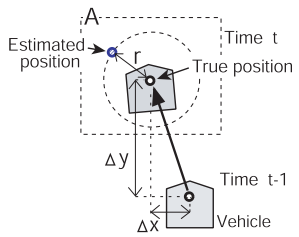

Fig. 11. Vehicle motion and error.

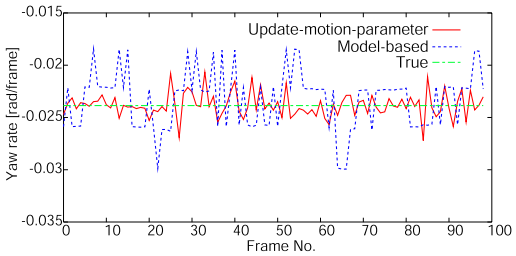

Fig. 12. Yaw rate.

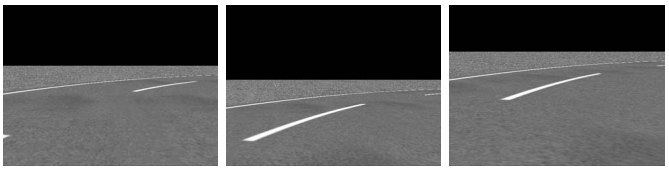

Fig. 9. Synthetic image sequence of right camera (left to right: Frame No. 0, 10, 30).

TABLE I

RMS ERROR OF MOTION ESTIMATION.

Planar pose parameter	Ego-motion estimation method	RMS error	
		Translation [m/frame]	Yaw rate [rad/frame]
Estimated result	model-based	4.672e-2	2.803e-3
	update-parameter	1.902e-2	1.141e-3
Fixed	model-based	6.185e-1	1.492e-2

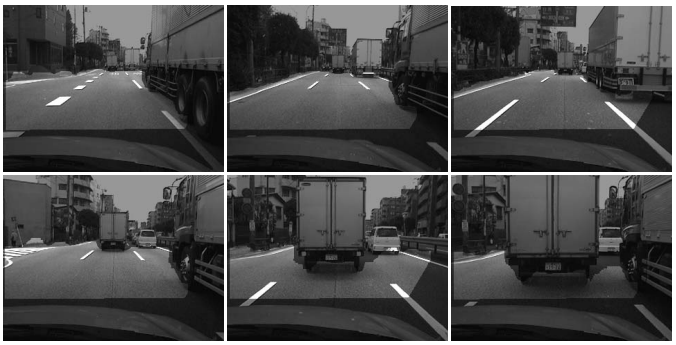

Fig. 13. Scene 1: Input image sequence (every 80 frames, upper left to lower right). The brighter areas in the images are the extracted road regions.

pose changes. The VPP image scale is set as 20 [pixel/m][1]. The vehicle motion has rotation by a constant rate and vibration on a pitch angle. This situation is shown in Fig. 9.

Figure 10 shows the difference between the true position and estimated position, which is equal to r in Fig. 11. The assumption of motion is $|\Delta x|$ = 5.76e-2 [m], $|\Delta y|$ = 5.88e-1 [m]. This figure represents two results. One method uses the vehicle-motion model (Sec. IV-A). We call this method the model-based method. Another one is to update parameters of the model-based method (Sec. IV-C). We call this the update-parameter method.

Figure 12 depicts the yaw rate result. Table I shows the RMS error of motion estimation. These results show that the model-based method itself yields accurate results and the update-parameter method gives even more accurate results. The column of "Planar pose parameter" in this table indicates whether the planar pose is dynamically estimated or fixed. If the parameter is fixed, the RMS error becomes larger by one digit.

2) Real Images: We show the experimental results in real environments.

Our experimental apparatus includes two cameras (320 ×240 [pixel] XC-003; Sony Corp.) mounted approximately parallel on a vehicle. The vehicle has a 2.7 [m] wheelbase and the distance between the front wheel to the camera is 0.85 [m]. The height of the camera from the road plane is about 1.2 [m] in a resting state. The depression angle is about 5 [degree], but it is altered by vibration.

Scene 1

Figure 13 shows the real images in an urban environment. Our vehicle travels in the midst of other moving vehicles, so this scene has some moving objects. Our vehicle slows down and finally stops.

Figure 14 represents the velocity along Y axis. This graph indicates slowing and the phenomenon corresponds to the fact. The solid line in this graph shows the results of parameter updating and the dashed line shows the results after Kalman filtering of them. Figure 15 depicts the yaw rate; it indicates that our vehicle is traveling nearly straight.

Scene 2

Figure 16 is a scene showing driving around a slow curve. Figure 17 represents the velocity along Y axis. This graph reads out an almost constant driving speed. Figure 18 shows the yaw rate, slowly turning right and back to center.

A trajectory of the vehicle is estimated using the motion data sequence. Figure 19 is the estimated trajectory overlaid on a map.

VI. CONCLUSIONS

In this paper, we propose a method of ego-motion estimation using vehicle-mounted stereo cameras. First, road regions are extracted and a homography matrix is estimated for the road plane. Through decomposition of the homography matrix, the normal vector of the road and the distance between road and

[1]Internal parameters of the virtual camera can be set to an arbitrary value. If VPP images are made at higher resolution, the motion estimation accuracy would be increased, but calculation costs would also increase; consequently, we set the VPP image as 20 [pixel/m].

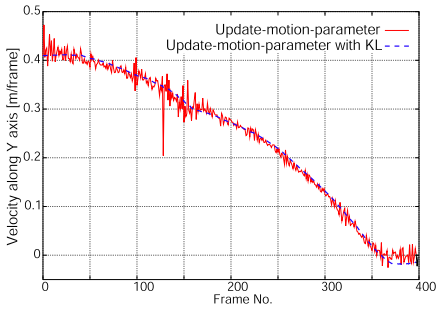

Fig. 14. Velocity along Y axis (Scene 1).

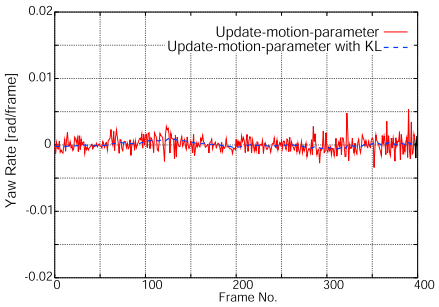

Fig. 15. Yaw rate (Scene 1).

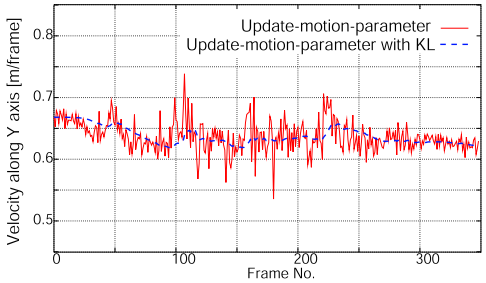

Fig. 17. Velocity along Y axis (Scene 2).

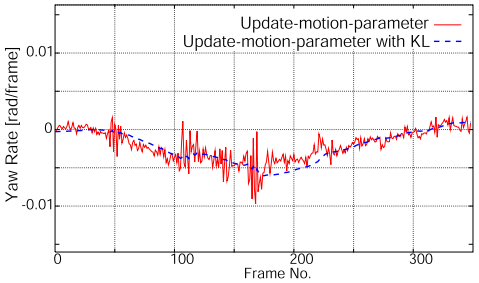

Fig. 18. Yaw rate (Scene 2).

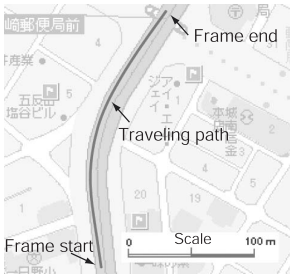

Fig. 19. Trajectory of the vehicle (Scene 2).

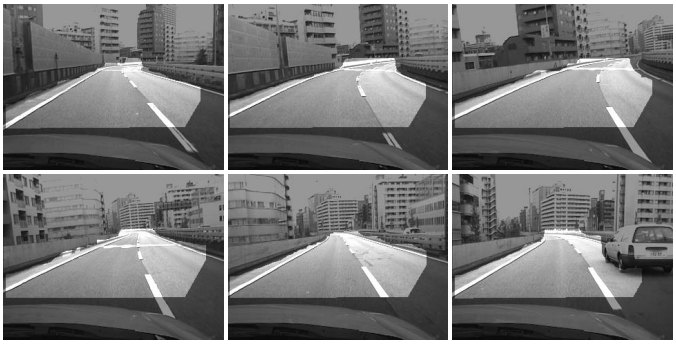

Fig. 16. Scene 2: Input image sequence (upper left to right, lower left to right). The brighter areas in the images show the extracted road regions.

camera are estimated. Next, we propose the use of a VPP image, which is an image as if we were to observe the top of the road using these parameters. Matching the sequential VPP images provides the vehicle's ego-motion. By applying the vehicle-motion model and incorporating the fact that the yaw rate is very small, motion parameters can be computed robustly. After obtaining motion parameters, the parameters are used to set the initial state of the Euclidean matrix and update the matrix for estimating more precise motions.

Finally, we applied the proposed method for synthetic images and real images and showed the effectiveness of our method.

ACKNOWLEDGMENTS

The authors would like to thank Mr. Shigeki Sugimoto for offering his software program of the road region extraction method. We also thank Mr. Akira Iihoshi, Mr. Tomo Fujimaki, and Mr. Yasushi Ueda with Honda R&D Co.,Ltd, and Mr. Kenichi Tanaka for obtaining many stereo images used for experiments.

REFERENCES

[1] T. Nakamori, N. Ishikawa, and N. Nakajima, "Vehicle front scene watching from the squence of road images," *IEICE technical report*, 2002, pp. 1–6.

[2] S. Kawamata, S. Katahara, and M. Aoki, "Location detection system using road surface marks taken by on-board slit camera," in *Symposium on Sensing via Image Information*, 2002, pp. 71–76.

[3] T. Suzuki and T. Kanade, "Measurement of vehicle motion and orientation using optical flow," in *Proc. IEEE Intelligent Transportation Systems*, 1999, pp. 25–30.

[4] G. Stein, O. Mano, and A. Shashua, "A robust method for computing vehicle ego-motion," in *Proc. IEEE Intelligent Vehicles Symposium*, 2000, pp. 362–368.

[5] Q. Ke and T. Kanade, "Transforming camera geometry to a virtual downward-looking camera: Robust ego-motion estimation and ground-layer detection," in *Proc. IEEE CVPR*, June 2003, pp. 390–397.

[6] W. Mark, D. Fontijne, and L. Dorst, "Vehicle ego motion estimation with geometric algebra," in *Proc. IEEE Intelligent Transportation Systems*, 2002, pp. 58–61.

[7] M. Okutomi, K. Nakano, J. Maruyama, and T. Hara, "Robust estimation of planar regions for visual navigation using sequential stereo images," in *Proc. IEEE ICRA*, 2002, pp. 3321–3327.

[8] O. Faugeras and F. Lustman, "Motion and structure from motion in a piecewise planar environment," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 2, no. 3, pp. 485–508, 1988.

[9] M. Abe, *Zidousya No Undo To Seigyo (Second Edition in Japanese)*. Sankaido, 2003.

[10] H. Hayashi, *Zituyo Zidousya Ziko Kantei Kogaku(in Japanese)*. Gizyutu Syoin, 1992.

[11] H. Shum and R. Szeliski, "Systems and experiment paper: Construction of panoramic image mosaics with global and local alignment," *International Journal of Computer Vision*, vol. 36, no. 2, pp. 101–130, 2000.

[12] R. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.