Camera position estimation method based on matching of Top-View images for running vehicle

Tomoaki Teshima[a], Hideo Saito[a], Shinji Ozawa[a], Keiichi Yamamoto[b] and Toru Ihara[b]

[a]Graduate School of Science and Technology, Keio University
[b]Mitsubishi Fuso Truck & Bus Corporation

ABSTRACT

In this paper, a method which estimates the trajectory of the vehicle from a single vehicle camera is proposed. The proposed method is a model based method which assumes that the vehicle is running on a planar road. The input image is converted to a *Top-View* image and a matching (registration) between the next *Top-View* image is done. The registration is done based on an assumed velocity parameter, and repeated with entire candidate parameters. In this paper, a simple model and the particle filter is introduced to decrease the computation cost. Simple model gives a constraint to the registration of the *Top-View* images, and the particle filter decreases the number of the candidate parameters. Position of the camera is obtained by accumulating the velocity parameters. Experiments shows 3 results. Enough decreasement of the computation cost, suitable estimated trajectory and small enough computation cost to estimate the trajectory of the vehicle.

Keywords: ITS, Homography, Vehicle Position Estimation, Vehicle Camera

1. INTRODUCTION

Recently, the importance of the pattern recognition is increasing in region of the ITS (Intelligent Transport Systems), especially in the safety drive support system. The trajectory information of the vehicle is useful to detect a dangerous driving.

For many safety drive support system, a lane marker recognition is used to detect the trajectory of the vehicle. Martinet *et al.* proposed an automatic driving vehicle which follows the lane marker.[1] Sotelo proposed the "VIRTUOUS" which detects many explicit markers from an urban like environment.[2] Since those methods rely on an explicit marker, those methods can only be applied under limited condition. These methods can be categorized as a "marker based methods".

On the other hand, a method which do not rely on marker has been researched. There are many works which assumes a model to the situation and evaluate the velocity parameters to estimate the trajectory. These methods can be called as a "model based methods". The model based methods repeat the assumption and the evaluation process to estimate the correct parameter. Bergen *et al.* proposed a method to estimate the velocity by assuming a hierarchical model for the environment for the running vehicle.[3] Black *et al.* proposed another method to estimate the flow fields by estimating adequate parameters of the model for the situation.[4]

Usually, an on-vehicle system requires a small calculation cost because of its real-time process. The marker based method has small computation cost against the model based method. This is because model based methods require the iteration of the evaluation of the candidate of the parameter. An easy way to implement a model based method to a real vehicle system, is to decrease the number of the candidate of the parameters. On the other hand, the accuracy of the result is trade-off with computation cost.

In this paper, we propose a method which estimates the trajectory of the vehicle from a single camera on a road that lane marker is not available. The target situation is a running vehicle on an express way. Since the "marker based method" is not applicable to the target situation, we propose a "marker based methoed". To implement the proposed method to a real vehicle, we reduce the computation cost by introducing a simple model

Further author information: (Send correspondence to Keio University)
Keio University: E-mail: tomoaki@ozawa.ics.keio.ac.jp, 3-14-1 Hiyoshi Yokohama Kanagawa 223-8522, Japan
Mitsubishi Fuso: 10 Okura Kawasaki Kanagawa 211-0023, Japan

Real-Time Image Processing 2007, edited by Nasser Kehtarnavaz, Matthias F. Carlsohn
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 6496, 649605, © 2007 SPIE-IS&T · 0277-786X/07/$18

and combining the "particle filter". By assuming a model that vehicle is running on a flat road, the velocity of the camera can be estimated by the matching (registration) of the *Top-View* images. The simple model gives a constraint to the registration of the *Top-View* images from 6 DOF(degree of freedom) to 2 DOF. The particle filter decreases the number of the candidate parameters.

The rest of this paper is organized as follows. Section 2 explains about the proposed method. Detail of each process is explained in each subsection. Section 3 shows the result of the experiments, which has been applied to some synthesized images and real images. This paper will be concluded by Sect. 4.

2. EXPLANATION OF THE METHOD

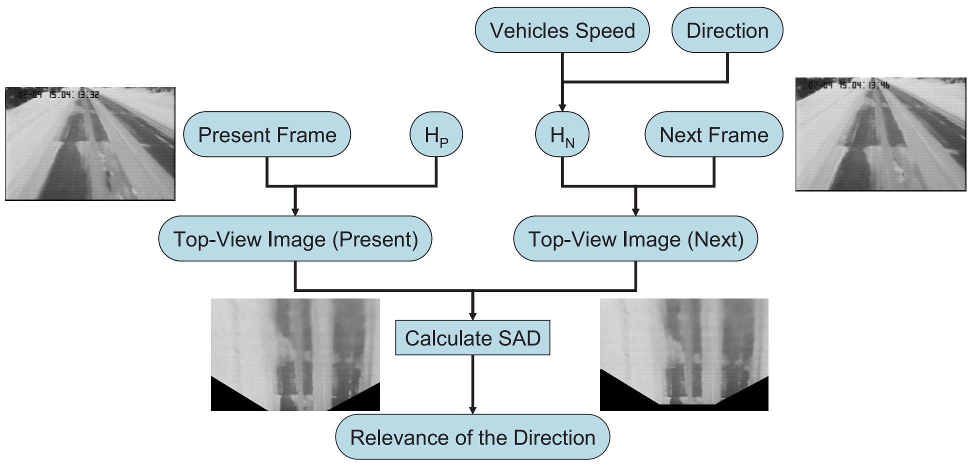
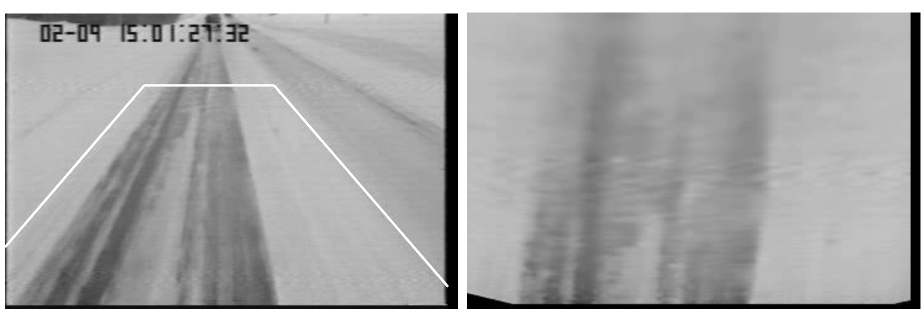


Figure 1. The outline of the registration process



(a) Input image (b) *Top-View* image

Figure 2. Example of the *Top-View* image

The proposed method is based on a model which is suitable for a running vehicle. The model which is used in the proposed method is described in Sect. 2.2. In a "model based method", the result, which is a velocity parameter of the camera, is estimated by a iteration of the assumption and the evaluation. If the parameter fits to the model, its evaluated value becomes high. Proposed method estimates the parameters by repeating the evaluation of each candidate parameters.

For the evaluation, registration of the *Top-View* images is used in the proposed method. Figure 1 shows the outline of the registration process of the *Top-View* images and fig.2 is an example of the *Top-View* images. In this process, the pairs of the parameters and the relevance of itself are obtained. Section 2.3 and 2.4 explain about the detail of the registration.

For the search of the correct parameter, evaluation of the entire candidate parameter in a certain region is an easy approach. Thus, this approach's computation cost is too expensive to implement to a real vehicle. For the reduction of the number of the candidate, the particle filter is used in the proposed method. Particle filter decreases the number of the candidate based on the result from the previous frame. Section 2.1 explains about the particle filter.

The self-position of the camera is obtained as a velocity at each frame, and then accumulated.

2.1. Particle Filter

The *Particle Filter*, or the *Condensation Algorithm*,[5] is a method for a tracking which is robust to the change of the object, such as occlusion[6] or an illuminance change.[7]

The particle filter selects a candidate of the parameters which gives a high evaluated value in the previous frame. The near-field region of the candidate is searched heavily based on the evaluated value of itself. Thus, the region where near to the correct parameter is always searched.

Not only the reduction of the computation cost, but the estimation of the parameter is guaranteed to finish in a certain period of time by the particle filter. This is an important thing to implement the proposed method to a real vehicle.

2.2. The Movement Model

The camera and the movement model used in the proposed method is simple.(See Fig.3).

Figure 3 (a) shows the relation between the world coordinate and the camera's position. The ground is the plane which is expressed by $Y_W = 0$, the origin of the world coordinate is fixed at the point on the ground directly under the camera, the axis of the camera which is camera's Z axis, is coplanar with the Y_W–Z_W plane and the camera's X axis is parallel to X_W axis.

Figure 3 (b) shows the relation of the camera positions in the two consecutive frames. The distance between the two cameras is expressed by the speed of the vehicle v. The angle of the direction θ becomes zero when the camera moves straight forward.

Seki[8] and Martinet[1] are using "bicycle model" to express the movement of the vehicle. Since the target of the proposed method is a vehicle on an express way, we can assume that the rotation movement of the vehicle is very small. According to the similar concept, we also use the movement model that is expressed only by two parameters, θ and v. The height and the angle of the camera h, ϕ are assumed that they do not change since the camera is fixed on to the vehicle. By introducing this model to the proposed method, the computation cost has been decreased. The experiments shown in Sect. 3 will show that these assumptions are reasonable.

2.3. Top View Transform

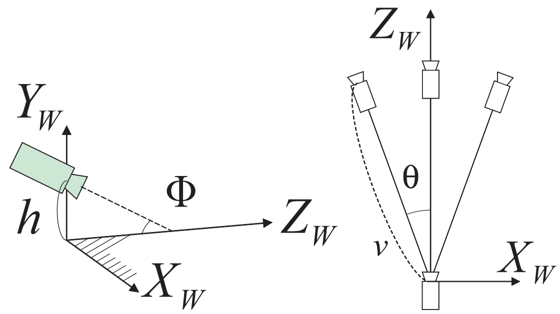
Two homographies are calculated to convert the consecutive 2 frame images to *Top-View* images. The $\mathbf{H}_P$ and the $\mathbf{H}_N$ in Eq.(1) and (2) are the homographies for the present and the next frame. $\mathbf{A}$ denotes the intrinsic parameter matrix, $\boldsymbol{r}_1$ and $\boldsymbol{r}_3$ denote the first and the third column of the rotation matrix.

Since we assumed that ϕ and h are constant, $\mathbf{H}_P$ is calculated only once when the camera is calibrated.

The homography $\mathbf{H}_N$ is calculated by only 2 parameters, v and ϕ. By involving the 2 parameters of the velocity, $\mathbf{H}_N$ will create the *Top-View* of the same region used in the present frame.

Figure 4 shows the example of the input images and the *Top-View* images. As shown in fig.4, in the *Top-View* image of the next frame, there is a black gap at the bottom of the image. This gap corresponds to the distance which the vehicle moved in 1 frame, in another word, v.

In the proposed method, homography $\mathbf{H}_N$ is calculated for each assumed pair of parameters. Thus, each pair will obtain 2 *Top-View* images.



(a) Between the world coordinate and the camera

(b) Between the two frames

Figure 3. The camera model used in the proposed method

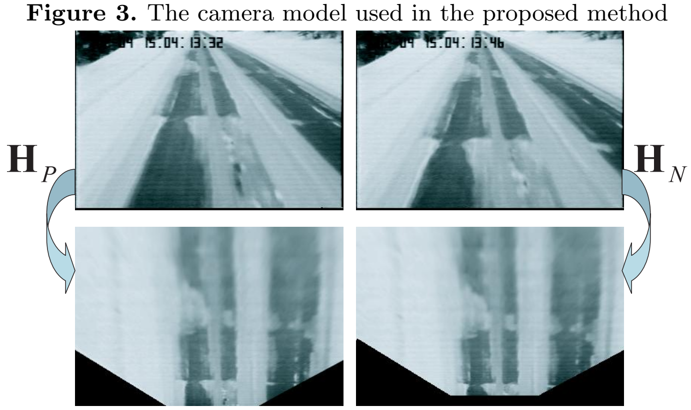


Figure 4. The conversion from the input images to the *Top-View* images

$$\mathbf{H}_P = \mathbf{A}\left(\boldsymbol{r}_1\boldsymbol{r}_3\boldsymbol{t}\right) = \mathbf{A}\begin{pmatrix} 1 & 0 & 0 \\ 0 & \sin\phi & h\cos\phi \\ 0 & \cos\phi & h\sin\phi \end{pmatrix} \tag{1}$$

$$\begin{aligned} \mathbf{H}_N &= \mathbf{A}\left(\boldsymbol{r}_1\boldsymbol{r}_3\boldsymbol{t}\right) \\ &= \mathbf{A}\begin{pmatrix} \cos\theta & \sin\theta & v\sin\theta \\ -\sin\phi\sin\theta & \sin\phi\cos\theta & h\cos\phi + v\cos\theta\sin\phi \\ -\cos\phi\sin\theta & \cos\phi\cos\theta & h\sin\phi + v\cos\theta\cos\phi \end{pmatrix} \end{aligned} \tag{2}$$

2.4. Evaluation of the Parameters

In the beginning, we assumed that in front of the vehicle, there is a region that no moving object is included. If the 2 velocity parameters, v and θ are correct, the 2 *Top-View* images must become identical. The similarity of the 2 *Top-View* images corresponds to the relevance of the 2 velocity parameters. For each *Top-View* images, the similarity is calculated. For the similarity evaluation, SAD (Sum of Absolute Difference) value is used in the proposed method.

Now, the relevance of the each velocity parameters are obtained by the processes which are described in Sect. 2.3 and 2.4. These processes are equal to create a cost function in the parameter space. The pair of the parameters which gives the minimum SAD value will be chosen as the correct velocity.

3. EXPERIMENTS

To evaluate the accuracy and the computation cost of the proposed method, some experiments are performed using both synthesized images and real images are done.

3.1. Experiments on Synhtesized Images

To evaluate the accuracy of the proposed method, an experiment on synthesized images has been done. Input images were synthesized by POV-Ray for Windows ver 3.6. Table 1 shows the camera parameters for the input images. In this experiment, 20 particles are used.

Table 1. Each Camera Parameters

parameter	value
ϕ	8.13 °
h	2.0(m)
focal length	360(pixels)
width of the image	360(pixels)
height of the image	240(pixels)

As shown in Fig.5, 3 types of input images were synthesized. The first type includes the lane marker and the guard rail(MARKER). The second type includes only the guard rail(RAIL). The third type has only the texture drawn on the road(TEXTURE). In each input images, camera moved along the same path and the lateral position changes as shown in Fig.6. To reduce the low-frequency noise, the average value was subtracted from the raw result.

The result obtained by the proposed method in each input image is shown in Fig.6. Each result show small difference from the ground truth position, since the proposed methods do not extract any explicit feature from the input images. The average distance between the ground truth and the result from the TEXTURE was 25cm, which is 6.25% against the width of the vehicle's lateral movement, 4m. This means that the proposed method can be applied to any kind of surface of the road, as long as it is a planar.

The average calculation time was 0.272 seconds per 1 frame.

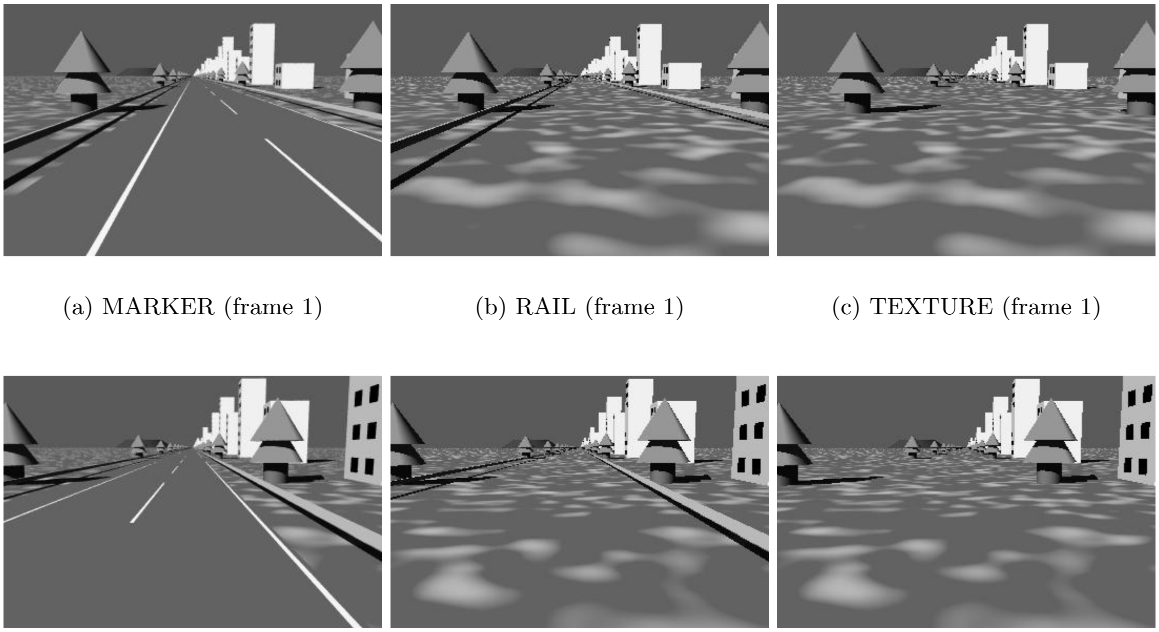
3.2. Experiments on Real Images

To evaluate the accuracy of the proposed method in the real environment, another experiment in the real environment has been done. Table 2 shows the environment for the camera parameters, Fig.7 shows the example of the input images.

In this section, the result obtained from the lane marker based method is considered as the ground truth value since the lane marker is available. To compare each result, the result of the lane marker based method has been normalized by a proper value, since the lane marker based method gives the result by the pixel and the proposed method gives the result by the millimeters. An average value has been subtracted from each result to decrease the effect of the low frequency noise.

Figure 8 shows the estimated result. We consider that the absolute difference of the result between the proposed method and the lane marker based method is not important. Please note that the final goal of the proposed method is not to detect a very accurate position of the camera, but to detect the fluctuation of the lateral position of the vehicle. Thus, we consider this result as a good enough result to detect the lateral movement of the vehicle.

The average calculate time was 0.277 seconds per 1 frame.



(a) MARKER (frame 1) (b) RAIL (frame 1) (c) TEXTURE (frame 1)

(d) MARKER (frame 300) (e) RAIL (frame 300) (f) TEXTURE (frame 300)

Figure 5. Synthesized input images

Table 2. Camera parameters for the environment

parameters	value
ϕ	5.40°
h	2.5(m)
focal length	577.7(pixel)
width of the image	360(pixel)
height of the image	240(pixel)

3.3. Experiments on the Snow road

To verify the feature that proposed method do not extract any explicit feature, such as lane marker from the image, another experiment has been done on the snow road. Figure 9 shows the example of the input image on the snow road. As Fig.9 shows, the road is straight and the vehicle runs straight ahead. Also, no lane marker is available and no running vehicle is seen in the image.

For a comparison, another method which relies on the FOE based method[9] has been applied since the method based on the lane marker can't be applied to this scene.

Figure 10 shows the estimated lateral position. The lateral position obtained by the proposed method shows a reasonable result. This means that the proposed method can estimate the lateral position even on the snow road.

On the other hand, the lateral position estimated by the FOE based method shows an unstable result.

Therefore, proposed method has more wide applicability and is more accurate than the FOE based method.

The average calculate time was 0.278 seconds per 1 frame.

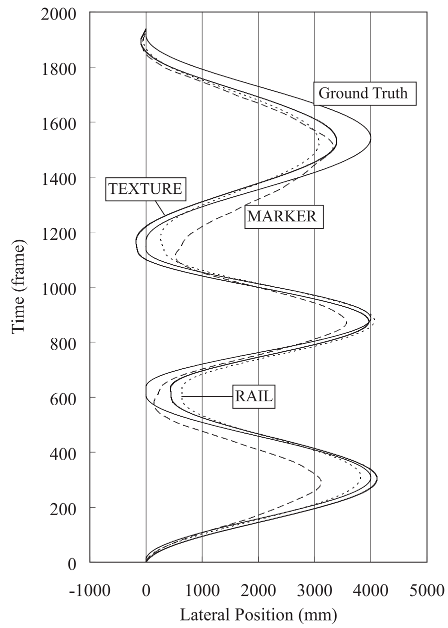
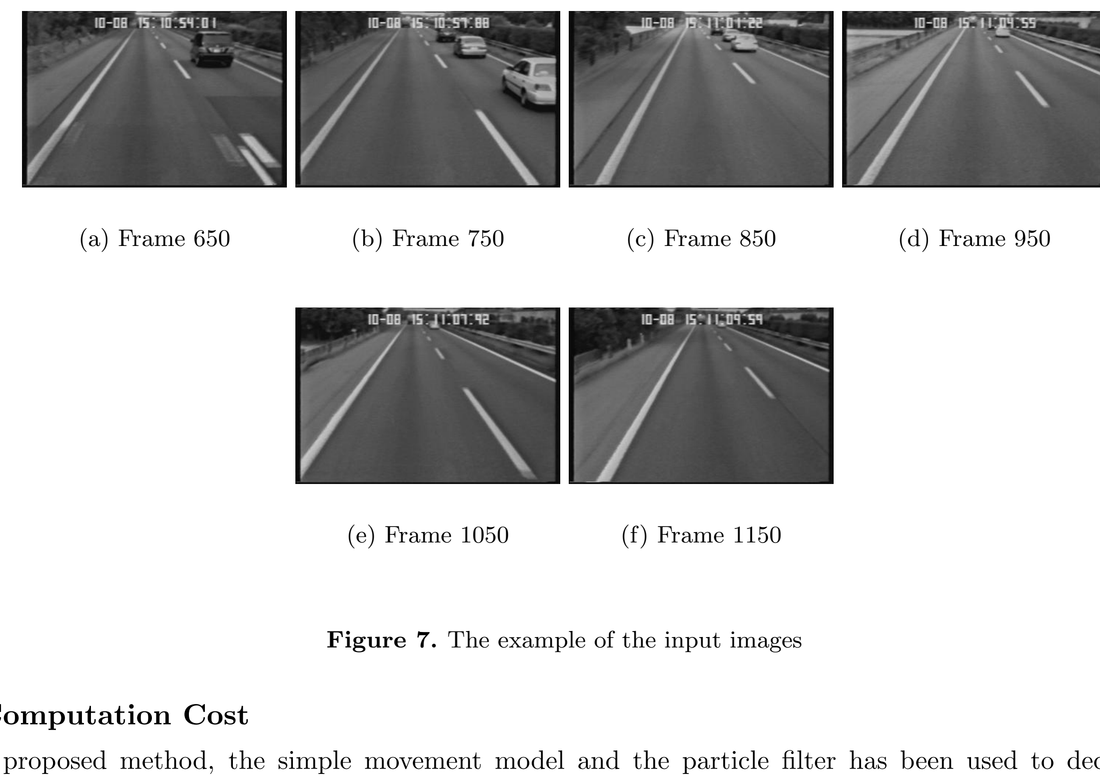


Figure 6. The estimated self position of the camera using the synthesized image



(a) Frame 650 (b) Frame 750 (c) Frame 850 (d) Frame 950

(e) Frame 1050 (f) Frame 1150

Figure 7. The example of the input images

3.4. Computation Cost

In the proposed method, the simple movement model and the particle filter has been used to decrease the computation cost. To verify the advantage of the particle filter, the comparison between the method with the particle filter and without the particle filter has been done. The method without the particle filter checks the relevance on every single candidate.

The computation time in each method in each situation are shown in table 3.4. In each situation, the

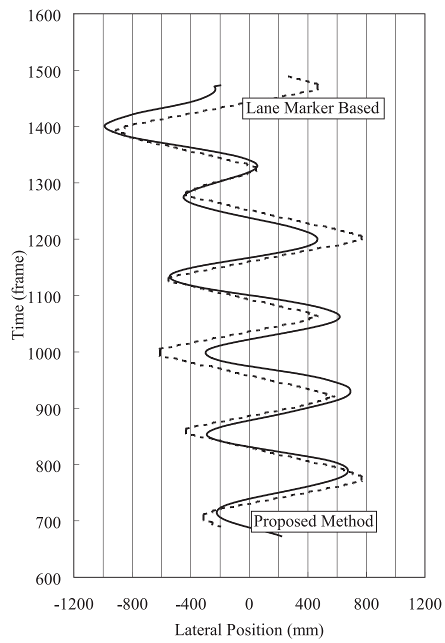


Figure 8. The estimated self-position of the camera using the real images

Table 3. The parameters used for the experiment on the snow road

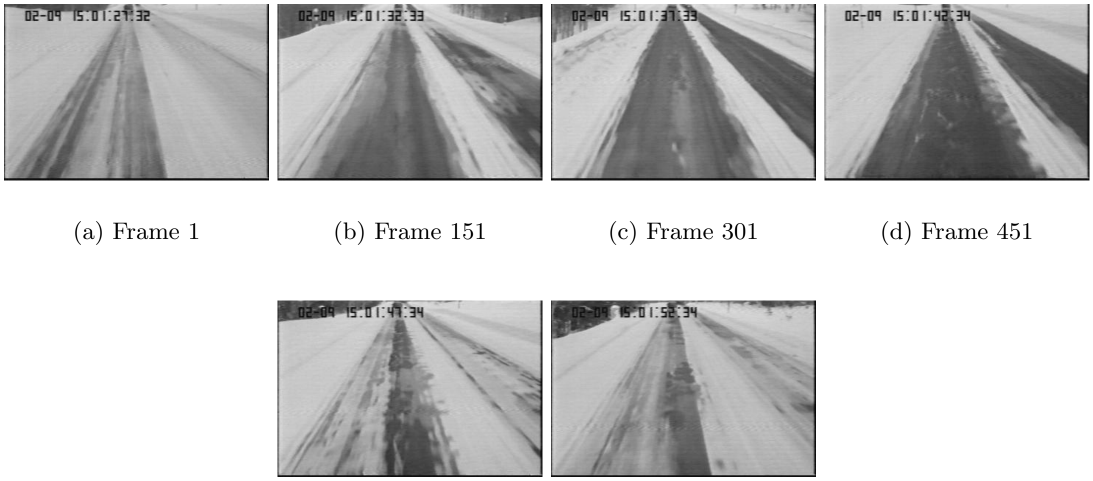
Parameter	Value
ϕ	5.40°
h	2.5(m)
Focal Length	577.7(pixel)
Width of the Input Image	360(pixel)
Height of the Input Image	240(pixel)

computation cost has been decreased by introducing the particle filter in the proposed method, which shows the advantage of the particle filter. In each situation, the proposed method shows the computation cost around 0.3 seconds. Althought this computation cost is large for a vehicle system, it is small enough to estimate the vehicle's trajectory in real time, since the trajectory of the vehicle doesn't change so rapidly.

As Oka *et. al*[7] and Pupilli *et al.*[6] show, the particle filter will need either many particles, or to decrease the degree of the freedom. In the situation described in this paper, the computation time is limited. Thus, handling many particles is not applicable. This result shows the advantage of the model used in the proposed method which decreases the dimension of the parameters from 6 to 2. Also, the results of the other experiments show that the model used in the proposed method is reasonable for this situation.

Table 4. Computation Time for 1 Frame

Situation	Proposed Method	Entire Search
Lane Marker	0.28sec	3.07sec
Snow Road	0.28sec	3.16sec
Synthesize	0.27sec	2.77sec



(a) Frame 1 (b) Frame 151 (c) Frame 301 (d) Frame 451

(e) Frame 601 (f) Frame 751

Figure 9. The input image on the snow road.

3.5. Real Time System

The proposed method's final goal is to be implemented on a real vehicle and estimate the position in real time. We created a pilot system running in real-time for this demonstration. The system was implemented on a laptop computer with a web camera connected via the USB. The system specifications are described in table 5 and 6. The examples of the input images are shown in Fig.11.

In this pilot system, the vehicle speed is collected by a speed sensor installed in the vehicle. By treating the information of the speed sensor's information as correct value, the degree of freedom has been decreased to only θ. In such situation, without applying the particle filter, the computation time has reached to 0.25 sec. This time is short enough to estimate the lateral position of the vehicle.

The example of estimated result is shown in Fig.12. This result is obtained when the driver rolls the vehicle from side to side of the lane for imitating drowsing drive. The result obtained from the position of the lane marker is considered as a ground truth in this experiment. The detected trajectory by the pilot system is partially close to the ground truth. We consider that it is satisfactory result even though some part is not close to the ground truth, because it is possible to detect the drowsiness of the driver from the fluctuating pattern of the trajectory.

Table 5. Specification of the Web camera

Parameters	Value
ϕ	18.68°
h	2.4(m)
Focal Length	419.539(pixel)
Width of the Image	320(pixel)
Height of the Image	240(pixel)
Frame Rate	15fps

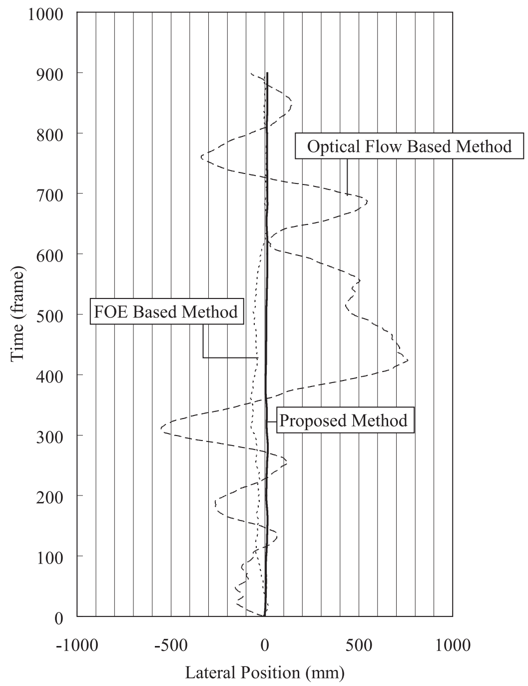


Figure 10. The estimated lateral position on the snow road

Table 6. Specification of the laptop computer

CPU	Pentium4 2.6GHz
Memory	2.0GB
OS	WindowsXP Pro(SP2)
Compiler	Visual Studio .NET VC++
Platform	VC++ + OpenCV

4. CONCLUSION

In this paper, we proposed a method which estimates a trajectory of the vehicle in real time by registrating 2 *Top-View* images. The *Top-View* image is converted from the input image by using the homography. Thoough it's computation time is around 0.3 seconds per frame, it is fast enough to estimate the trajectory of the vehicle.

To reduce the computation cost, a simple model has been introduced. The degree of the freedom has been decreased from 6 to 2. By combining the particle filter, the number of the iteration has been decreased. These two approace has decreased the computation cost, as shown in Sect. 3.4. Also, the proposed method shows enough accuracy in many situations shown in Sect. 3.1, 3.2, 3.3.

ACKNOWLEDGMENTS

The experiment described in Sect.3 was supported by Kohji Sugita and Yohji Hiramoto at Mitsubishi Fuso Truck & Bus Corporation. The authors appreciate for their help in the experiments. This work was supported in part by the "Optical and Electronic Device Technology for Access Network" COE project of Keio University.

(a) Frame 65 (b) Frame 365 (c) Frame 665 (d) Frame 965

(e) Frame 1265 (f) Frame 1565

Figure 11. Example of the input image on real-time

REFERENCES

1. P. Martinet and C. Thibaud, "Automatic guided vehicles: Robust controller design in image space," *Autonomous Robots* **8**(1), pp. 25–42, January 2000.
2. F. J. R. M. A. Sotelo and L. Magdalena, "Virtuous: Vision-based road transportation for unmanned operation on urban-like scenarios," *IEEE Transactions on Intelligent Transportation Systems* **5**(2), pp. 69–83, June 2004.
3. K. J. H. J. R. Bergen, P. Anandan and R. Hingorani, "Hierarchical model-based motioin estimation," in *European Conference on Computer Vision*, **10**(31), pp. 237–252, May 1992.
4. M. J. Black and P. Anandan, "The estimation of multiple motions: Parametric and piecewise-smooth flow fields," *Computer Vision and Image Understanding* **63**(1), pp. 75–104, 1996.
5. M. Isard and A. Blake, "Condensation – conditional density propagation for visual tracking," *International Journal of Computer Vision* **29**(1), pp. 5–28, August 1998.
6. M. L. Pupilli and A. D. Calway, "Real-time camera tracking using a particle filter," in *Proceedings of the British Machine Vision Conference*, September 2005. Oxford.
7. Y. N. K. Oka, Y. Sato and H. Koike, "Head pose estimation system based on particle filtering with adaptive diffusion control," in *Proc. IAPR Conference on Machine Vision Applications*, pp. 586–589, May 2005.
8. A. Seki and M. Okutomi, "Ego-motion estimation by matching dewarped road regions using stereo images," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 901–907, May 2006.
9. S. O. K. Y. T. Teshima, H. Saito and T. Ihara, "Estimation of foe without optical flow for vehicle lateral position detection," in *Proc. IAPR Conf. Machine VIsion Applications*, pp. 406–409, May 2005.

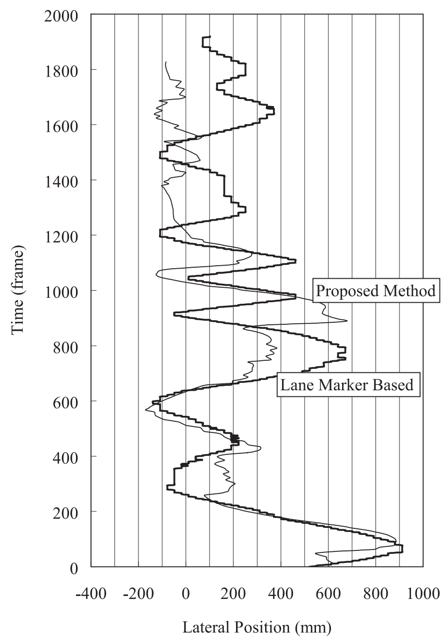


Figure 12. The result estimated in real time