# Obstacle Detection Using a Moving Camera

Shao Hua Qian[1], Joo Kooi Tan[1], Seiji Ishikawa[1], Takashi Morie[2]

[1]*Department of Mechanical & Control Engineering,* [2]*School of Brain Science & Engineering*
*Kyushu Institute of Technology*
*Sensuicho 1-1, Tobata, Kitakyushu, 804-8550, Japan*
*E-mail: {qian, etheltan, ishikawa}@ss10.cntl.kyutech.ac.jp, morie@brain.kyutech.ac.jp*

***Abstract***: This paper proposes a method of detecting obstacles from a video taken by a moving camera mounted on a vehicle by background subtraction. The background subtraction is often used to detect moving objects when camera is static. But according to the characteristics of a road, we can also employ Gaussian mixture model to detect all objects (either static or moving objects) on the road in the case of moving camera. Then we use two consecutive image frames, and warps the first image according to the geometrical relationship between these two images. The road area is then extracted by comparing the warped image with the second image. Using this road area, we can delete all things which are not obstacles. In the performed experiments, it is shown that the proposed method is able to detect obstacles such as vehicles and pedestrians on a road.

***Keywords***: Obstacle detection, moving camera, monocular vision, Gaussian mixture model, road area detection.

## I. INTRODUCTION

Thousands of people die by car accidents year by year. Many of those accidents could be avoided or alleviated by vision-based driving assistance systems. These systems cause drivers to respond more quickly in the face of danger. In these systems, the ability to detect obstacles from a vehicle moving on a planar road surface is essential. In recent years, many obstacles detection approaches have been developed. There are mainly three popular methods, based on a-priori knowledge, based on optical flow, and based on stereo vision. The method based on a-priori knowledge is often used to detect specific objects or limited objects classes, such as vehicles or pedestrians. We often call this method pattern recognition. Optical flow and stereo vision methods can detect arbitrary objects which pose a threat to safe driving. But these two methods are sensitive to vehicle motion, and when obstacles have small or null speed, optical flow techniques fail.

In this paper, we propose a method of estimating a road area in general road environments. This method uses two consecutive image frames, and warps the first image according to the geometrical relationship between these two images. The road area is then extracted by comparing the warped image with the second image.

## II. OUTLINE OF THE PROPOSED METHOD

The process flow of the proposed method is shown in **Fig. 1**.

In the first place, we employ a Gaussian mixture model [1,2] in reconstructing the background from a video image sequence taken by a moving camera. According to road characteristics, we can assume camera and road are static, and then we can get an imaginary scene. In this scene, the background is a road area; objects and pedestrians on the road, buildings, road marks and zebra crossings are foreground objects. Because foreground image includes everything on the road, buildings, shadows and road marks (such as zebra crossing, lane lines). Our goal is to extract obstacles on the road, so we need to delete other things in the foreground image.

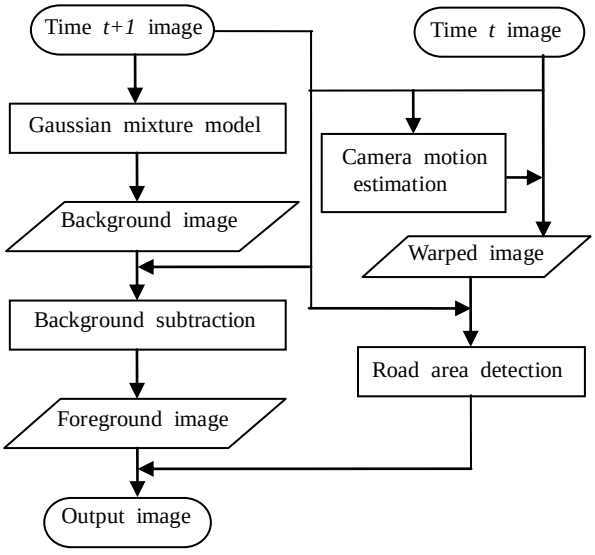


Fig. 1. Overview of the proposed method.

In the second place, we estimate the camera motion parameters from the correspondences of feature points between two successive images [3]. The camera motion parameters consist of 3D rotation and 3D translation parameters. In this method, it is assumed that the camera is calibrated, i.e. the internal parameters of the camera are known. Two consecutive images are used at any time, i.e. the image taken at time *t* and that taken at time *t*+1 are used at time *t*+1. In the third place, we warp time *t* image using camera motion parameters [4,5,6]. Comparing the warped image with time *t*+1 image, we can get the road area at time *t*+1. Finally, delete other things in the foreground image using this road area, and then we can get the obstacles on the road.

## III. GAUSSIAN MIXTURE MODEL

The values of a particular pixel over time are a time series of pixel values. At any time *t* (*t*=1,2,…,*T*), a particular pixel ($x_0$, $y_0$) has $X_t$ pixel values:

$$\{X_1,.....,X_T\}=\{I(x_0,y_0,t):1\le t\le T\} \quad (1)$$

where *I* is the image sequence. GMM method models each pixel by a mixture of *K* Gaussian distributions. The probability to observe the current pixel value *X* is

$$P(X_t)=\sum_{k=1}^{K}\omega_{k,t}*\eta(X_t,\mu_{k,t},\sigma_{k,t}^2) \quad (2)$$

where *K* is the number of distributions (currently, from 3 to 5 are used); $\omega_{k,t}$ is an estimate of the weight of the $i^{th}$ Gaussian in the mixture at time *t*; $\mu_{k,t}$ is the mean value of the $i^{th}$ Gaussian in the mixture at time *t*; $\sigma_{k,t}^2$ is the covariance matrix of the $i^{th}$ Gaussian in the mixture at time *t*; $\eta$ is a Gaussian probability density function defined by

$$\eta(X_t,\mu_{k,t},\sigma_{k,t}^2)=\frac{1}{\sqrt{2\pi}\sigma}\exp\{-\frac{1}{2}\frac{(X_t-\mu_{k,t})^2}{\sigma_{k,t}^2}\} \quad (3)$$

## IV. CAMERA MOTION ESTIMATION

### 1. Feature point detection

In the image at time *t*, we detect feature points using Harris corner detector [7], then, using Lucas-Kanade method [8], we detect the corresponding points of the Harris feature points in the image at time *t*+1, and by using RANSAC [9] we delete outliers. The camera motion parameters are then estimated from the correspondence of the feature points.

### 2. Fundamental matrix

The fundamental matrix *F* is the algebraic representation of epipolar geometry. And *F* is the unique 3*3 rank 2 homogeneous matrix which satisfies

$$m_{t+1}{}^{T}Fm_t=0 \quad (4)$$

for all corresponding points $m_t \leftrightarrow m_{t+1}$.

Here, $F=\begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix}$

$$m_t=\begin{bmatrix} x_t \\ y_t \\ 1 \end{bmatrix} \qquad m_{t+1}=\begin{bmatrix} x_{t+1} \\ y_{t+1} \\ 1 \end{bmatrix}$$

In this paper, we use the 8-point algorithm [10] to compute the fundamental matrix. The key to success with the 8-point algorithm is proper careful normalization of the input data before constructing the equations to solve. In the case of the 8-point algorithm, the suggested normalization is a translation and scaling of each image so that the centroid of the reference points is at the origin of the coordinates and the RMS distance of the points from the origin is equal to $\sqrt{2}$.

Algorithm:

Step_1: Normalization: Transform the image coordinates according to $\hat{m}_t=T_t m_t$ and $\hat{m}_{t+1}=T_{t+1}m_{t+1}$, where $T_t$ and $T_{t+1}$ are normalizing transformations consisting of a translation and scaling.

Step_2: Finding the fundamental matrix $\hat{F}$ corresponding to the matches $\hat{m}_t \leftrightarrow \hat{m}_{t+1}$.

Step_3: Denormalization: Set $F=T_{t+1}^{T}\hat{F}T_t$. Matrix *F* is the fundamental matrix corresponding to the original data $m_t \leftrightarrow m_{t+1}$.

### 3. Camera motion parameters

The relationship between the fundamental and essential matrices is

$$E=K^{T}FK \quad (5)$$

Where, *K* is a camera calibration matrix.

The essential matrix can be represented by motion parameters of a camera between two images, i.e., the rotation matrix *R* and the translation *T*.

$$E=[T]_X R \quad (6)$$

Here, $[T]_X = \begin{bmatrix} 0 & -t_Z & t_Y \\ t_Z & 0 & -t_X \\ -t_Y & t_X & 0 \end{bmatrix}$ is the corresponding skew-symmetric matrix of the translation T.

By applying the singular value decomposition to the matrix *E*, we have

$$E = U\Sigma V^T \tag{7}$$

Using the results of the singular value decomposition (Eq.(7)), we calculate the rotation matrix *R* and the translation *T* as follows;

$$R = UWV^T \text{ or } R = UW^TV^T \tag{8}$$

$$[T]_X = U\Sigma WU^T \text{ or } [T]_X = U\Sigma W^TU^T \tag{9}$$

Here, $W = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

There are four possible choices of the camera motion parameters, based on the two possible choices of *R* and two possible choices of *T*.

The selection of the camera motion parameters is done by using the four combinations to do the motion compensation then check which combination's compensation image is correct.

## V. ROAD AREA ESTIMATION

### 1. Motion compensation

The location of the camera at time *t* and *t+1*is shown in **Fig. 2**.Suppose the world coordinate system coincides with the camera coordinate system at time *t*. Then the projection equation at time *t* and time *t+1* are given by

$$sm_t = K[I \quad 0]M \tag{10}$$

$$sm_{t+1} = K[R \quad T]M \tag{11}$$

where $m_t$ and $m_{t+1}$ are the 2-D points on the image plane at time *t* and time *t*+1; ***M*** is the 3-D point in the world coordinate system; K is the camera calibration matrix; *R* and *T* are the camera motion parameters; *I* is the unit matrix; *s* is a scalar.

Eq.(10) and Eq.(11) are easily written in terms of the known coordinates $m_t$ and $m_{t+1}$ as follows;

$$s\begin{bmatrix} x_t \\ y_t \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{12}$$

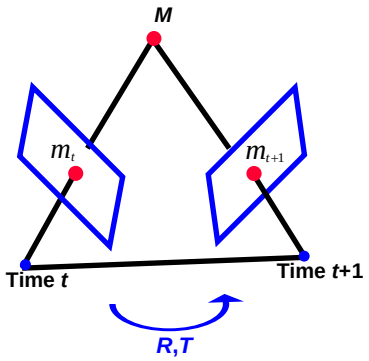


Fig.2. The location of the camera at time *t* and *t*+1

$$s\begin{bmatrix} x'_{t+1} \\ y'_{t+1} \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_X \\ r_{21} & r_{22} & r_{23} & t_Y \\ r_{31} & r_{32} & r_{33} & t_Z \end{bmatrix}\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{13}$$

First, for each pixel in the image at time *t*, we calculate the corresponding coordinate of 3-D point *M* based on Eq. (12). Here, we assume all 3-D points *M* are on the road plane, so *Y* is equal to the height of a camera above the ground.

Then we use *X*, *Y* and *Z* obtained from the above procedure and we calculate $x'_{t+1}$ and $y'_{t+1}$ based on Eq. (13). We create a new image, in which the pixel value at ($x'_{t+1}$, $y'_{t+1}$) is the pixel value at ($x_t$, $y_t$) in the image at time *t*. This new image is the warped image.

### 2. Road area

In order to detect the road area at time *t*+1, we calculate NCC (Normalized Cross-Correlation) between the warped image and the image at time *t*+1. In the experiment, we use a $7 \times 7$ window for calculating NCC. The road area can be obtained by extracting those pixels that have a NCC value below a specified threshold.

## VI. EXPERIMENTAL RESULTS

Experiments have been done on a video under the existence of a person passing in front of a camera. The video is taken by a camera mounted at the front seat of a car, and includes image sequences of frontal road environments while the car is driving in the town. **Fig. 3** shows the result of the detection of a pedestrian crossing the road. (a) are the input images; (b) are the foreground images obtained from GMM; (c) are road area detection results; and (d) are the results of obstacles detection.

## VII. DISCUSSION AND CONCLUSIONS

This paper proposes a method of detecting obstacles on a video taken by a vehicle-mounted monocular camera. As shown in the experimental results, performance of the proposed method is reasonable. This method has some advantages over other existing methods. This method can detect arbitrary objects which may pose a threat to safe driving on the road, not only a specific object or limited object classes, but it can also detect both static and moving objects. Moreover it can be employed in both static and moving camera cases.

## ACKNOWLEGDEMENT

This work was partly supported by a grant of Knowledge Cluster Initiative implemented by MEXT.

## REFERENCES

[1] C. Stauffer and W.E.L. Grimson(1995), Adaptive back-ground mixture models for real-time tracking. Proc. Conference on Computer Vision and Pattern Recognition, Vol.2, 246-252.

[2] P. KaewTraKulPong and R. Bowden(2001), An improved adaptive background mixture model for real-time tracking with shadow detection. Proc. 2nd European Workshop on Advanced Video Based Surveillance Systems, Computer Vision and Distributed Processing.

[3] R. Hartley and A. Zisserman(2004), Multiple View Geometry in Computer vision. Cambridge University Press, 2nd edition.

[4] K.Yamaguchi, A. Watanabe and T. Naito(2008), Road region estimation using a sequence of monocular images, Proc. the 20th International Conference on Pattern Recognition.

[5] K.Yamaguchi, T. Kato and Y. Ninomiya(2006), Vehicle ego-motion estimation and moving object detection using a monocular camera. Proc. the 18th International Conference on Pattern Recognition.

[6] K.Yamaguchi, T. Kato and Y. Ninomiya(2005), Obstacle detection in road scene using monocular camera. Proc. Information Processing Society of Japan, 69-76. (in Japanese)

[7] C. Harris and M. Stephens(1988), A combined corner and edge detector. Proc. Alvey Vision Conference, 147-151.

[8] B. Lucas and T. Kanade(1981), An iterative image registration technique with an application to stereo vision. Proc. International Joint Conference on Artificial Intelligence, 674-679.

[9] M. Fischler and R. Bolles(1981), Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Proc. Communications of the ACM, 24(6): 381-395.

[10] R. Hartley(1977), In defence of the eight-point algorithm. Proc. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(6):580-493.

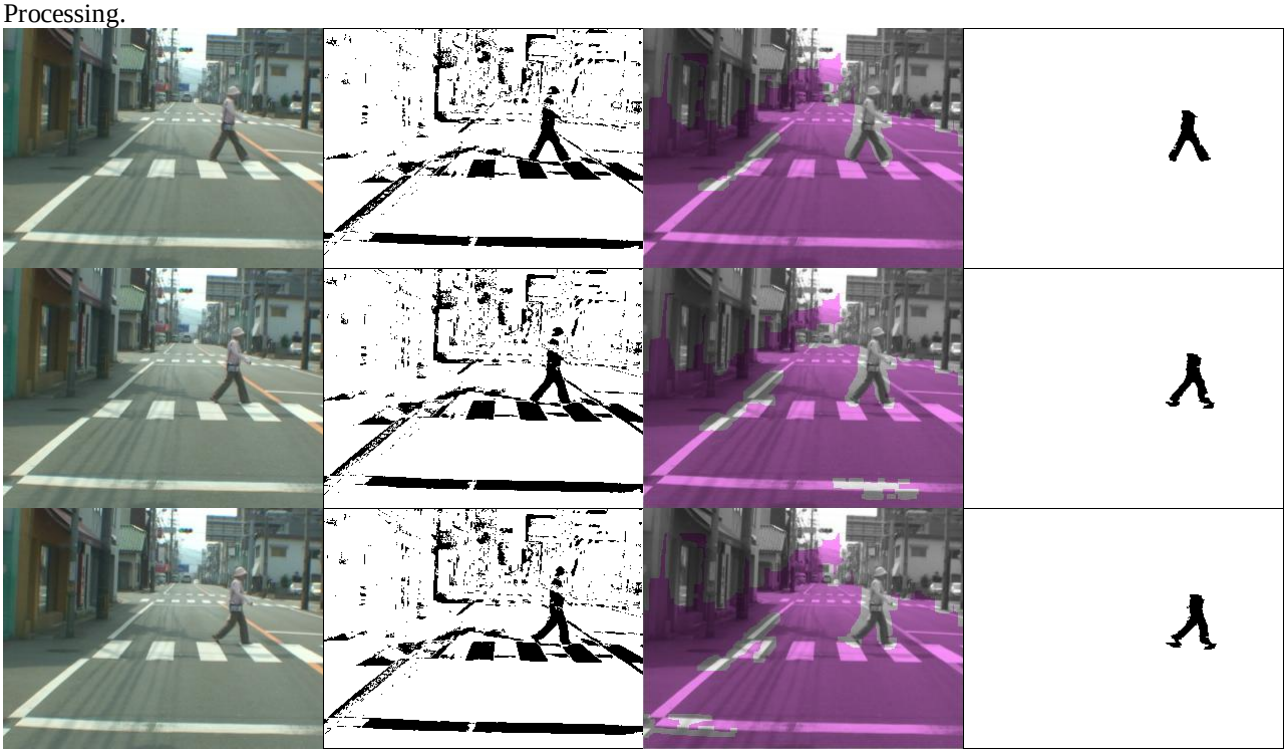

Fig.3.Experimental result: (a) Original video image frames; (b) foreground images; (c) road areas; (d) obstacle.