Journal of Information Processing Vol. 17 1–13 (Jan. 2009)

*Regular Paper*

# Vehicle Detection Using Probability Fusion Maps Generated by Multi-camera Systems

Francisco Lamosa,[†1] Zhencheng Hu[†1]
and Keiichi Uchimura[†1]

In this paper we describe a multi-camera traffic monitoring system relying on the concept of probability fusion maps (PFM) to detect vehicles in a traffic scene. In the PFM, traffic images from multiple cameras are inverse perspective-mapped and registered onto a common reference frame, combining the multiple camera information to reduce the impact of occlusions. Although the unconstrained perspective projection is non-invertible, imposing the condition that the image points be co-planar allows the relaxation of this constraint. In images of road scenes, the road surface is locally planar, and as such can be inversely mapped. We show that computing the perspective undistortion by finding 4 matching points between a camera image and an ideal non-perspectively distorted image, a system of linear equations can be solved that corresponds to applying the rotation, translation, and re-projection onto the common reference plane without needing calibrated cameras. We show that this method yields good results in the detection of vehicles for subsequent tracking and monitoring.

## 1. Introduction

Traffic monitoring is becoming a significant tool in efforts to address traffic problems arising from the increased traffic volumes in the world. These problems include long term issues such as environmental degradation and economic inefficiency. In this role, traffic monitoring can assist in the development of the reliable traffic models needed to optimize the transportation networks by acquiring the empirical data for the modeling. However, there are also immediate concerns such as the handling of emergency situations that perturb the normal behavior of traffic. Events such as accidents, major fires, and terrorism often require real-time traffic monitoring, in order to provide the timely information needed for appropriate traffic management.

†1 Graduate School of Science and Technology, Kumamoto University

Approaches to traffic monitoring can be classified into two broad categories: spot monitoring, such as the ground (magnetic) loop, and area monitoring, such as video cameras.

Ground (magnetic) loop detectors are accurate but are expensive and difficult to implement and maintain, requiring significant work, such as the excavation of road surfaces, while providing limited information, since their data is only for a point in space.

Video monitoring systems, however, are easy to install and maintain, making them significantly more flexible, since changes can easily be implemented. They also have the potential to provide more information than can be obtained from point detectors as they capture information over a wide area. Information that requires area monitoring includes lane changing frequency and vehicle trajectories. Moreover, other data that can better be evaluated using an area monitoring system include vehicle type classification. However, video camera performance is more susceptible to environmental conditions, although there is work on techniques to reduce their impact, such as through the use of infra-red lighting. Moreover, due to the limitations of camera position and orientation, they are also subject to occlusions in the presence of multiple opaque features that overlap within the camera field of view. These occlusions lead to the miscount of traffic and the loss of the target during the tracking process.

In this paper we present an approach to address the problem of occlusion that uses multiple cameras to acquire several data sets to offset the effects of the missing information where occlusions occur. The rest of this paper is organized as follows: Section 2 briefly discusses previous work; the proposed algorithm is presented in Section 3; Section 4 describes the experimental results a brief conclusion is presented in Section 5.

## 2. Previous Work

Early video surveillance work generally used a single camera with tripwires set to act as visual "ground loops" [1]. Thus, images were captured when an off-road sensor was triggered and processing was performed on a static 2-D image of a given region of space.

A common approach to the vehicle detection task in these systems was to break

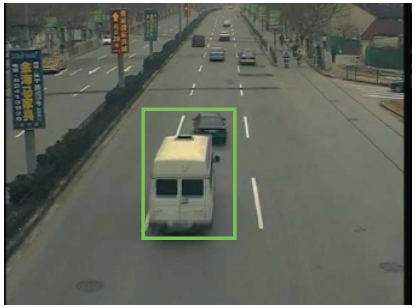

**Fig. 1** The problem of occlusion with a single camera.

down the problem into several steps [2]:

a. Segment the scene to identify individual vehicles; track these targets within the tracking zone.
b. Compute traffic parameters from vehicle tracking results.
c. Transmit traffic data to a traffic management centre.

There are a number of different approaches to each of these steps, as discussed briefly in Ref. 2). Some of the approaches to tracking include 3-D model tracking [3], region based tracking using an adaptive Kalman-filter background model [4], active contour-based tracking [5),6)], and feature-based tracking.

In Ref. 2), the focus is on the feature-based tracking using groupings of features having similar motions to segment the vehicles.

However, single camera based approaches have two limitations: the inability to use depth information to assist in the vehicle detection and tracking and the possibility of occlusion of features of interest. This problem is illustrated in **Fig. 1**.

In efforts to address this issue, recent work has explored the use of multi-camera systems for traffic monitoring [3),7),11),12)]. In Ref. 3) 3-D models are used to generate a series of sparse 2-D templates that can be used to identify vehicles and their respective classes in the presence of occlusions. These 2-D models are assumed to operate on small regions of the image under the assumption of locally orthographic projections. In Ref. 7), cameras are calibrated using a method that relies on the geometric constraints of the road. It makes use of projective geometry to allow the calculation of the objects' heights. In Refs. 11) and 12), a single plane Probability Fusion Map (PFM) based on the inverse-map of a multi-camera image data was proposed. The single plane PFM successfully addressed the problem of occlusion. However, since the co-planarity of image points is not strictly true, the single plane PFM is subject to distortions, which leads to less accurate measurement of target positions and dimensions.

Much work has been done in terms of 3-D model matching for tracking and vehicle classification [8)–10)]. However, in these approaches it is important that there be no occlusion in the first few frames for correct matches to be established. Otherwise, any subsequent tracking is suspect.

Stereo approaches have also been considered recently, but the correspondence problem and the computational intensity of searching over a large range of depths makes this approach difficult for real-time applications.

## 3. The Probability Fusion Map (PFM)

Much of the work using multiple cameras relies on the extraction of 3D models or scenes for analysis, which is a time consuming process. In the PFM, the problem has been cast as a data fusion issue. The goal is to combine all the available vehicle information to compensate the missing information in one or more views, which is available in the remaining views. To achieve this, the images from the multiple cameras are registered onto a common framework and are combined to assign probabilities for each pixel in the regions representing a vehicle. This process is described below.

### 3.1 Background Subtraction

Although the goal of PFM is to make the vehicle detection more robust by increasing the amount of available information, the suppression of clearly identifiable non-vehicle regions reduces the probability of errors in vehicle detection. Thus, removing background features by subtraction from the image is used in our approach. However, with this approach, it is important to have an accurate representation of the background, as artefacts may be detected as vehicles if this condition is not met.

Over short periods of time, it is possible to approximate a reasonably accurate background from single images with no foreground features present. This was

shown in our initial tests, as discussed below. However, over longer periods of time, where illumination may be changing, a robust model that evolves over time is more appropriate.

In the well-known median filter background model, a sequence of images is maintained in a buffer queue. It is assumed that a pixel stays in the background for more than half the time, so that the foreground object pixel values are outliers that are rejected by median filtering. As this is memory intensive, a different approach, the Adaptive Median Filter, AMF, was developed for this work.

In the AMF, an initial estimate is generated, based on a single frame. For each subsequent frame, for each pixel, the new background is computed as:

$$B(x,y,t) = \begin{cases} B(x,y,t-1) + \delta & \text{if } I(x,y,t) > B(x,y,t-1) \\ B(x,y,t-1) - \delta & \text{if } I(x,y,t) < B(x,y,t-1) \end{cases} \quad (1)$$

where $B(x,y,t)$ is the background pixel at point $(x,y)$ and at time $t$ and $I(x,y,t)$ is the pixel value at $(x,y)$ at time $t$.

This approach only requires two buffers, storing the current pixel value and the previous estimate, and is thus much more efficient in terms of memory usage. Moreover, by using a learning factor, $\delta$, it imposes a limited correction, especially for large deviations from the estimated background value, thus acting as a limit on outliers (as a median filter does). The selection of $\delta$ determines the rate at which the background model adapts to changes in environmental conditions and must be adjusted for the conditions. If $\delta$ is too big, the background model changes can be too large, leading to fluctuating errors in the background subtraction. Alternatively, if it is too small, the background model changes too gradually leading to a decrease in the performance of the background subtraction.

Shadowing was another issue of concern in the background subtraction. There were two sources of shadows and each required different handling. First, there are fixed structures that cast shadows on the road surface. Although these shadows change over time, they are subject to the same illumination as the road surface, with the result that the road model updates incorporate changes to these shadows. However, moving feature shadows are more variable, depending on the positions of the vehicles, the relative positions between the vehicles, the camera, and the illumination source.

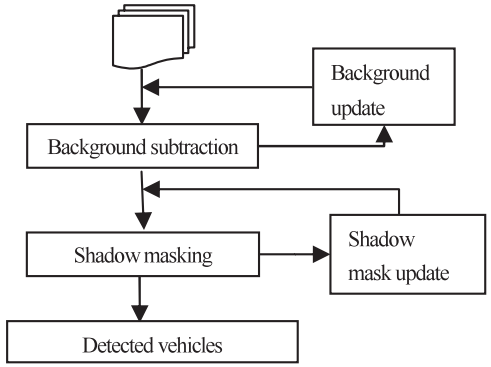


**Fig. 2** Background subtraction process.

For these moving shadows, an initial shadow model was developed by segmenting the shadows in a number of sample images to estimate the shadow pixel values. The shadow model pixel values were estimated row-by-row. Shadow pixels from matching rows were grouped to estimate the shadow model pixel values for the given row. These row estimates of the shadow pixel values were then combined for the initial shadow model.

This model was then updated, using an AMF filter, to account for illumination changes.

First, a background subtraction of the most recent frame was carried out. The current shadow model was then subtracted from the result to extract the vehicles. The shadow pixel values were extracted from the difference between the vehicle only images and the vehicle + shadow images. However, only the shadows of the high-contrast vehicles were used to calculate the estimated new shadow pixel values. These new values for the shadow model were then combined with the existing shadow model values using the AMF model.

**Figure 2** illustrates the subtraction process. For every newly captured image, the background is subtracted. The shadow mask is then applied to remove shadows from the result of the background subtraction process. However, after removing the foreground from each new frame, the resultant background is used to update the background model using the AMF. Likewise, the segmentation of the background result extracts the new shadow pixel values which are used to

update the shadow model.

### 3.2 Image Fusion

The PFM relies on the inverse mapping of the different viewpoint images onto a common framework. Since each image is a perspective projection of a set of world points by a camera having a specific rotation and translation, $\boldsymbol{R}$, and $\boldsymbol{T}$, it is necessary to adjust for the individual camera $\boldsymbol{R}$ and $\boldsymbol{T}$ prior to fusing the images. This is achieved through the inverse perspective projection described below.

(1) **Inverse Projection**

A camera having a focal length, $f$, a principal point, $(o_x, o_y)$, being rotated by an amount, $\boldsymbol{R}$, and translated by a distance, $\boldsymbol{T}$, with respect to a world coordinate system, will project a set of physical points, denoted in the homogeneous world coordinate system as $\boldsymbol{X^W} = (X^W, Y^W, Z^W, 1)$, into a set of image points denoted, in a homogeneous image coordinate system, by $\boldsymbol{x^I} = (x^I, y^I, w^I)$, as follows:

$$x_i^I = A[RT]X_i^W \tag{2}$$

where $i$ is the camera identifier; $i = 1, 2, 3, \ldots, N$ with $N$ being the number of cameras in the system. $\boldsymbol{R}$ is the $3 \times 3$ rotation matrix denoting the orientation of the camera. $\boldsymbol{T}$ is the $3 \times 1$ translation vector between the world and camera coordinate systems. $\boldsymbol{A}$ is the intrinsic parameters matrix given by the focal length and the principal point. Thus, the above equation can be expanded as:

$$\begin{bmatrix} x^I \\ y^I \\ w^I \end{bmatrix} = \begin{bmatrix} f_x & 0 & o_x \\ 0 & f_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3}$$

This equation shows the homogeneous image coordinates, that is, the image coordinates scaled by the factor, $w$. The non-homogeneous image coordinates, $x'$ and $y'$ are simply obtained as shown here:

$$x' = x/w$$
$$y' = y/w \tag{4}$$

Equation (2) can be broken down into two parts:

$$\lambda x^I = AX^C$$

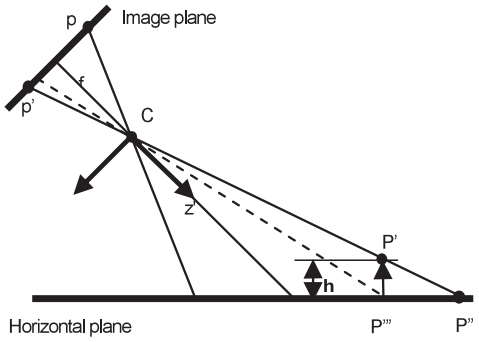


**Fig. 3** Perspective projection.

$$X^C = [RT]X^W \tag{5}$$

where $\boldsymbol{X^C}$ represents the world points expressed in the camera coordinate system, $\lambda$ is the scaling factor, and $\boldsymbol{x}^I$ is the image point in homogeneous coordinates, $(x, y, 1)$. The image points are inverse mapped into the camera coordinate system as shown in Eq. (6):

$$\begin{bmatrix} X^C \\ Y^C \\ Z^C \end{bmatrix} = \lambda A^{-1} \begin{bmatrix} x^I \\ y^I \\ 1 \end{bmatrix} \tag{6}$$

Equation (5) shows that $\lambda$ is the depth in the scene (in camera coordinates), $Z^C$. Along with Eq. (6), we can see that the inverse mapping only exists if the scaling factor, $\lambda$, is constrained, as is the case when all the world points are assumed to lie on a common plane.

From this stage, the mapping into a common reference system is easily implemented by applying rotations and translations, as shown in Eq. (7).

$$X^W = R^{-1}(X^C - T) \tag{7}$$

In our approach, we have mapped into a reference system corresponding to a bird's eye view of the scene. However, while the road surface points are on a common plane, the points corresponding to vehicles are not, as our method assumed. This introduces distortions as illustrated in **Fig. 3** and **Fig. 4**.

Figure 3 shows that the point P′, at a height, $h$, above the horizontal plane,

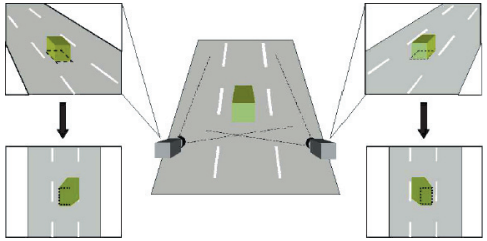

**Fig. 4** Inverse perspective mappings showing distortions due to off-plane image points.

and P″, which is on the horizontal plane, both project onto the image point p′. Therefore, under the inverse mapping, the world point P′ would in fact appear at P″ on the horizontal plane. This horizontal shift of P′ from its horizontal position P‴ to P″ is the error in the inverse mapping resulting from the off-plane distance $h$ of P′. These errors lead to distortions in the rotated and translated images used in estimating the vehicles on a road surface as shown in Fig. 4.

In the method presented in this paper, the inverse mapping and rotation translation is not implemented explicitly. Instead, the approach is to use perspective un-distortion, in which four features which are distorted by perspective are re-mapped to the corresponding features that are not subject to perspective distortion. This yields a set of equations of the form:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \lambda \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \tag{8}$$

where $(x, y, 1)^T$ are the perspective distorted, homogeneous image coordinates, $(x', y', w')^T$ are the undistorted, homogeneous image coordinates, and $\lambda$ is a scale factor, given by $a_{31}x + a_{32}y + a_{33}$.

This approach can be understood in terms of a composition of homographies. In Ref. 7), the projection of a road surface is described as a homography, of the form:

$$\lambda A(r_1^c \quad r_2^c \quad T) = (h_1^c \quad h_2^c \quad h_3^c) \tag{9}$$

where $r_i^c$ is the $i^{th}$ column of the rotation matrix, $T$ is the translation vector, and $h_i^c$ is the $i^{th}$ column of the homography. Denoting the homographies between the road plane and cameras 1 and 2, as $H_1$ and $H_2$ respectively, the projections of points on the road onto the individual images are given by $H_1\boldsymbol{X} = \boldsymbol{x}_1$ and $H_2\boldsymbol{X} = \boldsymbol{x}_2$ where $\boldsymbol{X}$ are the world points and $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are the image points of cameras 1 and 2 respectively. Therefore, any mapping of the form $M\boldsymbol{x}_1 = \boldsymbol{x}_2$ must also satisfy $M = H_2H_1^{-1}$. Because of the decomposition of the homography into the rotation, translation, and projection components shown in Eq. (9), the transformation $M$ can be written as:

$$M = \lambda' A_2(r_1^{2c} \quad r_2^{2c} \quad T^2)(r_1^{1c} \quad r_2^{1c} \quad T^1)^{-1}A_1^{-1} \tag{10}$$

We can also write:

$$(r_1 \quad r_2 \quad t) = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} 1 & 0 & \\ 0 & 1 & R^{-1}t \\ 0 & 0 & \end{pmatrix} \tag{11}$$

Therefore, Eq. (11) can be written as:

$$M = \lambda' A_2 R_2 \begin{pmatrix} 1 & 0 & \\ 0 & 1 & R_2^{-1}T^2 \\ 0 & 0 & \end{pmatrix} \begin{pmatrix} 1 & 0 & \\ 0 & 1 & R_1^{-1}T^1 \\ 0 & 0 & \end{pmatrix}^{-1} R_1^{-1}A_1^{-1} \tag{12}$$

The translation component of Eq. (11) is, within a multiplicative factor, a 2-D affine transformation, which in Eq. (11) and Eq. (12) represents a scaling and a translation (in the $x$, $y$ direction). Thus, $M$ can be decomposed into an inverse projection, rotation, translation, and re-projection. However, in our approach, the intrinsic and extrinsic parameters are not explicitly specified, eliminating the need for camera calibration.

The key aspect of this work is that four features lying within a common field of view of the multiple-camera system were selected and were re-projected onto four corresponding perspectively undistorted points representing the top-down view from a virtual camera. By solving for the system of equations relating these two sets of points, a warping, $M$, was computed, that can be shown to correspond to a composition of two homographies. The composition of the homographies can be represented as a sequence of inverse perspective mapping, inverse rotation, translation and rotation, and re-projection, although the calibration parameters are not known explicitly. Thus, each warping allowed for the projection of the

individual image information into a common reference frame without calibration between cameras. Because the virtual camera was user-defined, and all images were re-projected onto this single virtual camera, the registration error was reduced to the interpolation error arising from the rotation, translation, and re-projection of the original images.

The subsequent contribution of this work was to show that with the computed warping matrices, fusion of the camera images could compensate for the information missing due to factors such as occlusions.

The mechanism for the data fusion is described below.

(2) **Probability Maps**

The backgrounds are subtracted from the camera images to identify vehicle candidate regions. However, as image points representing vehicles are not on the common plane, i.e., the road surface, they will be incorrectly mapped, as shown in Fig. 3. Those points closest to the surface will be the least distorted, as can be seen with the bottom of the box, highlighted by dashed outline in Fig. 4.

However, in the absence of these markings, the inverse perspective-mapped vehicle candidate region includes pixels representing both the actual vehicle image points and distorted vehicle image points. Thus, the delineation of the vehicle is not clear. The degree of distortion depends on the camera orientation, the aperture angle, and the off-plane distance of the world point (discussed below).

Image pixels that lie outside the distorted vehicle candidate region have zero probability of representing a vehicle. The pixels within the distorted vehicle candidate region are assumed to have a fixed equal probability of representing a vehicle as of being the result of distortion. Denoting the probability of a pixel representing a point on the vehicle as $P(x, y)$ and the inverse mapped image as $I(x, y)$, the probability map is generated as follows:

$$P_i(x,y) = \begin{cases} 0 & \text{if } I'(x,y) = 0 \\ k & \text{if } I'(x,y) = 255 \end{cases} \tag{13}$$

where the subscript $i$ indicates that the probability map is generated for camera $i$.

However, given $N$ cameras of a scene, combining the individual probability maps, $P_i(x, y)$ will yield the probability fusion map, denoted as:

**Fig. 5** Each camera in the system generates its own probability map. Each color in the image on the right is a region having a non-zero probability of representing the vehicle.

$$P_F(x,y) = \sum_i P_i(x,y) \tag{14}$$

A further weighted average filter was discussed in Refs. 10) and 12) to be adaptive to different camera resolutions and distances to the common detection area. However, because this filter relied on a 'judgment factor', it introduced uncertainties. Therefore we considered it simpler, and more consistent, to simply assume a constant probability in the vehicle candidate region. Moreover, as distortions are dependent on the geometry between the target and the camera, they are not consistent between cameras, leading to smaller probabilities in the regions of the PFM containing contributions from distortions.

Because multiple views must be consistent in the common field of view, the vehicles will be located in the region where the value of $P_F(x, y)$ will be greatest.

The maximum value of $P_F(x, y)$ is given by:

$$P_F(x,y)_{\max} = Nk \tag{15}$$

where $N$ is the number of cameras in the system). **Figure 5** gives the PFM of a real scene.

Vehicle detection was implemented by a computing the threshold of the resulting PFM, $P_F(x, y)$. Because of the limited number of cameras (3), we chose to use the contribution of all cameras for the segmentation of the PFM. Thus, the threshold was set at $T = Nk - 1$.

(3) **Distortions**

Although the common features in the views must be consistent, the distortions in the images are dependent on the off-plane distance and the position of the feature in the field of view and the orientation of the camera (see Figs. 3 and 5 and

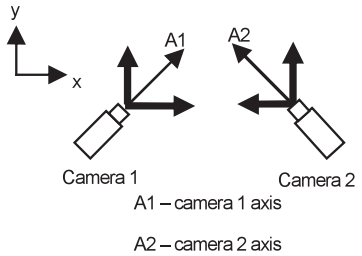


**Fig. 6** The two cameras showing the orientation of their axes and the components of the distortion.

Section 3.3). Thus, when the probability maps are merged to generate the PFM, the distortions, which are camera-dependent, will be suppressed to a certain extent in the PFM (see **Fig. 6**).

Because the camera orientations have components in both $x$ and $y$ directions, the distortion has components in both of these directions also. As Fig. 6 shows, in the case where the camera axes are convergent, the distortions along the $x$-axis tend to be in opposite directions, so that the regions containing distortions will not lead to a region of high probability for the vehicle presence. However, the distortions in the $y$-direction do not cancel out. Thus, the PFM is still subject to some distortion in the $y$-direction as shown in Fig. 6.

The white region in the PFM (see Fig. 5) is the area of maximum probability for the presence of a vehicle since it is the area where vehicle candidate regions match. However, due to distortions, the region of maximum probability has a tapered shape. However, it has been shown in Refs. 10) and 12) and the work presented here, that the detection is sufficiently accurate to allow the tracking needed for successful monitoring.

### 3.3 Synopsis of PFM Algorithm

Finally, before discussing the experimental results presented in this paper, a short summary of the overall algorithm is presented here. It can be broken down into the following steps:

(1) Model generation
(2) Background subtraction
(3) Shadow removal
(4) Image warping and threshold on each camera to create probability maps
(5) Morphological processing to suppress artefacts due to image noise
(6) Combine the probability maps

Although the first step is not an intrinsic part of the algorithm presented here, it is obvious that accurately eliminating points that are clearly not part of the vehicles improves the probability maps. Thus, having a good background model will improve the result of the background subtraction, increasing the probability of accurately detecting potential vehicles. Obviously, this will, in turn, increase the overall accuracy of the system.

The third step, shadow removal, significantly eliminates image points that while clearly not part of the vehicle, despite the background subtraction, are detected as being part of the vehicle. Their elimination yields significantly improved probability maps especially after the warping stage, where distortions are introduced. One of features of the approach using PFM that is very useful is that there are a number of pixels corresponding to points of the vehicle sufficiently close to the ground plane so as to be subject to limited distortion. These points are either the front or the rear of the vehicle, depending on the direction of travel. In the case of shadows, while the points are indeed on the ground plane, and are not therefore distorted in the inverse mapping, the accurate positioning of the vehicle boundary in the probability map is no longer clearly identifiable, so that there are no anchor points to constrain the result of the probability map fusion.

## 4. Experiments and Results

### 4.1 Simulation Tests

For testing purposes, a test environment was initially set up in the lab to simulate road traffic, albeit under controlled conditions. The test environment is shown in **Fig. 7** with the resulting PFM shown in **Fig. 8**. The detected vehicles are shown in **Fig. 9**. It consists of a scale model of a road with remote-control operated model vehicles. Using this environment, various traffic density situations could easily be set up to test the effectiveness of the algorithm in detecting and segmenting vehicles, even in the presence of occlusions.

The processing in the work presented here was implemented using a combination of commercial libraries (Matrox Imaging Library) and open source libraries

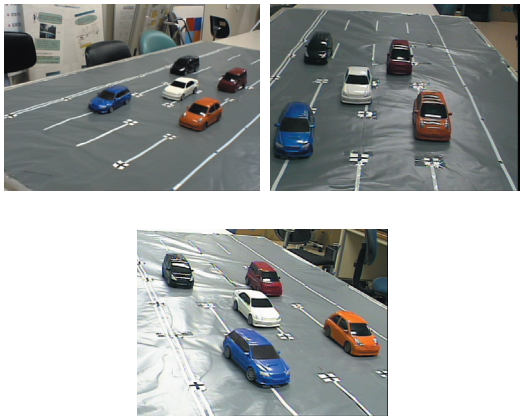

**Fig. 7** This is a simulation of traffic showing partial occlusion (bottom image, partial overlap between white red vehicles) and almost an occlusion (in the left image).

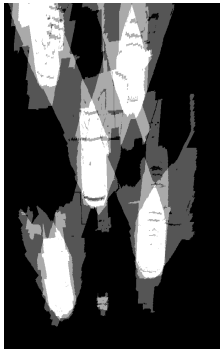

**Fig. 8** PFM of the vehicle arrangement in the test setup shown in Fig. 7.

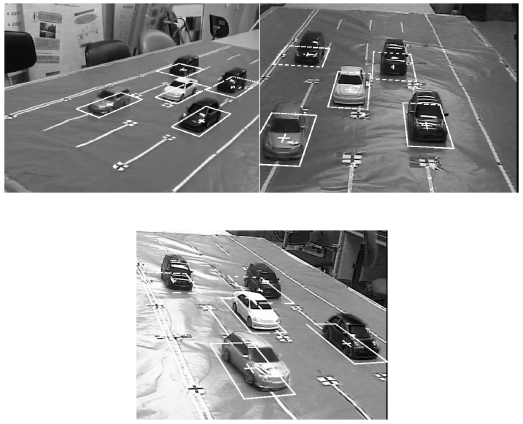

**Fig. 9** Detected vehicles projected onto the 3 views.

**Fig. 10** Real traffic scene in Shanghai, China.

(OpenCV). The results of the PFM obtained from this test setup are shown in Figs. 8 and 9.

It should be noted that in Fig. 7, although there is an occlusion in the left camera, and almost an occlusion in the right camera, the result of the PFM shows the vehicles clearly separated. See Fig. 8. Greater occlusions were detected in actual traffic scenes, as shown in Fig. 13.

### 4.2 Real Traffic Scene Tests

Subsequent to initial tests, actual traffic video was acquired in Shanghai to collect larger sampling statistics. A number of different tests were run on these images. In the first set of tests, a short video clip, 7.5 minutes in length was processed (**Fig. 10**).

In this test, no background model was generated, the background being

**Table 1** Traffic count for a PFM based solution with no robust background model.

| | |
|---|---|
| Count (moving features)-truth value | 445 |
| Number detected with PFM | 440 |
| Number of False Positives | 3 |
| Accuracy of Detected features (AD) | 99.3% |
| Number of False Negatives | 8 |
| Detection rate accuracy (DR) | 98.2 % |

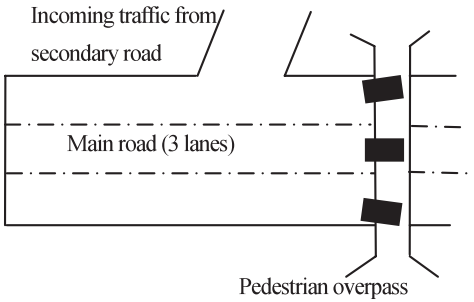


**Fig. 11** Configuration of the cameras and the region of observation (showing the road merging with the main road) used in the experiments. The three cameras used in the experiments are shown as the black boxes. These are oriented so as to have axes that are slightly convergent.

acquired from a single image with no vehicles present. Because of changing conditions over time, it does limit the time over which the processing can be carried out with the same background. The vehicle counts were sampled at 1 second intervals and the results are shown in **Table 1**.

The camera configuration is shown below in **Fig. 11**.

In terms of camera positioning, there is no constraint except for the need to have a common field of view between the cameras provided the homographies can be established. With regards to the number of cameras, it is a compromise between the increased information that can be obtained by having more cameras and the increased cost of the system, a significant consideration in real-world applications. Due to regulatory issues, our cameras had to be set up on existing structures, which meant we had to rely on overpasses. Thus, our cameras were

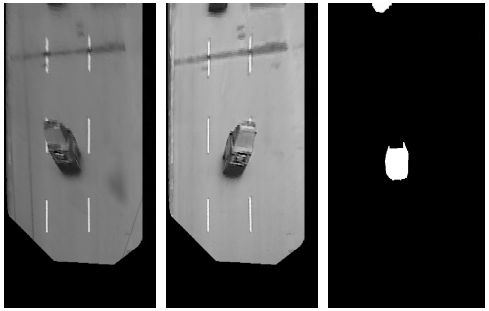

**Fig. 12** The PFM shows a vehicle in the upper, left of centre, part of the image. Although it is slightly visible in the right image, it is not visible in the left image, and would not be detected in the PFM. No shadow removal was used in the generation of this PFM.

laid out in a linear arrangement. In this case, going from 3 to 4 or more cameras was not judged to provide significantly more information.

The accuracy of the detected features, using the PFM, was computed as follows:

$$AD = (C - FP)/C \tag{16}$$

where $AD$ is the accuracy of the detection count, $FP$ is the number of false positives, and $C$ is the number of moving features counted. This measures the fraction of the detected moving features that are true moving features. The detection rate accuracy, $DR$, is a measure of the fraction of moving features detected by the PFM, and is computed as:

$$DR = (C - FP)/(C - FP + FN) \tag{17}$$

where $FN$ is the number of false negatives.

The truth value was established manually, by analysing all the images from each camera. A vehicle was deemed present if it was present in all the camera images as this provided confirmation, especially if only a small fraction of the vehicle was present. It is worth noting that there were two main sources of errors: shadows, and the incompleteness of the feature in the field of view of the PFM. Examples of these are shown below in **Fig. 12**.

It is also worth noting that in higher traffic density situations, vehicles were accurately detected as shown in **Fig. 13**.

Finally, the improvement in performance obtained by the use of a robust

**Fig. 13** Moderate to high volume traffic detection. Occlusion is seen in the left image. This occlusion is removed in the lower left image by the use of the IPM (Inverse Perspective Mapping) and the fusion of the images.

**Fig. 14** Background model obtained by AMF.

background model was examined. An adaptive median filter was used for the development of the background which is shown in **Fig. 14**.

The Shanghai images were processed over a one hour interval, which is sufficiently long for outdoor illumination to start showing variation. A shadow model was used to segment shadow area form the target region which is shown in **Fig. 15**. The results were shown in 10 minute intervals to examine the vehicle count over time intervals with minimal illumination changes. These were compared to a conventional single camera system and the results are shown in

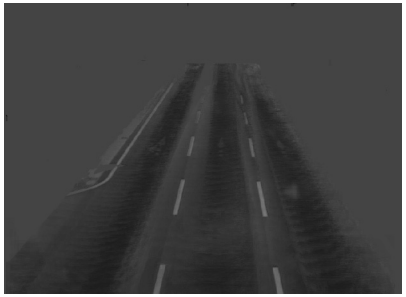

**Fig. 15** Shadow model.

**Table 2** Traffic count comparisons of a PFM solution and a conventional single camera based solution.

| Time (min) | Truth count | PFM solution | | Conventional solution | |
|---|---|---|---|---|---|
| | | Cnt | % | Cnt | % |
| 0-10 | 215 | 213 | 99.5% | 231 | 92.7% |
| 10-20 | 182 | 179 | 98.3% | 191 | 89.5% |
| 20-30 | 151 | 151 | 100% | 159 | 94.4% |
| 30-40 | 158 | 155 | 98.1% | 174 | 90% |
| 40-50 | 214 | 213 | 99.5% | 239 | 88.6% |
| 50-60 | 154 | 155 | 99.2% | 168 | 90.2% |
| Total | 1074 | 1066 | 99.3% | 1162 | 91.8% |

**Table 2**.

The conventional single camera system used background subtraction, shadow removal and threshold for vehicle detection. A counter-intuitive observation was made during the analysis of the results from the single camera system. It was observed that there were more vehicles detected with the conventional system than with the multi-camera PFM algorithm. This was mainly due to over-segmentation errors and illumination artefacts present in the single camera system.

In addition, the system was examined in terms of speed measurement accuracy with respect to the ground truth established by a single camera system. Results of this comparison are shown in **Table 3**. For the speed measurements, lane

**Table 3** Comparison of speed measurements of the PFM based solution with the ground truth.

| No. | Truth (km/h) | PFM (km/h) | Accuracy (%) |
|---|---|---|---|
| 1 | 41 | 45.9 | 88.0 |
| 2 | 64 | 65.5 | 97.7 |
| 3 | 54 | 59.7 | 89.9 |
| 4 | 64 | 62.1 | 97.2 |
| 5 | 36 | 38.1 | 94.2 |
| 6 | 64 | 59.0 | 92.2 |
| 7 | 54 | 55.5 | 97.2 |

markings, being of known dimensions and positioning, were used as reference marks. Because the points of the vehicles closest to the road are subject to the least distortion, especially in the PFM, the vehicle positions were determined as the position of the rear of the vehicle (since vehicles were moving away from the cameras). Time information was implicit in the frame number, consecutive frames being $1/30^{\text{th}}$s apart. Because the displacement over $1/30^{\text{th}}$ of a second, even for a vehicle travelling at 120 km/hr (a speed significantly greater than the speeds of the vehicles in our experiments), is on the order of 1 m, corresponding to a few pixels, tracking was simply implemented by establishing the proximity between the new positions and the previous positions.

The vehicle count results shown in Table 2 indicate that the PFM based approach to traffic monitoring is quite successful. This conclusion is further supported by the results of the speed measurements shown in Table 3.

Some vehicle extraction results obtained by the PFM solution are illustrated in **Fig. 16**.

Figure 16 shows that even in the presence of occlusions, the vehicles are differentiated in the PFM. However, Fig. 16 also shows rather clearly the general tapered shape of the high-probability region of the PFM. This has not posed a significant problem, although at higher traffic densities, it has the potential to become so. These tails, which are largely the result of the distortions due to the off-plane distance of the vehicle image points, are being examined and work is currently on-going to address this issue.

**Fig. 16** Results obtained with the PFM approach.

## 5. Conclusion

In this paper, a new approach to combining multi-camera information for the effective detection of vehicles has been presented. In this solution, the various images are re-projected into a common framework by perspectively undistorting the individual images. We have shown that this method implicitly uses the intrinsic and extrinsic parameters although no calibration is explicitly required. In various tests, it has been shown to achieve a high accuracy in vehicle count and speed measurements.

Currently we are pursuing further work in ancillary areas, such as shadow removal, in order to improve the results of the background subtraction. As background subtraction is a significant pre-processing step in the PFM solution, the PFM is potentially subject to issues that affect background subtraction, although

in many cases, it is less of an issue than it is for a single camera solution.

Furthermore, for practical applications, the algorithm will need to have real-time execution speed, that is, close to 30 frames per second (for NTSC standard video) or 25 frames per second (for PAL standard video). There is ongoing work in the optimization of the algorithm, especially with the advent of multi-core processor systems. This algorithm is also currently being ported to a DSP having an SIMD architecture to take advantage of the parallelism of DSP to improve its performance.

In addition, work is being undertaken to offset the errors introduced by the assumption that the image points represent a planar feature. Such developments should enhance the proposed system by rendering positional accuracy and shape extraction, especially in the third dimension (height), more robust, thus allowing the system to be used for vehicle classification.

**Acknowledgments** This work was partially supported by the MEXT Grant under the grant number of 18700184.

## References

1) Coifman, C., Beymer, D., McLauchlan, P., et al.: A real-time computer vision system for vehicle tracking and traffic surveillance, *Transport Research, Part C* (1998).
2) Beymer, D., McLauchlan, P., Coifman, B., et al.: A real-time computer vision system for measuring traffic parameters, *CVPR'97*, pp.495 (1997).
3) Sullivan, G., Baker, K., Worrall, A., et al.: Model based vehicle detection and classification using orthographic approximations, *British Machine Vision Conference* (1996).
4) Kilger, M.: A shadow handler in a video-based real-time traffic monitoring system, *IEEE Workshop on Applications of Computer Vision*, pp.1060–1066, Palm Springs, CA. (1992).
5) Koller, D., Weber, J. and Malik, J.: Robust multiple car tracking with occlusion reasoning, *ECCV*, pp.189–196, Stockholm, Sweden, May 2–6 (1994).
6) Koller, D., Weber, J., Huang, T., et al.: Towards Robust traffic scene analysis in real-time, *ICPR*, Israel (Nov. 1994).
7) Douret, J. and Benosman, R.: A volumetric multi-cameras method dedicated to road traffic monitoring, *IEEE IV*, Parma, Italy, June 14–17 (2004).
8) Ferryman, J.M., Worrall, A. and Maybank, S.: Learning 3D Object-Centred Appearance Models for Tracking, *Proc. IEEE Workshop on the Integration of Appearance and Geometric Methods in Object Recognition* (*WIAGMOR*) (in conjunction with *CVPR'99*), Fort Collins, Colorado, USA, pp.34–43, June 26 (1999).
9) Ferryman, J.M., Maybank, S. and Worrall, A.: Visual Surveillance for Moving Vehicles, *International Journal of Computer Vision*, Vol.37, No.2, pp.187–197 (June 2000).
10) Ferryman, J.M., Worrall, A. and Maybank, S.: Visual Surveillance for Moving Vehicles with Multiple Cameras, Orphanoudakis, S., Trahanias, P., Crowley, J. and Katevas, N. (Eds.), *Computer Vision and Mobile Robotics Workshop*, *CVMR'98*, 17–18 September 1998, Santorini, Greece, pp.147–154 (1998).
11) Lamosa, F., Uchimura, K. and Hu, Z.: Vehicle Detection using Probability Fusion Maps Generated by a Multi-Camera System, *Proc. IWAIT 2007*.
12) Hu, Z., Wang, C. and Uchimura, K.: 3D Vehicle Extraction and Tracking from Multiple Viewpoints for Traffic Monitoring by using Probability Fusion Map, *IEEE Intelligent Transportation Systems Conference* (*ITSC 2007*), pp.30–35 (2007).

(Received March 31, 2008)
(Accepted October 7, 2008)
(Released January 7, 2009)

**Francisco Lamosa** is a native of Montreal, Canada, born in 1969, who is currently a doctoral candidate at Kumamoto University since October 2004. He holds a B.Sc. (physics) from Concordia University in Montreal where he graduated in 1994. Upon graduation, he took courses in image processing at McGill University. Since 1997 he has worked as a software engineer in computer vision related systems first in Canada and then in the United Kingdom prior to moving to Japan to pursue his doctoral studies. During this time he has worked on traffic monitoring software systems and quality control inspection systems in a number of different industries. His current research interests are multi-camera and multi-sensor data fusion and multiple view geometry.

**Zhencheng Hu** received his B.Eng. degree from Shanghai Jiao Tong University, China in 1992, and his M.Eng. degree from Kumamoto University, Japan, in 1998. He received his Ph.D. degree in System Science from Kumamoto University, Japan, in 2001. Dr. Hu has held various positions in computer science and machine vision industry. He is currently an associate professor with the Department of Computer Science, Kumamoto University, Japan. His research interests include camera motion analysis, augmented reality, machine vision applications in industry and ITS. Dr. Hu is a member of IEEE, and the Institute of Electronics and Information Communication Engineers of Japan (IEICE).

**Keiichi Uchimura** received the B.Eng. and M.Eng. degrees from Kumamoto University, Kumamoto, Japan, in 1975 and 1977, respectively, and the Ph.D. degree from Tohoku University, Miyagi, Japan, in 1987. He is currently a Professor with the Department of Computer Science, Kumarnoto University. He is engaged in research on intelligent transportation systems, and computer vision. From 1992 to 1993, he was a Visiting Researcher at McMaster University, Hamilton, ON, Canada. Dr. Uchimura is a Member of the Institute of Electronics and Information Communication Engineers of Japan.