Appeared in First ACM International Workshop on Viedo Surveillance, November 7, 2003

Motion Analysis of Omni-Directional Video Streams for a Mobile Sentry

Tarak Gandhi
Computer Vision and Robotics Research Laboratory
University of California – San Diego
tgandhi@ucsd.edu

Mohan M. Trivedi
Computer Vision and Robotics Research Laboratory
University of California – San Diego
mtrivedi@ucsd.edu

ABSTRACT

A mobile platform mounted with Omni-Directional Vision Sensor (ODVS) can be used to monitor large areas and detect interesting events such as independently moving persons and vehicles. To avoid false alarms due to extraneous features, the image motion induced by the moving platform should be compensated. This paper describes a formulation of parametric ego-motion compensation for an ODVS. Omni images give 360 degrees view of surroundings but undergo considerable image distortion. To account for these distortions, the parametric planar motion model is integrated with the transformations into omni image space. Prior knowledge of approximate camera calibration and vehicle speed are integrated with the estimation process using Bayesian approach. Iterative, coarse to fine, gradient based estimation is used to correct the motion parameters for vibrations and other inaccuracies in prior knowledge. Experiments with camera mounted on a mobile platform demonstrate successful detection of moving persons and vehicles.

Categories and Subject Descriptors

I.4.8 [**Computer Vision**]: Scene analysis—*Motion*

General Terms

Algorithms

Keywords

Motion detection, Optical flow, Panoramic vision, Dynamic vision, Mobile robots, Intruder detection, Surveillance

1. INTRODUCTION AND MOTIVATION

Computer vision researchers have for long recognized the importance of visual surveillance related applications while pursuing some of the outstanding research issues in dynamic scene analysis, motion detection, feature extraction, pattern and activity analysis and biometric systems. Recent world events are demanding practical and robust deployment of video based solutions for a wide range of applications [9, 24, 27]. Such wider acceptance of the need of the technology does not mean that these systems are indeed ready for deployment. There are many important and difficult research problems that are still outstanding. In this paper we present a research study focused on one of such challenging research problem, that of developing an autonomous system which can serve as a "Mobile Sentry" to perform the tasks that a person posted on a guard duty around a perimeter of a base gets assigned. The mobile sentry with video cameras should be able to detect "interesting" events, record and report the nature and location of the event in real-time for further processing by a human operator. This is indeed an ambitious goal and it requires resolution of several important problems from computer vision and intelligent robotics area. In this paper we are focused on the problem of how to detect and compensate for the ego motion of the mobile platform. One of the novel feature of our research is the omnidirectional video streams we use as the input to the vision system.

When the camera is stationary, background subtraction is often used to extract moving objects [15, 32]. However, when the camera is moving, the background also undergoes ego-motion, which should be compensated. To distinguish objects of interest from extraneous features on the ground, the ground is usually approximated by a planar surface, whose ego-motion is modelled using a projective transform [10, 23] or its linearized version. Using this model, the ego-motion of the ground can be compensated in order to separate the objects with independent motion or height. This approach has been widely used for object detection from moving platforms [5, 11, 23].

Omni-Directional Vision Sensors (ODVS) or omni-cameras which give a 360 degree field of view of the surroundings are very useful for applications in surveillance and robot navigation [3]. Motion estimation from moving ODVS cameras has recently been a topic of great interest. Rectilinear cameras usually have a smaller field of view, due to which the focus of expansion often lies outside the image, causing motion estimation to be sensitive to the camera orientation. Also, the motion field produced by translation along horizontal direction is similar to that due to rotation about vertical axis. As noted by Gluckman and Nayar [13], ODVS cameras avoid both these problems due to their wide field of view. They project the image motion on a spherical surface using Jacobians of transformations to determine ego-motion

IWVS'03, November 7, 2003, Berkeley, California, USA.

of a moving platform terms of translation and rotation of the camera. Vassalo et al. [34] use the spherical projection to determine ego-motion of a moving platform terms of translation and rotation of the camera. Experiments are performed using robotic platforms in indoor environment and the ego-motion estimates are compared with those from odometry. Shakernia et al. [29] use the concept of back-projection flow, where the image motion is projected to a virtual curved surface in place of spherical surface and make the Jacobians of transformation simpler. Using this concept, they have adapted ego-motion algorithms designed to planar surface for use with ODVS sensors. Results using simulated image sequences show the basic feasibility of the approach.

In our own research, the emphasis is on robustness, efficiency, and applicability in outdoor environments encountered in surveillance and physical or base security. The main contribution of this paper is to perform detection of events such as independently moving persons and automobiles from ODVS image sequences, and apply it to video sequences obtained from a moving platform for surveillance applications.

2. EGO-MOTION COMPENSATION FRAMEWORK FOR ODVS VIDEO

Parametric motion estimation based on image gradients, also known as the "direct method" has been used for rectilinear cameras in for planar motion estimation, obstacle detection and motion segmentation [19, 22]. The advantage of the direct methods is that they can use motion information not only from corner-like features, but also edges, which are usually more numerous in an image. Here, the concept of direct method is extended for use with ODVS. This approach was also used for detecting surrounding vehicles from a moving car in [17]. An ODVS gives full 360 degree view of the surroundings, which reduces the motion ambiguities often present in the rectilinear cameras. However, the images undergo considerable distortion which should be accounted for during motion estimation.

The block diagram of the event detection system is shown in Figure 1. The initial estimates of the the ground plane motion parameters are obtained using the approximate knowledge about the camera calibration and speed. Using these parameters, one of the frames is warped towards another to compensate the motion of the ground plane. However, the motion of features having independent motion or height above the ground plane is not fully compensated. To detect these features, the normalized image difference between the two images is computed using temporal and spatial gradients. This suppresses the features on ground plane and enhances the interesting objects. Morphological and other post-processing is performed to further suppress the ground features due to any residual motion and get the positions of the objects. The detected objects are then tracked over frames to form events.

However, the calibration and speed of the camera may not be known accurately. Furthermore, the camera could vibrate during the motion. Due to this, the motion of the ground plane may not be fully compensated, leading to misses and false alarms. In order to improve the performance, motion parameters are iteratively corrected using the spatial and temporal gradients of the motion compensated images using optical flow constraint in coarse to fine framework. The motion information contained in these gradients is optimally combined with the prior knowledge of motion parameters using a Bayesian framework similar to [23]. Robust estimation is used to separate the ground plane features from other features. The following sections deal with the individual blocks described above, along with the appropriate formulation for ODVS.

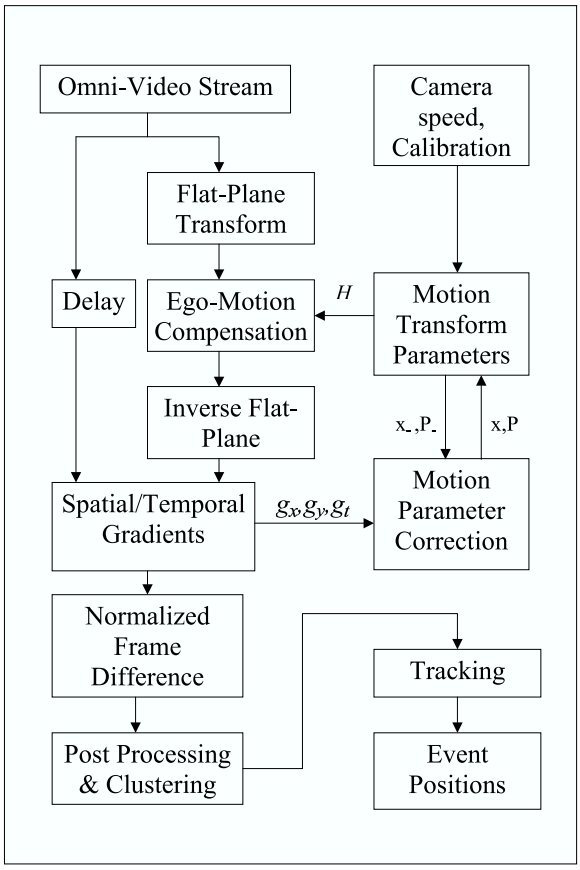


Figure 1: Block diagram for event detection and recording system based on ego-motion compensation from a moving platform.

3. ODVS MOTION TRANSFORMATIONS

To compensate the motion of the omni-directional camera, the transformation due to ODVS should be combined with that due to motion. These transforms are discussed below:

3.1 Flat plane transformation

The ODVS used in this work consists of a hyperbolic mirror and a camera placed on its axis. It belongs to a class of cameras known as central panoramic catadioptric cameras [3]. These cameras have a single viewpoint that permits the image to be suitably transformed to obtain perspective views. Figure 2 (a) shows a photograph of an ODVS mirror. An image from a camera mounted with an ODVS mirror is shown in Figure 2 (b). It is seen that the camera covers a 360 degrees field of view around its center. However, the image it produces is distorted with straight lines transformed into curves. A flat plane transformation is applied to the image to produce a perspective view looking downwards as

(a)

(b)

(c)

Figure 2: (a) A typical image from the ODVS. (b) Transformation to a perspective plan view.

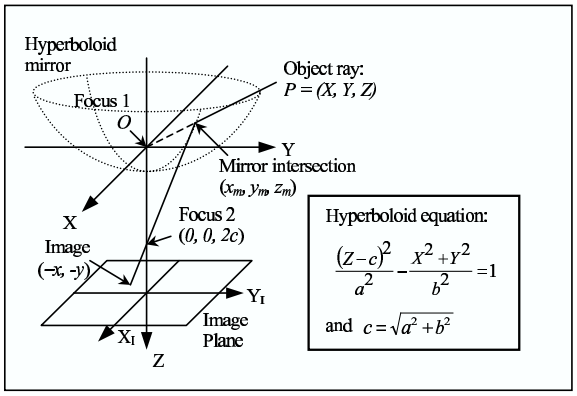


Figure 3: Omni-directional camera geometry

shown in Figure 2 (c), where the distortion is considerably reduced. Details of this transformation are discussed below.

The geometry of a hyperbolic ODVS is shown in Figure 3. According to the mirror geometry, a the light ray from the object towards the viewpoint at the first focus O is reflected so that it passes through the second focus, where a conventional rectilinear camera is placed. The equation of the hyperboloid is given by:

$$\frac{(Z-c)^2}{a^2} - \frac{X^2+Y^2}{b^2} = 1 \tag{1}$$

where $c = \sqrt{a^2+b^2}$.

Let $P = (X, Y, Z)^t$ denote the homogenous coordinates of the perspective transform of any 3-D point λP on ray OP, where λ is the scale factor depending on the distance of the 3-D point from the origin. It can be shown [1, 18, 29] that the reflection in mirror gives the point $-p = (-x, -y)^t$ on the image plane of the camera using the flat-plane transform F:

$$F(P) = p = \begin{pmatrix} x \\ y \end{pmatrix} = \frac{q_1}{q_2 Z + q_3 \|P\|} \begin{pmatrix} X \\ Y \end{pmatrix} \tag{2}$$

where

$$q_1 = c^2 - a^2,\ q_2 = c^2 + a^2,\ q_3 = 2ac \tag{3}$$

$$\|P\| = \sqrt{X^2+Y^2+Z^2} \tag{4}$$

Note that the expression for image coordinates p is independent of the scale factor λ. The pixel coordinates $w = (u, v)^t$ are then obtained by using the calibration matrix K of the conventional camera composed of the focal lengths f_u, f_v, optical center coordinates $(u_0, v_0)^t$, and camera skew s.

$$\begin{pmatrix} w \\ 1 \end{pmatrix} = K \begin{pmatrix} p \\ 1 \end{pmatrix} \tag{5}$$

or

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_u & s & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{6}$$

This transform can be used to warp an omni image to a plan perspective view. To convert a perspective view back to omni view, the inverse flat-plane transform p can be used:

$$\begin{pmatrix} p \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = K^{-1} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \tag{7}$$

$$F^{-1}(p) = P = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} q_1 x \\ q_1 y \\ q_2 - q_3 \sqrt{x^2+y^2+1} \end{pmatrix} \tag{8}$$

It should be noted that the transformation of omni to perspective view involves very different magnifications in different parts of the image. Due to this, the quality of the image deteriorates if the entire image is transformed at a time. Hence, it is desirable to perform motion estimation directly in the ODVS domain, but use the above transformations to map the locations to the perspective domain as required.

3.2 Planar motion transformation

To detect objects with motion or height, the motion of the ground is modeled using planar motion model [10, 21]. Let P_a and P_b denote the perspective transforms of a point on the ground plane in the homogenous coordinate systems corresponding to two positions A and B of the moving camera. These are related by:

$$\lambda_b P_b = \lambda_a R P_a + D = \lambda_a \left[R P_a + D/\lambda_a \right] \tag{9}$$

where R and D denote the rotation and translation between the camera positions, and λ_a, λ_b depend on the distance of the actual 3-D point. Let the ground plane satisfy the following equation at the camera position A:

$$\lambda_a K^t P_a = 1 \tag{10}$$

or

$$1/\lambda_a = K^t P_a \tag{11}$$

Substituting the value of $1/\lambda_a$ from equation (11) in equation (9), it is seen that P_a and P_b are related by a projective transform:

$$\lambda_b P_b = \lambda_a \left[R + D K^t \right] P_a = \lambda_a H P_a \tag{12}$$

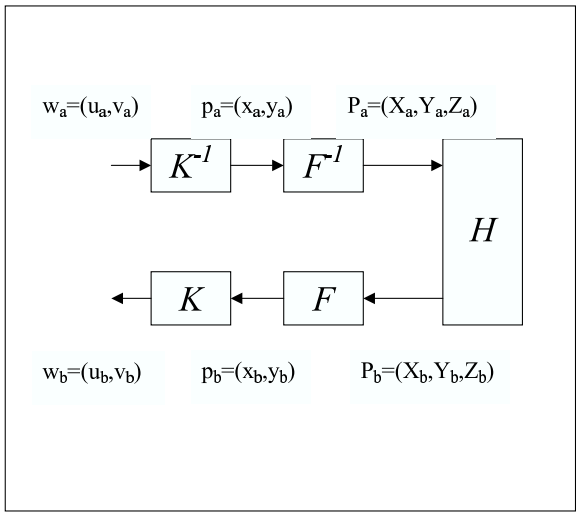


Figure 4: Transforming a pixel from omni image A to omni image B using (1) Inverse calibration matrix K^{-1}, (2) Inverse flat plane transform F^{-1} (3) Projective transform H for planar motion from A to B (4) Flat plane transform F (5) Calibration matrix K

or $P_b \equiv HP_a$ within a scale factor. This relation has been widely used to estimate planar motion for perspective cameras.

For performing motion compensation using omni-directional cameras, the above projective transform should be combined with the flat-plane transform as well as camera calibration matrix to warp every point in one image towards another. The complete transform for warping is shown in Figure 4.

4. PARAMETRIC MOTION ESTIMATION FOR ODVS

This section describes the main contribution of the paper. Direct methods based on image gradients have been applied for estimating the motion parameters for rectilinear cameras [4, 19]. Here, the direct method is generalized for ODVS cameras. Information from image gradients is combined with the a-priori known information about the camera motion and calibration in Bayesian framework to obtain optimal estimates of motion parameters for ego-motion compensation.

4.1 Use of optical flow constraint

Under favorable conditions, the spatial gradients (g_x, g_y), the temporal gradient g_t, and the residual image motion $(\Delta u, \Delta v)^t$ after current motion compensation satisfy the optical flow constraint [16].

$$g_x \Delta u + g_y \Delta v + g_t = 0 \tag{13}$$

However, there is only one equation between two unknowns for each point. Due to this, only the normal flow - i.e., flow in the direction of the gradient - can be determined using a single point. This is known as the aperture problem, illustrated in Figure 5 (a). To solve this problem, Lucas and Kanade [25] assumed that the image motion is approximately constant in a small window around every point. Using this constraint, more equations are obtained using the neighboring points, and the full optical flow can be estimated using least squares. Such an estimate is reliable near corner-like points where window has gradients in different directions. This is as seen in Figure 5 (b). This method has been used by Kanade, Lucas and Tomasi [30] to find and track corner-like features over an image. However, in case of ODVS the assumption of uniform optical flow needs to be modified due to the non-linear ODVS transform. Danilidis [7] has generalized the optical flow estimation to ODVS camera.

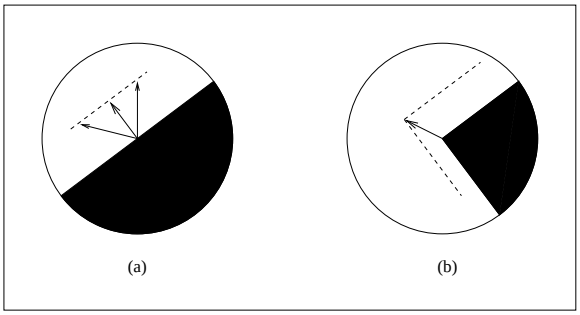


Figure 5: Aperture problem. (a) In the case of an edge, only the component of motion normal to the edge can be determined. (b) In the case of a corner, the aperture problem is avoided, and the motion can be uniquely determined.

However, this approach would use the motion information only at corner-like features. However, the edge features also have motion information. To use this information, the image gradients can be used directly to estimate the model parameters. This approach is known as direct method of motion estimation in literature and has been extensively used in obstacle detection using rectilinear cameras [4, 19]. Usually, a linearized version of projective transform is used:

$$\begin{aligned} \Delta u &= a_1 x + a_2 y + a_3 + a_7 x^2 + a_8 xy \\ \Delta v &= a_4 x + a_5 y + a_6 + a_7 xy + a_8 y^2 \end{aligned} \tag{14}$$

The expressions of image motion are substituted into the optical flow constraint in equation (13) to give:

$$g_x(a_1 x + a_2 y + a_3 + \ldots) + g_y(a_4 x + a_5 y + a_6 + \ldots) + g_t = 0 \tag{15}$$

This gives one equation for every point in 8 parameters, that can be solved using linear least squares. Since the quadratic parameters are more sensitive to noise, a 6-parameter affine model is also used.

4.2 Generalization for ODVS

To apply the motion estimation to ODVS camera, the non-linear flat-plane transform is used to go from omni to perspective domain and back. Since non-linearity has to be dealt with anyway, the projective transform H is used instead of a linear model so that large motions can be handled better. The motion parameters in the projective transform are parameterized as:

$$\mathbf{x} = \begin{pmatrix} h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 \end{pmatrix}^t \tag{16}$$

with

$$\frac{H}{H_{33}} = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & 1 \end{pmatrix} \tag{17}$$

The optical flow constraint equation is satisfied only for small image displacements up to 1 or 2 pixels. To estimate larger motions, a coarse to fine pyramidal framework [20, 31] is used. In this framework, a multi-resolution Gaussian pyramid is constructed for adjacent images in the sequence. The motion parameters are first computed at the coarsest level, and the image points at the next finer level are warped using the computed motion parameters. The residual motion is computed at the finer level, and the process is repeated until the finest level. Even within each level, multiple iterations of warping and estimation can be performed.

Let $\bar{\mathbf{x}}$ be the actual value of the motion parameter vector, and $\hat{\mathbf{x}}$ be the current estimate. Using the current estimate, the second image is warped towards the first image A to get the warped image B. Then, the transformation between A and B can be expressed approximately in terms of $\mathbf{\Delta x} = \bar{\mathbf{x}} - \hat{\mathbf{x}}$. Let $w_a = (u_a, v_a)^t$ be the projection of a point on the planar surface in image A. Then, the projection w_b of the same point in warped image B is a function of w_a as well as Δx, given using a composition of operations shown in Figure 4. The optical flow constraint between images A and B is then given by:

$$\begin{pmatrix} g_x & g_y \end{pmatrix} [w_b - w_a] = -g_t + \eta \tag{18}$$

where η accounts for the random noise in the temporal image gradient. For N points on the planar surface, the constraints can be expressed in a matrix form:

$$\mathbf{\Delta z} = \mathbf{c}(\mathbf{\Delta x}) + \mathbf{v} \tag{19}$$

where every row i of the equation represents the constraint for a single image point with

$$\begin{aligned} \mathbf{c}_i(\mathbf{\Delta x}) &= \begin{pmatrix} g_x & g_y \end{pmatrix}_i [w_b(w_a; \Delta x) - w_a] \\ \mathbf{\Delta z}_i &= -(g_t)_i \,,\ \mathbf{v}_i = \eta_i \end{aligned} \tag{20}$$

Due to the flat plane and the projective transforms, the function $\mathbf{c}(\cdot)$ is a non-linear. Hence, state estimate $\hat{\mathbf{x}}$ and its covariance $\mathbf{P}$ are iteratively updated using the measurement update equations of the iterated extended Kalman filter [2], with $\mathbf{C}$ denoting the Jacobian matrix of $\mathbf{c}(\cdot)$.

$$\mathbf{P} \leftarrow \left[\gamma \mathbf{C}^t \mathbf{R}^{-1} \mathbf{C} + \mathbf{P}_-^{-1}\right]^{-1} \tag{21}$$

$$\hat{\mathbf{x}} \leftarrow \hat{\mathbf{x}} + \Delta\hat{\mathbf{x}} = \hat{\mathbf{x}} + \mathbf{P}\left[\gamma \mathbf{C}^t \mathbf{R}^{-1} \mathbf{\Delta z} - \mathbf{P}_-^{-1}(\hat{\mathbf{x}} - \mathbf{x}_-)\right] \tag{22}$$

where $\mathbf{R}$ is the covariance of the temporal gradient measurements, $\mathbf{x}_-$ is the prior value of state obtained from camera calibration and velocity, and $\mathbf{P}_-$ is the prior covariance. The matrix $\mathbf{R}$ is taken as a diagonal matrix to simplify calculations. However, this would mean assuming that the pixel gradients are independent, which may not really be the case since gradients are computed from multiple pixels. Hence, the factor $\gamma \leq 1$ is used to accommodate the interdependence of the gradient measurements.

To compute the Jacobian $\mathbf{C}$, each row $\mathbf{C}_i$ is expressed using chain rule:

$$\mathbf{C}_i = \left(\frac{\partial \mathbf{c}_i}{\partial \mathbf{x}}\right) = \left(\frac{\partial \mathbf{c}}{\partial w_b}\frac{\partial w_b}{\partial p_b}\frac{\partial p_b}{\partial P_b}\frac{\partial P_b}{\partial \mathbf{x}}\right)_i \tag{23}$$

where $P_b = (X_b, Y_b, Z_b)^t$, $p_b = (x_b, y_b)^t$ and $w_b = (u_b, v_b)^t$ are the coordinates of point i in in the mirror, image, and pixel coordinate systems for camera position B.

Differentiating equation (20) w.r.t. w_b gives:

$$\left(\frac{\partial \mathbf{c}}{\partial w_b}\right)_i = \begin{pmatrix} g_x & g_y \end{pmatrix}_i \tag{24}$$

The calibration equation (6) can be differentiated to obtain:

$$\left(\frac{\partial w_b}{\partial p_b}\right)_i = \begin{pmatrix} f_x & s \\ 0 & f_y \end{pmatrix} \tag{25}$$

The Jacobian of the flat plane transform is obtained by differentiating equation (2) at $P = P_b$ as:

$$\begin{aligned} \left(\frac{\partial p_b}{\partial P_b}\right)_i &= \begin{pmatrix} \frac{\partial x_b}{\partial X_b} & \frac{\partial x_b}{\partial Y_b} & \frac{\partial x_b}{\partial Z_b} \\ \frac{\partial y_b}{\partial X_b} & \frac{\partial y_b}{\partial Y_b} & \frac{\partial y_b}{\partial Z_b} \end{pmatrix} \\ &= \frac{1}{(q_2 Z_b + q_3 \|P_b\|)_i \, \|P_b\|_i} \\ &\quad \cdot \begin{pmatrix} q_3 x_b X_b - q_1 \|P_b\| & q_3 x_b Y_b & q_3 x_b Z_b \\ q_3 y_b X_b & q_3 y_b Y_b - q_1 \|P_b\| & q_3 y_b Z_b \end{pmatrix}_i \end{aligned} \tag{26}$$

Since the ODVS transforms giving p_a and p_b do not change if the homogenous coordinates P_a and P_b are changed by scale factor, we can scale the right hand side of equation (12) to give:

$$P_b = \frac{1}{H_{33}} H P_a = \begin{pmatrix} h_1 X_b + h_2 Y_b + h_3 Z_b \\ h_4 X_b + h_5 Y_b + h_6 Z_b \\ h_7 X_b + h_8 Y_b + Z_b \end{pmatrix} \tag{27}$$

Taking the Jacobian w.r.t. $\mathbf{x} = (h_1 \ldots h_8)$ gives:

$$\left(\frac{\partial P_b}{\partial \mathbf{x}}\right)_i = \begin{pmatrix} X_b & Y_b & Z_b & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & X_b & Y_b & Z_b & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & X_b & Y_b \end{pmatrix}_i \tag{28}$$

4.3 Outlier removal

The estimate given above is optimal only when all points really belong to the planar surface, and the underlying noise distributions are Gaussian. However, the estimation is highly sensitive to the presence of outliers, i.e. points not satisfying the ground motion model. These features should be separated using a robust method. To reduce the number of outliers, the region of interest of road is determined using calibration information, and the processing is done only in that region to avoid extraneous features. To detect outliers an approach similar to the data snooping approach discussed in [8] has been adapted for Bayesian estimation. In this approach, the error residual of each feature is compared with the expected residual covariance at every iteration, and the features are reclassified as inliers or outliers.

If a point $\mathbf{z}_i$ is not included in the estimation of $\hat{\mathbf{x}}$ – i.e. is currently classified as an outlier – then the covariance of its residual is:

$$\mathrm{Cov}\left[\mathbf{\Delta z}_i - \mathbf{C}_i \Delta\hat{x}\right] \simeq \mathbf{R} + \mathbf{C}_i \mathbf{P} \mathbf{C}_i^t \tag{29}$$

However, if $\mathbf{z}_{in}$ is included in the estimation of $\hat{\mathbf{x}}$ – i.e. is currently classified as an inlier – then it can be shown that the covariance of its residual is given by:

$$\mathrm{Cov}\left[\mathbf{\Delta z}_i - \mathbf{C}_i \Delta\hat{\mathbf{x}}\right] \simeq \mathbf{R} - \mathbf{C}_i \mathbf{P} \mathbf{C}_i^t < \mathbf{R} \tag{30}$$

Hence, to classify in the next iteration, the Mahalanobis norm of the residual is compared with a threshold τ. For

point currently classified as inlier the following condition is used:

$$[\mathbf{\Delta z}_i - \mathbf{C}_i\Delta\hat{\mathbf{x}}]\left[\mathbf{R} + \mathbf{C}_i\mathbf{P}\mathbf{C}_i^t\right]^{-1}[\mathbf{\Delta z}_i - \mathbf{C}_i\Delta\hat{\mathbf{x}}] < \tau \quad (31)$$

For point currently classified as inlier the covariance $\mathbf{R}$ is used in practice instead of $\mathbf{R} - \mathbf{C}_i\mathbf{P}\mathbf{C}_i^t$ in order to avoid non-positive definite covariance because of approximations due to non-linearities. This would somewhat increase the probability of classifying as an outlier instead of inlier, which is to be on safer side.

$$[\mathbf{\Delta z}_i - \mathbf{C}_i\Delta\hat{\mathbf{x}}]\,\mathbf{R}^{-1}[\mathbf{\Delta z}_i - \mathbf{C}_i\Delta\hat{\mathbf{x}}] < \tau \quad (32)$$

Note that this method is effective only when there is some prior knowledge about the motion parameters, otherwise the prior covariance $\mathbf{P}_-$ would become infinite. If there is no prior knowledge, robust estimators can be used as in [26].

4.4 Motion parameter estimation algorithm

The algorithm for iterative estimation of motion parameters is described below:

- Form a Gaussian pyramid from the images A and B
- Set the initial parameters and the covariance matrix to their priors as: $\hat{\mathbf{x}} = \mathbf{x}_-$ and $\mathbf{P} = \mathbf{P}_-$
- Starting from coarsest to finest level, perform multiple iterations of the following steps:
  - Warp image B using current estimate $\hat{\mathbf{x}}$ of motion parameters according to Figure 4 to form image $W(B;\hat{\mathbf{x}})$.
  - Obtain spatial and temporal gradients between image A and the warped image $W(B;\hat{\mathbf{x}})$.
  - To reduce computations, perform non-maximal suppression on the spatial gradient magnitude image and select only the points that are local maxima.
  - Compare the residuals of these points with their expected covariances in equations (32) and (31) to reclassify them as inliers and outliers. In the first iteration, use equation (31).
  - Use optical flow constraint with parametric motion model on inlier points to apply incremental correction in motion parameters and their covariances according to equations (21) and (22).

5. DETECTION AND POST-PROCESSING

After motion compensation, the features on the ground plane would be aligned between the two frames, whereas those due to stationary and moving objects would be misaligned. Image difference between the frames would therefore enhance the objects, and suppress the road features. However, the image difference depends on residual motion as well as the spatial gradients at that point. In highly textured regions, the image difference would be large even for small residual motion, and in less textured regions, the image difference would be small even for large residual motion. To compensate this effect, normalized frame difference [33] is used. This is given at each pixel by:

$$\frac{\sum g_t\sqrt{g_x^2+g_y^2}}{k+\sum(g_x^2+g_y^2)} \quad (33)$$

Figure 6: Test-bed for the "Mobile Sentry" experiments: An ODVS camera is mounted on an electric cart during a mobile sentry experimental run.

where g_x, g_y are spatial gradients, g_t is the temporal gradient. Constant k is used to suppress the effect of noise in highly uniform regions. The summation is performed over a 3×3 neighborhood of each pixel. In fact, the normalized difference is a smoothed version of the normal optical flow, and hence depends on the amount of motion near the point. Blobs corresponding to object features are obtained using morphological operations. Nearby blobs are clustered into one, and tracked from frame to frame. The position of the center of the bounding box is converted to the angle it makes with the X axis and and given as output. For each track that survives over a minimum number of frames, the original ODVS image is used to generate a perspective view [18] of the event around the center of the bounding box.

6. EXPERIMENTAL VALIDATION AND RESULTS

The above approach was used for event detection from a mobile platform. To simulate a mobile platform, an ODVS camera was mounted on an electric cart as shown in Figure 6. The cart was driven on a campus road at speeds between 3 and 7 miles/hour. The approximate speed of the cart was determined using GPS and used as a-priori motion estimate. However, there were considerable vibrations of the camera. Using the parametric motion estimation process, the effect of these vibrations was suppressed. It was also observed that the ellipse corresponding to the entire FOV of the ODVS was oscillating, possibly due to relative vibrations between the camera and the mirror, or the automatic motion stabilization in the camera. These were suppressed by estimating the center of the FOV ellipse using a Hough transform, and translating it to a fixed position.

Figure 7 (a) shows an image from the ODVS video sequence. The estimated parametric motion is shown using red arrows. Figure 7 (b) shows the classification of points into inliers (gray), outliers (white), and unused (black) points. It should be noted that the outliers are usually identified when the edges are perpendicular to the motion. When an edge is parallel to the motion, aperture problem makes it

Table 1: Performance evaluation. The right column shows the ground truth number of relevant events in the image sequence. The other columns show the number and percentage of events detected by the system. The last three rows show the number of false alarms due to stationary objects and shadows. Ground truth is not relevant here.

Minimum track length	15 frames	10 frames	Ground truth
Total Events	14 (74%)	17 (89%)	19
– Persons	9 (90%)	9 (90%)	10
– Vehicles	5 (55%)	8 (89%)	9
Total false alarms	3	4	N.A.
– Stationary objects	1	2	N.A.
– Shadows	2	2	N.A.

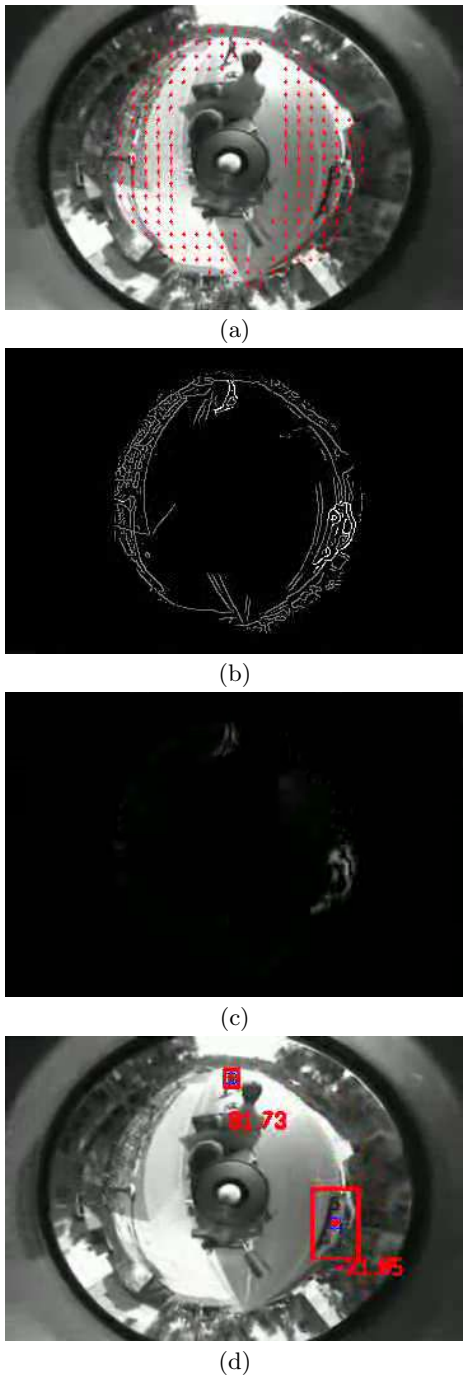
difficult to identify it. The estimation is done only using the inlier points. Image with the normalized frame difference between the motion compensated frames is shown in Figure 7 (c). It is seen that the independently moving car and person stand out whereas the stationary features on ground are attenuated in spite of ego-motion. Figure 7 (d) shows the bounding boxes around the moving car and person after post-processing.

Since the algorithm uses a planar motion model, stationary objects above the ground induce motion parallax, and are detected if they are sufficiently close to the camera, and included in the region of interest. Figure 8 shows the detection of a stationary structure. Only the part within the region of interest is detected.

The centroids of the detected bounding boxes were tracked over time and the tracks that survived over 10 or more frames were identified. Typical snapshots from these tracks were taken, and the distortion due to ODVS was corrected to get the perspective view looking towards the track position as in [18]. Figure 9 show the snapshots from these tracks, detecting the events.

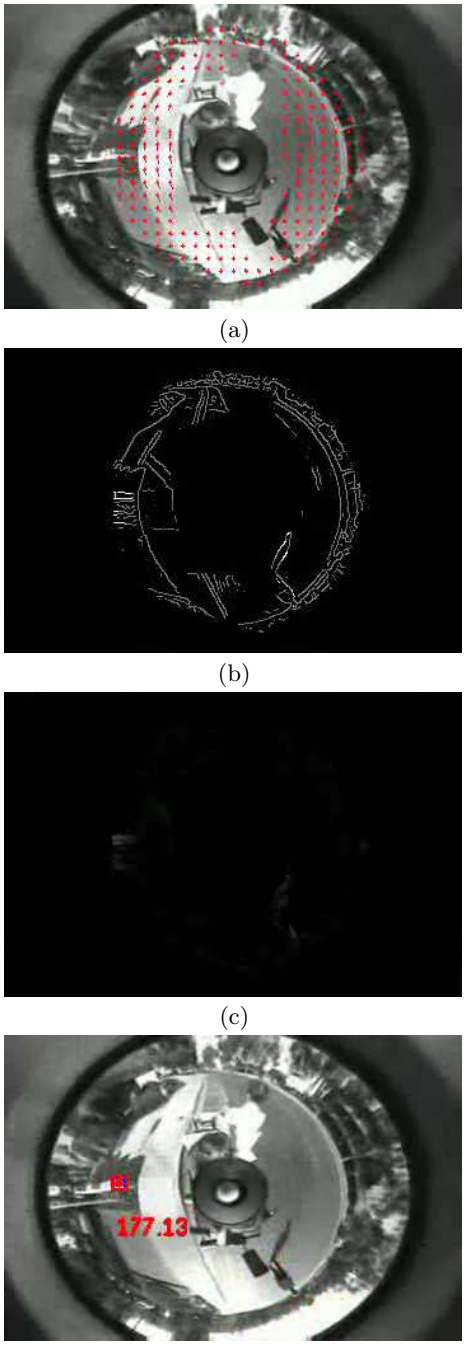
To evaluate the algorithm performance, the detection results were compared with ground truth obtained by manually observing the video sequence. The performance was compared for two different thresholds on the number of frames in which a track has to survive to be detected as an event. Table 1 shows the detection rate in terms of total number of events (ground truth), and the number of events actually detected. Note that stationary obstacles and shadows are classified as "false alarms", since they are currently not separated from independently moving objects. Lower threshold increases detection rate, but also increases false alarms. It was observed that an events corresponding to a moving person and cart were not detected at all due to the following reasons. The person and cart were quite far and especially the person had a small size in the image. Furthermore, the camera vehicle was turning, inducing considerable rotational ego-motion. Also, the objects were near the boundary of the region of interest that was analyzed. An image of this person is shown in Figure 10.

The events given by the motion algorithm can be analyzed in order to extract attributes such as the time, duration, position and classification of the event. For this purpose, the robustness for detection and localization of the event needs to be improved. But to illustrate the concept, the image



(a) (b) (c) (d)

Figure 7: Detection of moving objects (a) Image from a sequence using ODVS, with moving car and person (b) Expected motion of image points if they were on ground plane (taken within a region of interest) (c) Motion compensated difference image (d) Post-processed image showing detection of the moving car and person. The angle made with X-axis in degrees is also shown.



(a)

(b)

(c)

(d)

Figure 8: Detection of stationary structure (a) Image from a sequence using ODVS, with a stationary structure above ground (b) Expected motion of image points if they were on ground plane (taken within a region of interest) (c) Motion compensated difference image (d) Post-processed image showing detection of the the stationary structure. The angle made with X-axis in degrees is also shown. Note that only the part of the structure within the region of interest is detected.

(a)

(b)

Figure 9: Captured events: (a) Detected persons and vehicles. (b) Shadows and stationary obstacles currently considered false alarms.

Figure 10: Original image corresponding to a missed events. The moving person and vehicle were far from the camera and near the boundary of region of interest. Also, the camera vehicle was taking a turn, inducing considerable rotational ego-motion in the image.

Table 2: Summarization of event attributes. The left image shows the original ODVS image at the time of the event, middle image shows the output of detection algorithm, and the right image shows the snapshot of the event corrected for ODVS distortion



```
Event time (snapshot):  16:01:08.3
Event duration [seconds]:  3.6
Event position [meters]:  (7.1, -6.6)
Camera position [meters]:  (7.8, -5.3)
```



```
Event time (snapshot):  16:01:40.0
Event duration [seconds]:  1.9
Event position [meters]:  (53.1, 1.8)
Camera position [meters]:  (56.0, -3.1)
```

positions of some of the events as well as landmarks around them were manually marked. Using the world coordinates of the landmarks obtained from GPS and physical measurements, the approximate positions of the events could be determined. Figure 2 shows the way the events and their attributes could be summarized and stored in a database.

7. SUMMARY AND FUTURE WORK

This paper described an approach for event detection using ego-motion compensation from mobile omni-directional (ODVS) cameras. It applied the concept of direct motion estimation using image gradients to ODVS cameras. The motion of the ground was modeled as planar motion, and the features not obeying the motion model were separated as outliers. An iterative estimation framework was used for optimally fusing the motion information in image gradients with a-priori known information about the camera motion and calibration. Coarse to fine motion estimation was used and the motion between the frames was compensated at each iteration. A scheme based on data snooping was used to remove outliers. Experiments were performed by obtaining image sequences from a mobile platform, and detecting events such as moving persons and automobiles.

For future work, we plan to improve the robustness of the system especially for correct localization, especially for large objects. The algorithm currently detects regions containing edges where motion information is significant, but does not respond to uniform areas of large objects. Morphological operations were helpful in combining the detected regions, but a systematic approach based on region-based segmentation and clustering may be more appropriate for getting accurate localization in terms of bounding boxes. The events can then be classified into categories such as persons and vehicles using criteria such as size and shape. Learning based approaches such as [28] would also be useful for classification.

The method described above is appropriate for scenes where the background is predominantly planar, and the foreground consists of outliers in form of small objects. If the scene is not that simple, motion segmentation should be performed along with estimation. For example, the scene can be assumed to have a piecewise planar model and motion segmentation could be performed to separate multiple planar surfaces [12, 26]. Alternatively, the motion parameters can be estimated using a bootstrap method from small patches, and combine the patches having motion consistent with ground plane as done by Ke and Kanade [22]. For 3-D scenes with large variations in depths, structure from motion approach using epipolar constraint [6] is more appropriate. Alternatively, the plane+parallax method proposed by Irani and Anandan [19], can be used for wide variety of scenes including planar, piecewise planar, and 3-D.

To discriminate between independently moving objects and stationary objects above the ground, the rigidity constraint [19] could be used in the plane+parallax framework. We plan to generalize the piecewise planar motion segmentation as well as plane+parallax methods for use with ODVS cameras using non-linear motion models for complex scenes and independent motion discrimination.

The experiments for this work were performed using an ODVS camera mounted on an electric cart, driven by a person. Our laboratory has designed a system called Mobile Interactive Avatar (MIA) [14] in which cameras and displays are mounted on semi-autonomous robot to interact with people at a distance. We also plan to use MIA in our future experiments for the mobile sentry.

8. ACKNOWLEDGEMENTS

We are thankful for the grant awarded by the Technical Support Working Group (TSWG) of the US Department of Defense which provided the primary sponsorship of the reported research. We also thankful for the contributions and support of our colleagues from the UCSD Computer Vision and Research Laboratory.

9. REFERENCES

[1] O. Achler and M. Trivedi. Real-time traffic flow analysis using omnidirectional video network and flatplane transformation. In *Workshop on Intelligent Transportation Systems*, Chicago, IL, 2002.

[2] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with applications to tracking and navigation*. John Wiley and Sons, 2001.

[3] R. Benosman and S. B. Kang. *Panoramic Vision: Sensors, Theory, and Applications*. Springer, 2001.

[4] M. J. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63(1):75–104, 1996.

[5] S. Carlsson and J. O. Eklundh. Object detection using model-based prediction and motion parallax. In *European Conference on Computer Vision*, pages 297–306, April 1990.

[6] P. Chang and M. Herbert. Omni-directional structure from motion. In *IEEE Workshop on Omnidirectional Vision*, pages 127–133, Hilton Head Island, June 2000. IEEE Computer Society.

[7] K. Daniilidis, A. Makadia, and T. Bulow. Image processing in catadioptric planes: Spatiotemporal derivatives and optical flow computation. In *IEEE Workshop on Omnidirectional Vision*, pages 3–12, June 2002.

[8] G. Danuser and M. Stricker. Parametric model fitting: From inlier characterization to outlier detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(2):263–280, March 1998.

[9] K. L. Dean. Smartcams take aim at terrorists. Wired News, June 2003. `http://cvrr.ucsd.edu/press/articles/Wired News Smartcams.html`.

[10] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint*. The MIT Press, Cambridge, MA, 1993.

[11] T. Gandhi, S. Devadiga, R. Kasturi, and O. Camps. Detection of obstacles using ego-motion compensation and tracking of significant features. *Image Vision and Computing*, 18(10):805–815, 2000.

[12] T. Gandhi and R. Kasturi. Application of planar motion segmentation for scene text extraction. In *Int. Conf. on Pattern Recognition*, volume 1, pages 445–449, 2000.

[13] J. Gluckman and S. Nayar. Ego-motion and omnidirectional cameras. In *Proceedings of the International Conference on Computer Vision*, pages 999–1005, 1998.

[14] T. B. Hall and M. M. Trivedi. A novel interactivity environment for integrated intelligent transportation and telematic systems. In *5th International IEEE Conference on Intelligent Transportation Systems*, Singapore, September 2002.

[15] Haritaoglu, D. Harwood, and L. S. Davis. W4: Real-time surveillance of people and their activities. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 22(8):809–830, Aug. 2000.

[16] B. Horn and B. Schunck. Determining optical flow. In *DARPA81*, pages 144–156, 1981.

[17] K. Huang, M. Trivedi, and T. Gandhi. Driver's view and vehicle surround estimation using omnidirectional video stream. In *IEEE Conference on Intelligent Vehicles*, Columbus, OH, June 2003.

[18] K. C. Huang and M. M. Trivedi. Video arrays for real-time tracking of persons, head and face in an intelligent room. *Machine Vision and Applications*, 14(2):103–111, 2003.

[19] M. Irani and P. Anandan. A unified approach to moving object detection in 2D and 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(6):577–589, June 1998.

[20] B. Jähne, H. Haußecker, and P. Geißler. *Handbook of Computer Vision and Applications*, volume 2, chapter 14, pages 397–422. Academic Press, San Diego, CA, 1999.

[21] K. Kanatani. *Geometric Computation for Machine Vision*. Oxford University Press, Oxford, 1993.

[22] 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 Conference on Computer Vision and Pattern Recognition*, volume I, pages 390–397, June 2003.

[23] W. Kruger. Robust real time ground plane motion compensation from a moving vehicle. *Machine Vision and Applications*, 11:203–212, 1999.

[24] S. Lawson. Yes, you are being watched. PC World, December 27, 2002. `http://www.pcworld.com/news/article/0,aid,108121,00.asp`.

[25] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[26] J. M. Odobez and P. Bouthemy. Direct incremental model-based image motion segmentation for video analysis. *Signal Processing*, 66:143–145, 1998.

[27] D. Ramsey. Researchers work with public agencies to enhance super bowl security, February 15 2003. `http://www.calit2.net/news/2003/2-4_superbowl.html`.

[28] M. Saptharishi, J. B. Hampshire, and P. K. Khosla. Agent-based moving object correspondence using differential discriminative diagnosis. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, volume II, pages 652–658, June 2000.

[29] O. Shakernia, R. Vidal, and S. Sastry. Omnidirectional egomotion estimation from back-projection flow. In *IEEE Workshop on Omnidirectional Vision (in conjunction with CVPR)*, June 2003.

[30] J. Shi and C. Tomasi. Good features to track. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 593–600, 1994.

[31] E. P. Simoncelli. Coarse-to-fine estimation of visual motion. In *Proc. Eighth Workshop on Image and Multidimensional Signal Processing*, pages 128–129, Cannes, France, 1993.

[32] C. Stauffer and W. E. L. Grimson. Adaptive background mixture model for real-time tracking. In *Proceedings of IEEE Int'l Conference on Computer Vision and Pattern Recognition*, pages 246–252, 1999.

[33] E. Trucco and A. Verri. *Computer vision and applications: A guide for students and practitioners*. Prentice Hall, March 1998.

[34] R. F. Vassallo, J. Santos-Victor, and H. J. Schneebeli. A general approach for egomotion estimation with omnidirectional images. In *IEEE Workshop on Omnidirectional Vision*, pages 97–103, June 2002.