

# Dynamic Panoramic Surround Map: Motivation and Omni Video Based Approach

Tarak Gandhi and Mohan M. Trivedi  
Computer Vision and Robotics Research Laboratory  
University of California San Diego La Jolla, CA 92093, USA  
tgandhi@ucsd.edu, mtrivedi@ucsd.edu

## Abstract

Awareness of what surrounds a vehicle directly affects the safe driving and maneuvering of an automobile. Surround information or maps can help in ethnographic studies of driver behavior as well as provide a critical input in the development of effective driver assistance system. In this paper, we introduce the concept of Dynamic Panoramic Surround (DPS) map that shows the nearby surroundings of the vehicle, and detects the objects of importance on the road. Omnidirectional cameras which give a panoramic view of the surroundings can be useful for visualizing and analyzing the nearby surroundings of the vehicle. A novel approach for synthesizing the DPS using stereo and motion analysis of video images from a pair of omni-directional cameras on the vehicle is developed. Successful generation of DPS in experimental runs on an instrumented vehicle testbed is demonstrated. These experiments prove the basic feasibility and show promise of omni video based DPS capture algorithm to provide useful semantic descriptors of the state of moving vehicles and obstacles in the vicinity of a vehicle.

## 1. Introduction and Motivation

According to the studies conducted by U.S. National Highway Traffic Safety Administration (NHTSA), their report "Traffic Safety Facts 2003" [1] states that there have been 38,252 fatal crashes with 42,643 fatalities due to accidents on US roads in 2003. A significant number of these fatalities are due to running off the road or collision with other objects. In recent years, considerable research is being performed for developing intelligent driver support systems which warn the driver of a possible runoff or collision and allow the driver to take timely action to avoid or at least reduce their impact. These systems use a sensor suite which may contain video cameras mounted in various positions, thermal infrared imagers, and active sensors

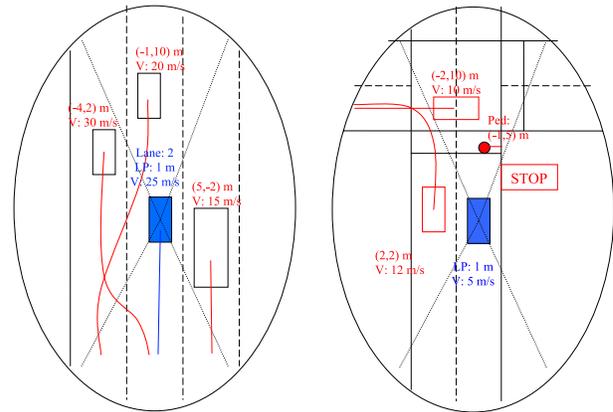


Figure 1. Illustration of a dynamic surround maps for Freeway and City streets.  $(x, y)$ : Coordinates of other objects w.r.t. ego-vehicle.  $V$ : Velocity w.r.t. road. LP: Lateral position of ego-vehicle w.r.t center of the lane

such as RADAR, LIDAR and ultrasonic sensors, as well as the data from vehicle dynamic sensors. Detecting lanes would help to determine the lateral position of the vehicle and warn the driver in case of lane departure and prevent scenarios such as collision with vehicles or fixed objects, or running off the road. Detecting front vehicles would be useful in preventing accidents due to sudden braking of the other vehicle. Monitoring sides of the vehicle containing blind spots would be especially useful when the driver intends to change lanes. Detecting non-motorists such as pedestrians and bicyclists is of particular interest for non-highway driving since pedestrians are harder to detect and are most vulnerable.

Thus, an effective driver assistance system should have a perception of complete surroundings including the events taking place in front, back, and sides of the car. Here, we introduce the concept of a dynamic surround map illustrated in Figure 1 that could be very useful for visualizing and analyzing the situation. The surround map would contain

lanes, vehicles, pedestrians, and other objects along with their attributes such as position, size, and velocity. Surround map generation would also be useful for off-line analysis of interesting events, especially in driver behavioral studies [14].

Omnidirectional cameras with a 360 degree panoramic view are naturally suited for synthesizing the surround maps. Due to their large field of view and lower resolution, they are appropriate for detecting nearby objects using small number of sensors. In the following we present our research dealing with the framework, development, and experimental verification of obtaining “Dynamic Panoramic Surround” (DPS) maps using a pair of omnidirectional cameras mounted on two sides of an automobile. Overlapping parts of video images from the two omni cameras are used to detect objects in front of the vehicle using binocular stereo. For each side of the vehicle, a monocular view is available from the respective omni camera. Motion stereo is applied on consecutive frames of the side views to detect objects on the sides.

## 2. Related Research

Autonomous driving and mobile robotics were the main drivers in the development of some of the early video based obstacle detection and navigation systems. A number of good survey papers present some of those efforts. In [4, 3], Bertozzi et al. give a comprehensive survey of computer vision techniques used for intelligent vehicles. Approaches for lane, pedestrian, and obstacle detection are described and analyzed. [8] presents a survey about vision techniques used for traffic analysis from stationary platforms as well as moving vehicles. In [15], a survey of lane detection techniques and their characteristics is performed, and a novel method using steerable filters is proposed to detect lanes as well as the “bots-dots” markers especially found on California highways.

Omnidirectional cameras with their panoramic view are naturally suited for complete surround monitoring of vehicles. In [5] a monocular omnidirectional camera is used to compensate road ego-motion, and detect vehicles with height or independent motion. In [12], the V-disparity approach for stereo developed by [11] is applied to a pair of coaxially mounted omni cameras for monitoring blind spots behind the vehicle especially during reversing of the car. In [7], an omni camera mounted in a car obtains a panoramic view consisting of the surroundings as well as the driver’s face. The system synthesizes a virtual view of the surroundings as seen by the driver by estimating driver’s head orientation. In [2], virtual side views generated from omni camera are used to track surrounding vehicles by detecting their wheels.

Recently, there has been a shift towards the development

of video based systems for driver assistance rather than autonomous driving. For this, it is important that the information about potential dangers is presented to the driver in a non-distracting and reliable manner [6]. In this paper, our focus is mainly on detection, localization, and mapping of objects (moving or stationary) in the immediate vicinity of a moving vehicle with video sensors. A key differentiator of the research discussed in this paper is the desire to develop a dynamic panoramic surround map, in somewhat “holistic” manner rather than considering independent and separate modules for detecting objects in the front or side of the vehicle.

## 3. Vehicle Surround Capture with Omni-Video

Due to the comparatively lower resolution of omni cameras, proper configuration is very important for obtaining good coverage, sensitivity, and foreground-background discrimination. For example, cameras mounted on top of roof [5] can see vehicles at farther distance with better resolution. However, the disparity difference between vehicle and road is small, making it difficult to isolate the vehicles purely by stereo. Furthermore, such a configuration is not suitable for standard cars. A pair of coaxially mounted omni cameras was used by [12] to monitor blind spots behind the car effectively. However, a large part of the FOV is occupied by the car itself and is therefore unused.

Here, we use of a pair of cameras each near side view mirror as shown in Figure 2. The view in front of the car overlaps in the two cameras and is used for stereo analysis. Monocular view on both sides is obtained from each of the cameras and used for motion analysis. The disparity difference between vehicle and ground is larger than for a top-mounted camera making stereo discrimination easier. As a trade-off, the objects such as cars have smaller frontal area with window-mounted camera, reducing their image size and making it more difficult to detect vehicles that are farther away.

In addition, the views of the driver and passenger are also obtained and can be used to analyze the driver behavior. For example, in [7] the driver’s face is detected in a similar setup by fitting an ellipse to edges using randomized Hough transform, and tracked using Kalman filter. The orientation of the face is estimated using Hidden Markov Models, and used to generate the view that the driver observes. This approach has been shown to be robust to illumination changes, shadows and other imaging problems.

The following coordinate systems are used for the analysis as shown in Figure 2:

1. Vehicle coordinate system: Origin is on the road,  $Z$  axis points in front,  $Y$  axis points down, and  $X$  axis points towards right. The coordinates in this system are denoted by:  $P_0 = (X_0, Y_0, Z_0)^T$

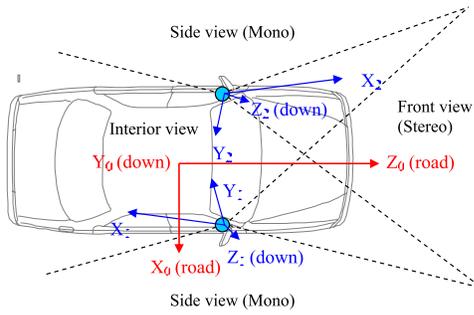


Figure 2. Camera configuration for surround capture and the relevant coordinate systems. Vehicle:  $(X_0, Y_0, Z_0)$ , Right omni:  $(X_1, Y_1, Z_1)$ , Left omni:  $(X_2, Y_2, Z_2)$

- Coordinate system of camera  $i$ : Origin is at camera optical center,  $Z$  axis is along optical axis, and  $X - Y$  axes form the image plane. The transformation between the vehicle coordinates  $P_0$  and the camera coordinates  $P_i$  for  $i = 1..2$  is represented as:

$$P_0 = R_i P_i + D_i, P_i = R_i^T (P_0 - D); \quad (1)$$

where  $R_i, D_i$  are the rotation matrix and translation vector of camera  $i$ .

- Omni pixel coordinates: Each point  $P_i$  in the camera  $i$ 's coordinate system is projected onto the omni camera pixels  $w_i = (u_i, v_i)^T$  by a many-to-one mapping  $f_i$  as:

$$w_i = f_i(P_i), P_i = \lambda g_i(w_i) + h_i(w_i) \simeq \lambda g_i(w_i) \quad (2)$$

where  $f_i$  denotes a many-to-one transform from camera to pixel, and  $g_i$  and  $h_i$  constitute the inverse transformation mapping every pixel  $w_i$  back to a 3-D line in the camera coordinate system. For central panoramic cameras, all projection lines pass through a single point on  $Z$  axis, which can be taken as origin, so that  $h_i = 0$ .

The omni cameras used in this study are central panoramic cameras, each consisting of a hyperbolic mirror and a camera placed on its axis, with the center of projection of the camera on one of the focal points of the hyperbola.

The block diagram of the video analysis framework is shown in Figure 3. Video sequences are obtained from a pair of omni cameras mounted on two sides of the vehicle. Camera calibration is performed off-line to determine the relationship between the vehicle and pixel coordinates. Using calibration information, the images are rectified to obtain virtual perspective views looking forward towards the road. Rectification simplifies the stereo geometry making it easier to match corresponding features between the two images. Area-based correlation is used to perform stereo

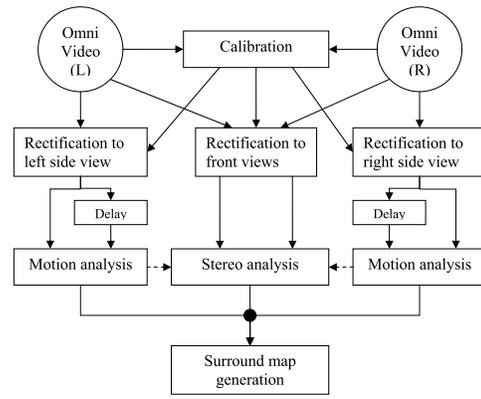


Figure 3. Block diagram of video analysis from vehicle mounted omni cameras.

matching between features and form a disparity map. Features with similar disparity are grouped into objects, and distance to the objects is computed. For analyzing the side views from each camera, motion analysis is applied to consecutive virtual perspective images from each camera on the respective sides. The disparity map has components corresponding to the object distance as well as independent motion. Detection is performed by assuming that the whole object has nearly same disparity and its position is computed assuming it touches the road. The object positions obtained from stereo and motion analysis are converted to the vehicle coordinates to insert them into the surround map.

#### 4. Camera Calibration

In order to match the points between multiple cameras, and map them to the 3D space, it is necessary to calibrate the intrinsic and extrinsic parameters of the cameras. The intrinsic parameters for transforming the pixel coordinates  $w_i$  and camera coordinates  $P_i$  can be pre-computed before the cameras are installed using a setup in [5] or [12]. The extrinsic parameters relating vehicle coordinates to each of the camera coordinates need to be calibrated when the cameras are mounted on the vehicle. In particular, the rotation matrices  $R_i$  of the cameras are used to obtain rectified perspective views of the overlapping FOVs of the omni cameras.

Calibration of extrinsic parameters is currently performed by taking a car into a scene with a number of parallel lines in the directions of the vehicle axes. Sample points on the lines are manually marked and using equation (2) along with intrinsic parameters, the pixel coordinates  $w_1, w_2 \dots w_K$  of sample points on each line are transformed to camera projective coordinates  $p_1, p_2 \dots p_K$  within scale factor, each corresponding to a ray from the camera origin in the direction of the line of sight. The co-

ordinates are normalized so that  $\|p_i\| = 1$ , and therefore correspond to projections on a virtual sphere centered at the camera origin. Singular value decomposition (SVD) of the following matrix is used to fit a line to the points.

$$L = \begin{pmatrix} p_1 & p_2 & \dots & p_K \end{pmatrix} \quad (3)$$

The left singular vector corresponding to the smallest singular value of  $L$  gives the vector  $l$  corresponding to line equation  $l^T p = 0$ . The procedure is repeated for a number of lines in the image.

Images of parallel lines intersect at the vanishing point corresponding to the point at infinity in the direction of the line. If a number of parallel lines have images represented by line vectors  $l_1, l_2 \dots l_N$ , the vanishing point  $m$  of these lines can be similarly obtained using SVD of:

$$M = \begin{pmatrix} l_1 & l_2 & \dots & l_N \end{pmatrix} \quad (4)$$

For lines along the length of the car (vehicle  $Z$  axis), the direction of the vanishing point is  $(0, 0, 1)^T$  in vehicle coordinate system. In camera coordinates, this is transformed to  $R^T(0, 0, 1)^T = r_3$  where  $r_3$  is the column 3 of  $R^T$  (or row 3 of  $R$ ). Similarly, the vanishing point of the lines across the length of the car (vehicle  $X$  axis) transforms to  $r_1$  (row 1 of  $R$ ) in the camera coordinates. For the  $Y$  direction, one can use vertical objects such as poles in the image if available. Otherwise, one can assume orthogonality and put  $r_2 = r_3 \times r_1$ . Due to errors in the determination of vanishing points, it is possible that the rows  $r_1, r_2, r_3$  matrix  $R$  may not all be exactly orthonormal. In such case, orthonormality can be enforced by taking the SVD  $R = USV^T$  and replacing  $R$  by  $UV^T$ .

## 5. Stereo and Motion Based Panoramic Surround Generation

The calibration obtained above is used to obtain virtual perspective view in front of the car from both the omni images. If the cameras are at the same height and longitudinal position, rectified images are obtained, in which the viewing directions of both images are parallel, and the baseline is perpendicular to the viewing direction. Suppose a point with vehicle coordinates  $(X, Y, Z)$  forms images at  $(x_l, y_l)$  and  $(x_r, y_r)$  in the respective virtual perspective planes of the two cameras with focal length  $f$  and baseline  $B$ . Then, these are related by:

$$x_l - x_r = fB/Z, \quad y_l = y_r \quad (5)$$

Stereo matching is simplified in this configuration where for every feature at  $(x_l, y_l)$  one tries to find the corresponding feature  $(x_r, y_r)$  which should lie along the same row. If difference between the camera heights and positions is

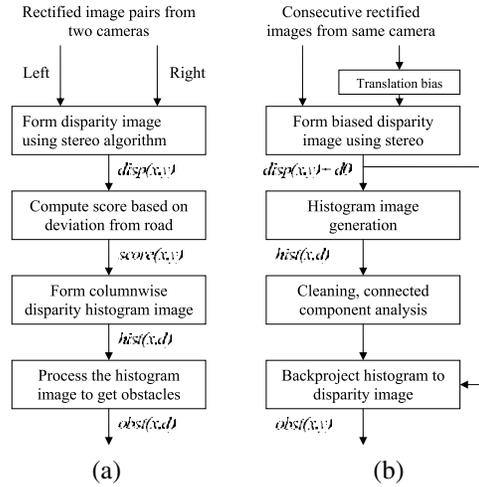


Figure 4. Flow charts for (a) Front object detection using binocular stereo analysis (b) Side object detection using motion analysis.

small, the vertical disparity  $y_l - y_r$  can be neglected for objects far from the cameras. We use area based correlation for matching in order to obtain dense depth map using the implementation in [9].

### 5.1. Stereo-based front object detection

The road is modeled as a planar surface, so that the disparity in rectified images is zero at the horizon, and linearly increases with  $y$ . Using the information from calibration module, the road disparity is computed. The pixels having disparities smaller than or equal to this disparity plus a small threshold are assumed to lie on the road and are suppressed. Also, pixels on a vehicle object are likely to have same disparity. To facilitate detection, pixels with same disparity in each column are clustered in a similar manner as in [11]. The steps in the above process (Figure 4 (a)) are:

1. Form an image  $score(x, y)$  based on the difference between the disparity  $d$  of the pixel and the corresponding disparity  $d_0$  if it were on road.

$$score(x, y) = \begin{cases} 0 & \text{if } disp(x, y) \leq d_0(y) \\ 1 & \text{if } disp(x, y) \geq d_0(y) + T \\ [disp(x, y) - d_0(y)]/T & \text{in between} \end{cases} \quad (6)$$

2. Form a columnwise disparity histogram image  $hist(x, d)$  such that:

$$hist(x, d) = \sum_{disp(x,y)=d} score(x, y) \quad (7)$$

Each column of this image is a histogram of the disparities in that column weighted by the score. Pixels in an

object at a particular distance would have nearly same disparity and form a horizontal line in this image.

3. Smooth the histogram image using Gaussian filtering and morphological closing. Threshold the image and find connected components.
4. Form a front obstacle image  $obst(x, d)$  by replacing pixels in each column of connected component in histogram image by the centroid. This averages the error in the disparity and gives a sub-pixel disparity value that is stored as the grey level in this image.

## 5.2. Motion-based side object detection

Each omni camera obtains a monocular view on the respective side of the car. Stereo matching is applied to consecutive frames from the same camera, so that the camera is displaced between the two frames. In absence of independent motion, the disparity of each feature is directly proportional to the camera velocity and inversely proportional to the distance to the object. However, in case of independently moving objects, their motion adds to the disparity, hence depth estimates cannot be derived directly from the disparity. But if the object is rigid and at nearly constant distance from the camera, its disparity is constant, and clustering can be used to extract the object. Note that the disparity can be negative in case of overtaking vehicles, whereas the stereo algorithm implementation [9] searches only positive disparities. Hence, the right image is displaced by a constant offset towards left, to bias disparity towards positive value. The object extraction procedure is a modification of that for binocular stereo (Figure 4 (b)) as described below:

1. Displace the right image by a fixed offset towards left to make disparities positive.
2. Use the implementation for binocular stereo [9] to form disparity image  $disp(x, y)$ .
3. Form the disparity histogram image  $hist(x, d)$  using steps similar to (1) to (3) in the previous algorithm.
4. For each component in  $hist(x, d)$ , find pixels in  $disp(x, y)$  contributing to the component. Form the side obstacle image  $obst(x, y)$  by replacing each column of these pixels in  $disp(x, y)$  with the bottommost pixel.
5. Assuming that the bottom of the object lies on ground compute its vehicle coordinates using the calibration.

## 5.3. Panoramic surround generation

The front and side obstacle maps obtained above are used to generate the surround map. Suppose the two omni cameras are situated at  $(-B/2, -H, Z_{off})$  and  $(+B/2, -H, Z_{off})$  in the vehicle coordinate system where  $B$  is the baseline,  $H$  is the camera height, and  $Z_{off}$  is the longitudinal distance between the camera and the car's center. Let the centers of the rectified images be at pixel  $(x_c, y_c)$  and  $f$  be the focal length corresponding to the pixel scale. Then, for each pixel  $(x_1, y_1)$  in the front obstacle map (w.r.t. left image), the 3-D vehicle coordinates are given by:

$$Z_0 = \frac{f}{d}B + Z_{off}, X_0 = \frac{x_1 - x_c}{d}B - \frac{B}{2}, Y_0 = H \quad (8)$$

In case of side obstacle map, the disparities do not give distance information for moving objects. Hence, it is assumed that the bottom of the bounding box of obstacle ( $y_b$ ) lies on ground and the  $y$  coordinate of the pixel is used to find the object distance. This gives:

$$X_0 = \mp \left[ \frac{y_b - y_c}{f}H + \frac{B}{2} \right], Z_0 = \pm \frac{x - x_c}{y - y_c}H + Z_{off} \quad (9)$$

where in  $\pm, \mp$  the top sign is for left camera and bottom sign is for right camera. The vehicle  $(X, Z)$  coordinates of each obstacle pixel in front and side are projected on the panoramic surround map forming contours corresponding to each component. To smooth the contours and fill the gaps, a morphological opening operation is applied to the contours to keep errors on the side of caution by assigning closer object distance when in doubt.

## 6. Dynamic Panoramic Surround Maps: Experimental Studies

The LISA-Q intelligent test bed shown in Figure 5 (a) is designed as a system capable of collecting large amounts of data from a variety of modular sensing systems and processing that data in order to be fed back to the human occupant. Sensor systems include rectilinear cameras, wide FOV camera systems, GPS and navigation systems, and the data from internal automobile vehicle state sensors available on the Controller Area Network (CAN) bus. The system contains an array of computers that serve for data collection as well as real-time processing of information. Detailed information about this test bed is described in [13].

For this study, a pair of omni cameras were mounted on the windows and the car was driven on the roads around and outside the campus. The video from the cameras was digitized with image size  $320 \times 240$  and stored on the system. Figure 5 (b) shows the images from the left and right omni

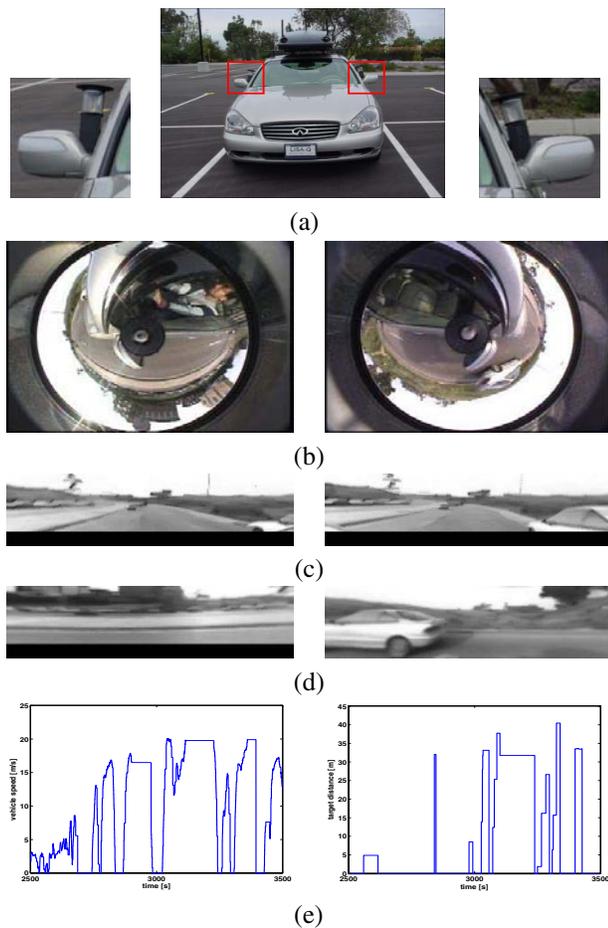


Figure 5. (a) The LISA-Q intelligent vehicle test bed. Inset are close up views of the omni cameras on two sides of the vehicle. (b) Video images from omni cameras on left and right side of the vehicle. (c) Virtual front views from left and right cameras. (d) Virtual side views from left and right cameras. (e) Time series of own vehicle speed and radar object distance from CAN bus.

cameras. The virtual front views generated from both cameras are shown in (c). The virtual side views on respective sides are in (d). The time series of own vehicle speed and radar object distance acquired from CAN bus are shown in (e).

For calibration, the car was moved around in a parking lot with parallel lines. Note that the parking lines are mapped to curves in the omnidirectional image. The points on the lines were manually marked as shown in Figure 6 (a)-(b) and calibration was computed using the method in Section 4. Rectification using this calibration generates virtual perspective views in front of the car as shown in Figure 6 (c)-(d). The camera parameters measured by this approach would be prone to camera vibrations and drift. To compensate for this error, small correction in form of vertical translation was manually applied to one of the recti-

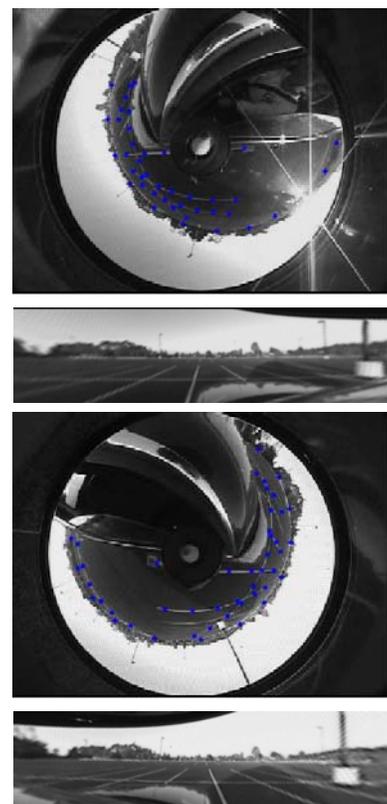


Figure 6. Points on the lines in parking lot as well on the horizon line are used to obtain the calibration for omni cameras. Virtual perspective images from the cameras corresponding to view in front of the car are shown.

fied images so that the disparities lie exactly along horizontal lines. For automatic operation, incremental calibration should be performed, possibly by measuring the image motion of the vehicle frame and using that to compute the changes in extrinsic parameters.

After calibration, the car was driven on roads outside the campus. Figure 7 shows the results of surround map generation on snapshots from the video sequence in which another car is overtaking from the left side. In the first two snapshots, the car is detected in side images using motion analysis. Each image set contains 6 images showing the detection steps. Top images show successive images from the virtual right side views obtained from the right omni camera. Implementation in [9] was used to compute the disparity map shown mid-right. One of the images was translated to bias the disparities to positive side. The superimposition of images is shown in mid-left and the disparity map is shown in mid-right. Though the car is nearer to the camera than the background, the independent motion of the overtaking car in the forward direction produces negative overall disparity, which when biased becomes a positive value smaller than that from the distant background. The histogram image of

the columns of the disparity map clusters these disparities as shown in bottom-left, with the vehicle and background layers concentrating as line segments. However, regions having no significant features do not get disparity estimates, hence there are breaks in the segments. For each column in the disparity histogram, pixels having the corresponding disparity are searched in the disparity map, and the bottom-most pixel in these is considered to be the bottom of the layer for that column as recorded in the bottom-right. These pixels are projected onto the surround map showing the detected car.

In the last two snapshots, the car goes in front view, and stereo analysis is used to detect it. The top images in each image set show the virtual views of the front from the two cameras. The superimposition of these two images in the mid-left image clearly shows the disparity between the images for the object car and the computed disparity map is shown mid-right. The histogram image of the columns of the disparity map was computed as shown in bottom-left with each row in this image corresponding to a particular disparity. Each pixel in this image has the brightness proportional to the weighted sum of the pixels in that column having the corresponding disparity. Since the front features of the car have nearly same disparity, they accumulate in form of a horizontal segment in the disparity image and can be easily isolated. In bottom-right, the histogram image is processed to give obstacle image showing the column position and the disparity of the object. Using this information and the camera calibration, the segment corresponding to the detected car is projected to the surround map. The transition of the detected car from the side to front view is demonstrated.

The size of the objects in the image determines the distance at which they can be detected reliably. It was observed that at the distance of 10 to 15 meters, a typical car had size of approximately  $16 \times 10$  pixels in omni image and  $25 \times 16$  pixels in the virtual perspective image. The disparity was about 20 pixels and the object could be detected reliably. However, at distances of 40 to 50 meters, the omni resolution decreased to  $5 \times 5$  pixels, virtual image resolution to  $10 \times 4$  pixels, and disparity to approximately 6 pixels. The detection at this distance was somewhat sporadic and the estimates of range and therefore lateral position were less accurate. A higher resolution camera such as  $1K \times 1K$  could be used to improve the accuracy and resolution.

## 7. Concluding Remarks

This paper emphasized the importance of vehicle surround for use in intelligent driver support systems as well as for ethnographic analysis of driver behavior. A novel framework for synthesizing a “Dynamic Panoramic Surround (DPS)” map using a pair of omni directional cameras

mounted on sides of the vehicle was described. For vehicle detection, stereo analysis was used on the overlapping views of the front of the car, and motion analysis was used for monocular views of the sides of the car. Experimental runs with an instrumented test vehicle were conducted to show the basic feasibility of omni video based surround capture algorithm. It was observed that resolution of omni cameras is low for the front parts of the image. Due to this, segmentation and depth estimation becomes more challenging. However, use of only two sensors for analyzing front as well as side views make the approach attractive for designers. Increasing the resolution of the omni image would give better reliability for detection and measurements of range and lateral position.

It is important to acknowledge that an effective and useful approach for surround based driver assistance system, needs systematic and careful “human-factors” oriented investigations in addition to the development of the novel surround capture technology. Such multidisciplinary studies consider optimum means for presenting information about potential dangers to the driver in a non-distracting and reliable manner. Collaborations with experts from human machine interactions, cognitive science, and psychology would be essential to make progress in this area [16, 6, 10].

## References

- [1] Traffic safety facts 2003 FARS/GES annual report (Final Edition). Technical report, US Dept. of Transportation, National Highway Traffic Safety Administration, 2004.
- [2] O. Achler and M. M. Trivedi. Vehicle wheel detector using 2D filter banks. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 25–30, June 2004.
- [3] M. Bertozzi, A. Broggi, M. Cellario, A. Fascioli, P. Lombardi, and M. Porta. Artificial vision in road vehicles. *Proceedings of the IEEE*, 90(7):1258–1271, 2002.
- [4] M. Bertozzi, A. Broggi, and A. Fascioli. Vision-based intelligent vehicles: State of the art and perspectives. *Robotics and Autonomous Systems*, 32(1), 2000.
- [5] T. Gandhi and M. M. Trivedi. Parametric ego-motion estimation for vehicle surround analysis using an omnidirectional camera. *Machine Vision and Applications*, 16(2):85–95, February 2005.
- [6] P. Green. Driver distraction, telematics design, and workload managers: Safety issues and solutions. In *Proceedings of the 2004 International Congress on Transportation Electronics*, number 2004-21-0022 in P-387, pages 165–180, Warrendale, PA, 2004. Society of Automotive Engineers.
- [7] K. Huang, M. M. Trivedi, and T. Gandhi. Driver’s view and vehicle surround estimation using omnidirectional video stream. In *IEEE Intelligent Vehicles Symposium*, pages 444–449, Columbus, OH, June 2003.
- [8] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis. A survey of video processing techniques for traffic applications. *Image and Vision Computing*, 21:359381, 2003.

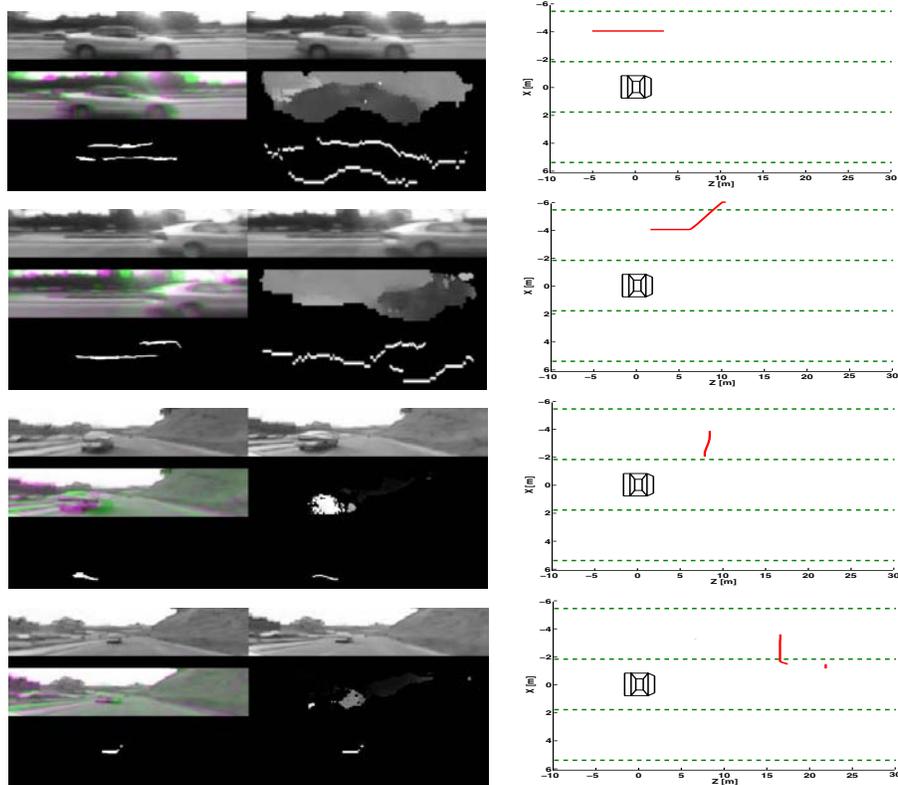


Figure 7. Results of surround generation detecting a vehicle passing from left. The first two image sets are the left side views analyzed using motion and the last two are front views analyzed using stereo. Each image set contains 6 images showing steps in detection: Top: Rectified left and right images for stereo, consecutive images for motion. Mid left: Composite image. Mid right: Disparity image. Bottom left: Disparity histogram image. Bottom right: Obstacle image. Surround maps show own car (black) and approximate positions of other vehicles (red).

- [9] K. Konolige. Small vision system: Hardware and implementation. In *Eighth International Symposium on Robotics Research*, pages 111–116, October 1997. <http://www.ai.sri.com/~konolige/papers>.
- [10] N. Kuge, T. Yamamura, O. Shimoyama, and A. Liu. A driver behavior recognition method based on a driver model framework. In *SAE 2000 World Congress, Session: Human Centered Driver Assistance Systems*, March 2000.
- [11] R. Labayrade, D. Aubert, and J.-P. Tarel. Real time obstacle detection in stereovision on non flat road geometry through V-disparity representation. In *IEEE Intelligent Vehicles Symposium*, volume II, pages 646–651, 2002.
- [12] L. Matuszyk, A. Zelinsky, L. Nilsson, and M. Rilbe. Stereo panoramic vision for monitoring vehicle blind-spots. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 31–36, June 2004.
- [13] J. McCall, O. Achler, and M. M. Trivedi. Design of an instrumented vehicle testbed for developing human centered driver support system. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 483–488, June 2004.
- [14] J. McCall, O. Achler, M. M. Trivedi, P. Fastrez, D. Forster, J. B. Haue, and E. B. J. Hollan. A collaborative approach for human-centered driver assistance systems. In *7th IEEE Conf. on Intelligent Transportation systems*, October 2004.
- [15] J. McCall and M. M. Trivedi. Video based lane estimation and tracking for driver assistance: Survey, algorithms, and evaluation. Technical report, CVRR, December 2004.
- [16] M. A. Recarte and L. M. Nunes. Mental workload while driving: Effects on visual search, discrimination, and decision making. *Journal of Experimental Psychology: Applied*, 9(2):119–137, 2003.