

Driver Assistance System Providing an Intuitive Perspective View of Vehicle Surrounding

Yen-Ting Yeh¹, Chun-Kang Peng², Kuan-Wen Chen³, Yong-Sheng Chen², and Yi-Ping Hung^{1,4}

¹ Grad. Inst. of Networking and Multimedia, National Taiwan Univ., Taipei, Taiwan

² Dept. of Computer Science, National Chiao Tung Univ., Hsinchu, Taiwan

³ Intel-NTU Connected Context Computing Center, National Taiwan Univ., Taipei, Taiwan

⁴ Dept. of Computer Science and Information Engineering, National Taiwan Univ., Taipei, Taiwan

Abstract. Driver assistance systems can help drivers to avoid car accidents by providing warning signals or visual cues of surrounding situations. Instead of the fixed bird's-eye view monitoring proposed in many previous works, we developed a real-time vehicle surrounding monitoring system that can assist drivers to perceive the vehicle surrounding situations in third-person viewpoints. Four fisheye cameras were mounted around the vehicle in our system. We developed a simple and accurate fisheye camera calibration method to dewarp the captured images into perspective projection ones. Next, we estimated the intrinsic parameters of each undistorted virtual camera by using planar calibration patterns and then obtain the extrinsic camera parameters by using the global patterns on a ground plane. A new method was proposed to tackle the brightness uniformity problem caused by the various lighting conditions of cameras. Finally, we projected the undistorted images onto a 3D hybrid projection model, stitched these images together, and then rendered the images from a third-person viewpoint selected by the driver. The proposed hybrid projection model is composed of a paraboloid model and a columnar model and can achieve rendering results with less distortion. Compared to conventional around-vehicle monitoring systems, our system can provide adaptive, integrated, and intuitive views of vehicle surroundings in a more realistic way.

1 Introduction

Blind spot of the vehicle is one of the major reasons causing car accidents. Helping drivers to perceive the situation around the vehicle while driving is the main goal of intelligent driving assistance systems. Toward this goal, many kinds of driving assistance systems have been developed. For example, parking sensors measure the distances of the obstacles from the vehicles and warn the drivers by beepers. Automatic braking system can stop or slow down the vehicle to prevent collisions. Rear-view cameras capture the views behind the vehicles and can help drivers to drive backward. Because the prices of cameras keep dropping, it

becomes practical to mount multiple cameras around the car to provide drivers with vehicle surrounding views for better perception of situations around the vehicles.

Ehlgen and Pajdla [1] proposed to use four omnidirectional cameras around a truck. They developed several ways to split the overlap area and provided a bird's-eye view image. But their integrated results are discontinuous on the seam between different cameras. In [2], Liu et al. mounted six fisheye cameras around the vehicle and stitched all six images together to provide a bird's-eye view of the vehicle surrounding environments. In their method, they applied one-dimensional stitching to improve the discontinuity issue on the seam at the expense of large computational costs. Recently, Nissan [3] developed an assistant system, Around View Monitor, which can render the surrounding view of the vehicle by using four wide-angle cameras. This system can provide drivers with both the aerial view and the original images captured by the four cameras. The Eagle Eye system from Luxgen [4] and Multi-View Camera System from Honda [5] used similar approaches to provide drivers better visual perception of surroundings. These two systems project the captured and undistorted images onto a ground plane and leave four black seams between adjacent cameras. Delphi Automotive also proposed a parking system, 360° Surround View System with Parking Guidance [6]. They applied a blending algorithm to provide smooth visual results from the aerial view. In driving situations like lane changing and passing, the visual information from aerial view may be inadequate for the drivers. Fujitsu also released a system, 360° warp-around video imaging system [7], which can project the acquired images onto a 3D curved plane and can provide drivers with a third-person view.

In this paper we propose a monitoring system for driving assistance, as its system flowchart illustrated in Fig. 1. First, we placed the fisheye camera on a rotating device controlled by a stepping motor, as shown in Fig. 4(a), to automatically calibrate the fisheye camera. Next, we mounted four of this kind of fisheye cameras around the vehicle and acquired distorted images from these cameras. Then we applied Zhang's method [8] to estimate the intrinsic parameters of each camera. Once the intrinsic parameters were obtained, these distorted images were dewarped into perspective ones and then mapped to the ground plane by using homography transformation. Extrinsic parameters of each camera were then calculated by the homography matrix and intrinsic parameters of this camera. Finally, we projected the undistorted images onto a proposed hybrid model and render the images from a driver-selected viewpoint. We used a look-up table approach and can finish the whole image mapping process in real time.

2 Correction of Image Distortion and Estimation of Intrinsic Parameters

This section describes the proposed methods of camera calibration and image dewarping. Because the field of view of the fisheye lens we adopted is so large

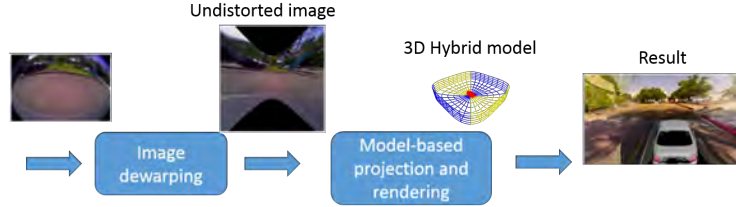


Fig. 1. Flowchart of the proposed system.

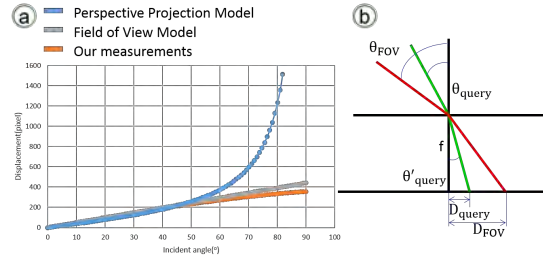


Fig. 2. (a) Relationship between incident angle and displacement of image point for perspective model, FOV model, and real fisheye camera. (b) Concept of the FOV model.

(183°), the widely-used FOV (field of view) fisheye lens model can not fit well. Instead of the full calibration procedure to determine the full set of camera parameters, therefore, we developed a simple method to dewarp fisheye camera images according to the relationship between the incident angle and the image formation distance of a point. Radial distortion of fisheye lens causes an inward or outward displacement of a given image point from its ideal location. There are many existing models for the calibration of fisheye cameras [9–12]. Devernay and Faugeras [13] used FOV model to calibrate the fisheye cameras. They used the idea that the distance between the ideal image point and the principal point is roughly proportional to the angle of incidence, as shown in Fig. 2(b). However, the FOV model is not suitable for all fisheye cameras. Fig. 2(a) shows the difference between the perspective model, FOV model, and our measurement. As described in Sec. 2.3, we conducted a simple experiment to figure out how the radial distortion changes when the incident angle increases. In Sec. 2.4, we describe how to apply Zhang’s method [8] to obtain the intrinsic parameters of the virtual camera, which is the undistorted counterpart of the fisheye camera. Combining with the homography matrix, we can estimate the extrinsic parameters of the virtual camera, as described in Sec. 3.

2.1 Aspect ratio of pixel

Because of the non-isometric pixel of fisheye camera, the circular shape of the fisheye lens border may form an elliptic shape in the captured image. The first

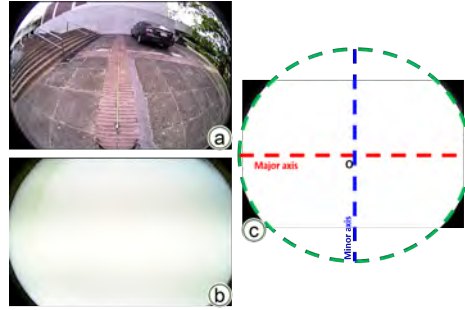


Fig. 3. (a) An image captured by a fisheye camera. (b) The image of a white wall captured by a fisheye camera. (c) Ellipse fitting result.

step of camera calibration is to correct the aspect ratio of pixels. As shown in Fig. 3, we performed the ellipse fitting [14] process to obtain the major and minor axes and center point of the ellipse. Then we rescaled the images along the minor axis so that the length of minor axis is equal to that of the major one. Once we obtained the circular images, we assume that the lens distortion is identical along any axis crossing the image center.

2.2 Feature point acquisition

In order to dewarp the fisheye image, we have to estimate the relationship between the incident angle and distortion measure. First, the fisheye camera was mounted on the center of a rotating table with its optic axis perpendicular to the normal vector of the rotating table. The rotating table was controlled by a stepping motor, as shown in Fig. 4(a). Next, we designated a feature point, which was a red circle pattern on a wall and adjusted the rotating table to locate this point at the image center of the camera, as shown in Fig. 4(b). In our experiments, the distance between the wall and the camera was 5 meters and we took a sequence of images while rotating the table, one image per 0.9 degrees. Furthermore, when the slanted fisheye camera mounted on the table was rotating, feature point on the image moved along an oblique line, as shown in Fig. 4(c). Consider the aspect ratio of pixels, α , the distance between the ellipse center (x_C, y_C) and the feature point (x_F, y_F) is:

$$D((x_C, y_C), (x_F, y_F)) = \sqrt{(x_C - x_F)^2 + (\alpha(y_C - y_F))^2} . \quad (1)$$

As a result, we can construct a table recording the mapping relationship between the rotation angle, which is equal to incident angle, and the distance from the feature point to the ellipse center. Here we denote this mapping relation as:

$$T(D_{f,i}) = \Theta_i , \quad (2)$$

where $D_{f,i}$ is the distance from the point i on fisheye image to the ellipse center and Θ_i is the corresponding incident angle.

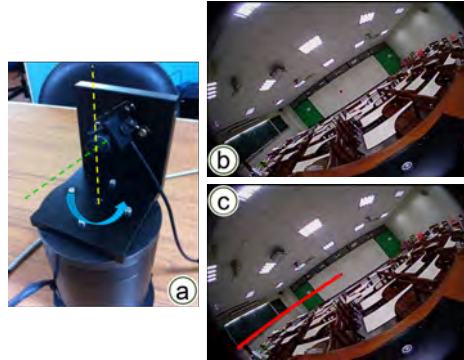


Fig. 4. (a) The rotating table controlled by a stepping motor. (b) The feature point on a wall five meters away from the camera. (c) Locus of feature points.

2.3 Mapping rule

According to the perspective projection model, incident angle Θ_i and the distance $D_{p,i}$ from the projection point on the projection plane to the center of the plane are related by:

$$f \tan(\Theta_i) = D_{p,i} , \quad (3)$$

where f is the focal length of the perspective projection camera model. FOV is the maximum of incident angle which can be projected on the plane. D_{FOV} is the $D_{p,i}$ when point i is projected with incident angle FOV , which is considered as a normalization term. We can then eliminate f by substitution and obtain the following equation:

$$\frac{D_{p,i}}{D_{FOV}} = \frac{\tan(\Theta_i)}{\tan(FOV)} . \quad (4)$$

By substituting (2) into (4), the relationship between $D_{f,i}$ and $D_{p,i}$ is derived as:

$$D_{p,i} = \frac{D_{FOV}}{\tan(FOV)} \tan(T^{-1}(D_{f,i})) . \quad (5)$$

We can use this relationship to undistort fisheye image to a perspective projection one. Two examples of undistorted fisheye images are shown in Fig. 5.

2.4 Intrinsic parameter

In order to obtain the relationship between world coordinate system and image coordinate system, we must find out the intrinsic and extrinsic parameters of our fisheye cameras. As mentioned before, the field of view of the fisheye lens we adopted is too large to use FOV model for calibration. Therefore, we estimate the parameters of the virtual camera, which is the undistorted counterpart of the fisheye camera. Following the standard procedure of Zhang's calibration

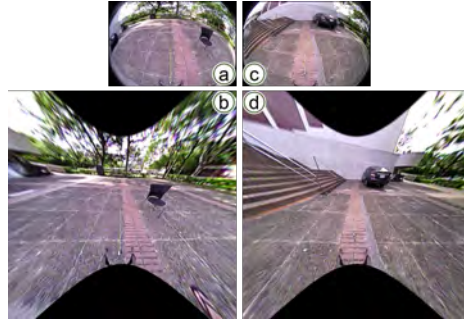


Fig. 5. (a)(c) The images captured by a fisheye camera. (b)(d) The corresponding undistorted results.



Fig. 6. (a) Chess board calibration pattern, (b)(e) the original fisheye images, (c)(f) undistorted images, and (d)(g) images in the calibration procedure.

method [8], we made a chess board as a calibration pattern and took multiple pictures with the chess board pattern in different positions and poses. Next, we undistorted the fisheye images by using the mapping rule described above. Finally we obtained camera intrinsic parameters by using the OpenCV procedure of Zhang’s method [8], as shown in Fig. 6.

3 Homography Transform and Extrinsic Parameters

3.1 Integration of camera views by homography

After fisheye camera calibration, the next step is to stitch four camera images to construct a monitoring image for vehicle surrounding view. We mounted four fisheye cameras around the vehicle for capturing images toward front, back, left, and right directions, as shown in Fig. 7(a). To stitch four images, we use the homography transformation to map the dewarped images onto a reference ground plane. In this work, the reference ground plane is a bird’s eye view image of a plaza captured on the roof of a building, as shown in Fig. 8(b). Forty-eight red circle patterns placed on the plaza were used as the features for the registration of dewarped fisheye images and the reference ground plane image, as shown in Fig. 7(a). At least four pairs of corresponding feature points were identified on both the ground plane image and the undistorted fisheye image. The

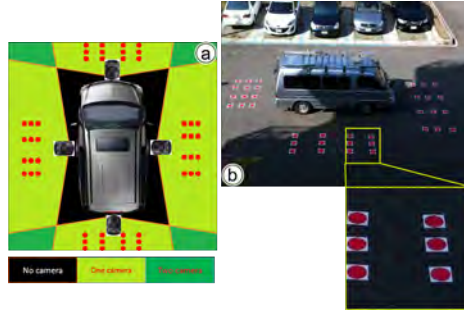


Fig. 7. (a) Four fisheye cameras mounted around the vehicle and 48 circle patterns for calibration. (b) Red circles for the estimation of homography transformation.

homography relationship and the coordinate transformation between these two images were then estimated from this corresponding relationship. RANSAC [15] (RANdom SAMple Consensus) was applied for better estimation results.

3.2 Extrinsic parameters

If we obtain the homography transformation matrix, we can know the relationship between each fisheye image and ground plane image. The extrinsic parameters, which denote the coordinate system transformation from fisheye camera coordinate system to the world coordinate system, are calculated by the following equations [8].

$$M_{ext} = [R_1 \ R_2 \ R_3 \ T] \quad (6)$$

$$H = [h_1 \ h_2 \ h_3] = s \cdot M_{int} \cdot [R_1 \ R_2 \ T] \quad (7)$$

$$R_1 = \frac{1}{s} \cdot M_{int}^{-1} \cdot h_1 \quad (8)$$

$$R_2 = \frac{1}{s} \cdot M_{int}^{-1} \cdot h_2 \quad (9)$$

$$T = \frac{1}{s} \cdot M_{int}^{-1} \cdot h_3 \quad (10)$$

and

$$R_3 = R_1 \times R_2 \ , \quad (11)$$

where h_1 , h_2 , h_3 are the column of homography matrix H , s is a scaling factor, M_{int} is the intrinsic matrix and M_{ext} is the extrinsic matrix which can be composed to the rotation matrix R_1 , R_2 , R_3 and translation matrix T .

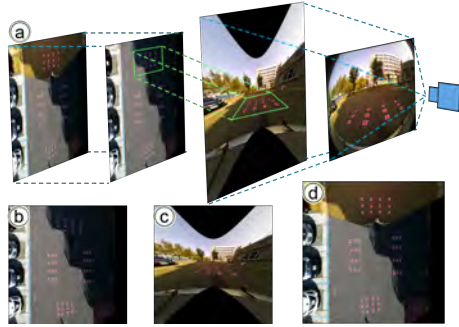


Fig. 8. (a) Homography mapping. (b) The ground reference plane. (c) Undistorted image captured by front fisheye camera. (d) Stitched images of (c) and (b) using homography transform.

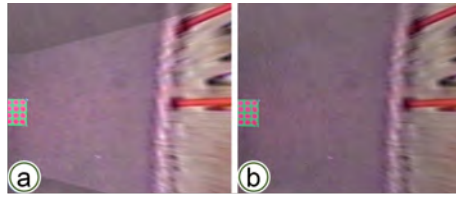


Fig. 9. (a) Before and (b) after brightness uniformity correction.

4 Brightness Uniformity and Image Blending

4.1 Brightness uniformity

Once the transformation matrix H is obtained, we use it to transform and map the fisheye image onto the ground plane image, as shown in Fig. 8(d). By repeating this process for each of the four fisheye cameras we can stitch all of the fisheye images into a panoramic one. But the various brightness of different cameras view may cause the obvious edges in the overlapping region, as shown in Fig. 9(a). In this work, we applied the following algorithm to tackle this problem.

Step 1: Separating the overlapping region from stitching image.

Step 2: Transforming from RGB to YUV color space and using Y value as luminance.

Step 3: Finding the sub-region gain of each pixel in overlapping region by the following equation:

$$G_{i,x,y} = \frac{L_{i,x,y} + L_{j,x,y}}{2 \times L_{i,x,y}}, G_{j,x,y} = \frac{L_{i,x,y} + L_{j,x,y}}{2 \times L_{j,x,y}}, \quad (12)$$

where i, j represent the indices of the cameras and $L_{i,x,y}$ represent the mean luminance of sub-region in camera i whose centroid coordinate is (x, y) . The size of each sub-region is 25×25 .

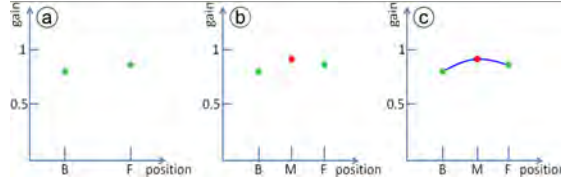


Fig. 10. (a) For each scan line, we first obtain the two boundary points named B and F. The green points indicate the two gain value of B and F. (b) We add a middle position gain named M whose gain value is calculated by averaging equations of the gains of B and F and unity. (c) Polynomial fitting to obtain the remaining gains.

Step 4: Smoothing the available gain with Gaussian kernel of size 25×25 .

Step 5: Polynomial fitting to obtain the remaining gains which are not in the overlapping region along horizontal or vertical scan line of stitching image. In order to maintain the automatic gain control mechanism of the used cameras, we set a center value close to unity in order to maintain the original information. As the polynomial fitting shown in Fig. 10, B and F are the two interpolation boundary called beginning and ending positions and M is the middle position of the scan line. The gain value on the position M is calculated by using the following averaging equation:

$$G_M = \frac{2 + G_B + G_F}{4} . \quad (13)$$

4.2 Linear blending

Because of the wide FOV angle of fisheye camera, each corner of the stitching image is within region overlapped to the view of other cameras. We applied linear blending process for the stitching smoothness. As shown below, linear blending is a linear combination process of overlapped pixels:

$$I^{linear}(x, y) = \frac{\sum_{i=1}^n I^i(x, y) W^i(x, y)}{\sum_{i=1}^n W^i(x, y)} , \quad (14)$$

where $I^i(x, y)$ is the image intensity of pixel in coordinate (x, y) of image i and $W^i(x, y)$ is the weight in coordinate (x, y) of image i . The shorter distance between overlapped pixel and image center, the larger the weight is. After the blending process, we can obtain the smooth stitched images, as shown in Fig. 11. Fig. 12 shows an overall result of dewarping and stitching images captured from four fisheye cameras.

5 Hybrid Model Projection

To build a system that can allow driver to select different third-person view and can synthesize the result image from the four images acquired by fisheye

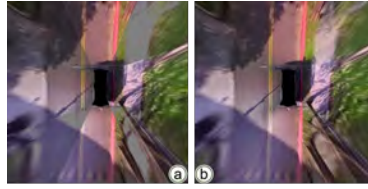


Fig. 11. (a) Stitched image without blending. (b) Stitched image with linear blending.

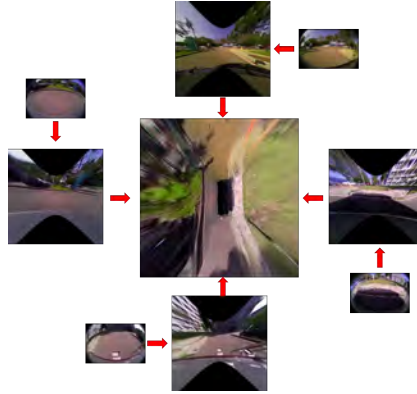


Fig. 12. The stitched result of four images from fisheye cameras.

cameras, an intuitive way is to perform view interpolation proposed by Chen [16]. However, in the vehicle surrounding monitoring application, computational cost is large to apply the algorithm of view interpolation when moving objects exist. Therefore, we design a 3D hybrid model and project four undistorted images onto it. Then we use intrinsic and extrinsic parameters of the selected viewpoint to synthesize the results. The concept of view interpolation is shown in Figure 13(a).

5.1 View Interpolation Using Hybrid Model

As other vehicle surrounding systems mentioned at section 1, we can stitch four dewarpped images into ground plane to see the objects around vehicle. However, there are two major problems in this way. First, distortion in the image is inevitable. Second, image textures above the vanishing point would be projected to an infinite position. In this way, the driver will not be able to see the scene above the vanishing line from the viewpoint. To solve these problems, we calculate the camera extrinsic parameters from homography matrix and camera intrinsic parameters and then project the acquired images to 3D model. We use back-projection to find the texture of 3D model. The projection equation is

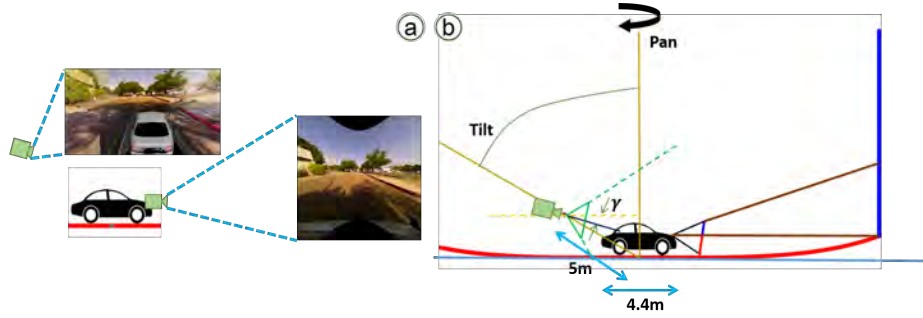


Fig. 13. (a) View interpolation. (b) Variable viewpoint.

shown in (15):

$$[x \ y \ 1]^T = \frac{1}{w} \cdot M_{int} \cdot M_{ext} \cdot [X \ Y \ Z \ 1]^T, \quad (15)$$

where $[x \ y \ 1]^T$ is the coordinate in dewarpped image, w is the homogeneous factor, M_{int} and M_{ext} are the intrinsic and extrinsic parameters of fisheye camera, respectively, and $[X \ Y \ Z \ 1]^T$ is the coordinate in 3D model.

We use four parameters to define the viewpoint, as shown in Figure 13(b). *Pan* is from $-\pi$ to π , describing the main direction specified by the driver. *Tilt* is set as $\frac{\pi}{3}$, which is the angle between the virtual camera and the normal vector of the ground. The elevation angle, γ , is set as $\frac{\pi}{12}$. The distance from the virtual camera to the centroid of curved surface is 5 meters.

5.2 Model Comparison

We compared the result images by projecting the acquired images into four different 3D models. The equation of these models are:

Model 1: Ground plane and cylinder surface:

$$\begin{cases} z = 0, & \text{while } x^2 + y^2 < d^2 \\ z \geq 0, & \text{while } x^2 + y^2 = d^2 \end{cases} \quad (16)$$

Model 2: Second degree paraboloid:

$$z = \frac{(x^2 + y^2)}{a^2} \quad (17)$$

Model 3: Fourth degree paraboloid:

$$z = \frac{(x^4 + y^4)}{a^4} \quad (18)$$



Fig. 14. The results using different 3D models. (a) Acquired image from fisheye camera. (b) After homography transformation. (c) Use ground plane and cylinder model. (d) Use second degree paraboloid. (e) Use fourth degree paraboloid. (f) Use hybrid model.

Model 4: Hybrid model:

$$\begin{cases} z = \frac{(x^4+y^4)}{a^4}, & \text{while } x^4 + y^4 < c^4 \\ z \geq \frac{(x^4+y^4)}{a^4}, & \text{while } x^4 + y^4 = c^4 \end{cases}, \quad (19)$$

where a , c and d are the coefficients of the paraboloid, columnar and cylinder. Figure 14 shows an example of comparison results. After image undistorted, we project the images into ground plane by using homography matrix, as shown in Figure 14(b). Distortion is large for tall objects like trees and buildings. Another problem is that the scene above the vanishing point does not appear on the ground plane. Therefore we project the acquired images onto a 3D surface to reduce the distortions. In Model 1 (Figure 14(c)), because the ground plane and cylinder is perpendicular, objects bend severely in the image. Although scene looks smooth in Models 2 and 3 (Figures 14(d)(e)), there is expansion effect between surroundings.

There are two reasons that we choose the hybrid model as our 3D model. First, the columnar surface makes the view more realistic in four directions. Second, the smoothness between fourth degree paraboloid and columnar is better. As shown in Figure 14(f), better result with less distortion is achieved by using the hybrid model.

5.3 Lookup Table

Instead of warping and stitching the whole images for each frame, we use a lookup table approach to decrease the computational cost. According to the

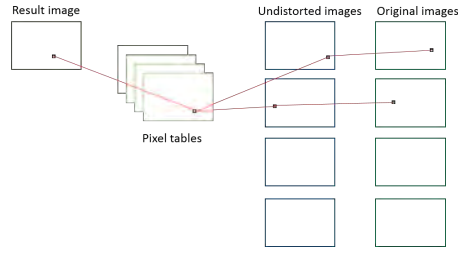


Fig. 15. Structure of the lookup table.

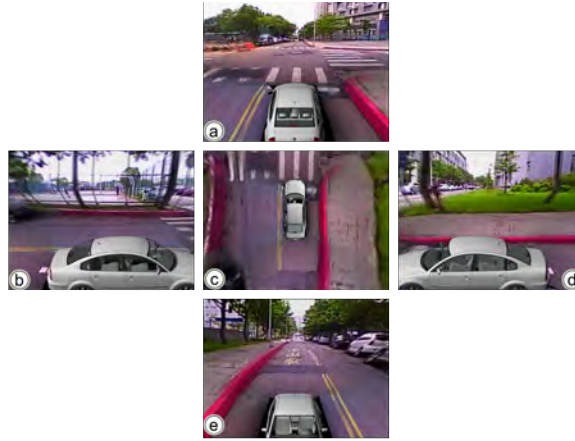


Fig. 16. The results of different viewpoints. (a) Front view. (b) Left view. (c) Top view. (d) Right view. (e) Rear view.

look-up table, we can get the pixel value of output image from the original fisheye camera images by at most seven add operations and six multiplication operations.

Each entry of the lookup table contains three data values and the information of each pixel can be retrieved from single camera frame. One is the camera ID and the other two are the coordinates (x, y) in that image. For those pixels in the overlapping area, there will be seven data values in the lookup table. They are two coordinates of two different images and one linear blending weighting factor. The structure of the lookup table is shown in Figure 15.

6 Experimental Results and Discussion

Figure 16 shows the result of our system in different viewpoint. We developed our system on a PC with an Intel Xeon 3.3G Hz CPU and 16G RAM. The software was developed with Visual Studio 2010, Release mode with full code optimization. In our experiments, four fisheye cameras were mounted on a car



Fig. 17. Ghost effect and image distortion. The color line shows the seam between paraboloid model and columnar model, and the blue line shows the overlapping area. (a) Ghost effect. (b) Image distortion.

and we tested our system in a campus and recorded about 300 seconds. The computation time of each frame was 0.7 seconds. Drivers can check every side of the vehicle carefully. By using this system, it will be safer to drive through a small lane, change lane in the highway, turn right or left in the intersection, and backward in a narrow area.

The major problem in the Fujitsu system is that the object may disappear in one view and show up again in another view because this system shrink the overlapping area between two neighbouring cameras. To tackle this problem, we choose the wide angle fisheye camera to enlarge the overlapping area. Inside the overlapping area, the images of objects appear in two different camera image simultaneously. Due to the unknown depth of objects, the same person in different cameras may be projected to different positions and cause the ghost effect in our system, as shown in Figure 17(a). Because we use one static 3D model for all vehicle surrounding scenes, no matter for the passing pedestrians or the trees in 15 meters away, we can project them onto the same 3D model and may cause distortions as shown in Figure 17(b). We need a 3D model which can adjust its shape according to the 3D positions of surroundings to solve these problems in the future.

7 Conclusions

In this work, we have developed a real-time vehicle surrounding monitoring system. It can provide a realistic and intuitive surrounding scene for drivers. To construct our system, a simple and precise method for fisheye camera calibration and image undistortion are proposed. Furthermore, a novel hyper projection model, which contains a paraboloid surface and a columnar surface, is used. It makes the final rendered view look more realistic. With this monitoring system, driver can use the third-person viewpoint to watch the surroundings of her vehicle. The proposed system can provide the drivers with good comprehension of the surrounding situations and reduce the risk of car accidents.

Acknowledgement. This work was supported in part by Industrial Technology Research Institute, Taiwan, with Grants 102-S-C20 and D301AR3R30.

References

1. Ehlgen, T., Pajdla, T.: Monitoring surrounding areas of truck-trailer combinations. In: Proceedings of the 5th International Conference on Computer Vision Systems. (2007)
2. Liu, Y.C., Lin, K.Y., Chen, Y.S.: Birds-eye view vision system for vehicle surrounding monitoring. In: Robot Vision. Springer (2008) 207–218
3. NISSAN: Around View Monitor TECHNOLOGICAL DEVELOPMENT ACTIVITIES. (<http://www.nissan-global.com/EN/TECHNOLOGY/OVERVIEW/avm.html>)
4. LUXGEN: Eagle View+ 360 surround imaging system. (http://www.luxgen-motors.vn/module-news-page-tech-cat_id-5-techid-14-lang-en.html)
5. HONDA: Honda Develops New Multi-View Camera System to Provide View of Surrounding Areas to Support Comfortable and Safe Driving. (<http://world.honda.com/news/2008/4080918Multi-View-Camera-System/>)
6. Yu, M., Ma, G.: 360 surround view system with parking guidance. Technical report, SAE Technical Paper (2014)
7. Fujitsu: 360 Wrap-Around Video Imaging Technology. (<http://www.fujitsu.com/us/semiconductors/gdc/products/omni.html>)
8. Zhang, Z.: A flexible new technique for camera calibration. Pattern Analysis and Machine Intelligence, IEEE Transactions on **22** (2000) 1330–1334
9. Weng, J., Cohen, P., Herniou, M.: Camera calibration with distortion models and accuracy evaluation. IEEE Transactions on pattern analysis and machine intelligence **14** (1992) 965–980
10. Slama, C.C., Theurer, C., Henriksen, S.W., et al.: Manual of photogrammetry. Number Ed. 4. American Society of photogrammetry (1980)
11. Duane, C.B.: Close-range camera calibration. Photogramm. Eng **37** (1971) 855–866
12. Faig, W.: Calibration of close-range photogrammetric systems: Mathematical formulation. Photogrammetric engineering and remote sensing **41** (1975)
13. Devernay, F., Faugeras, O.: Straight lines have to be straight. Machine vision and applications **13** (2001) 14–24
14. Fitzgibbon, A.W., Fisher, R.B., et al.: A buyer’s guide to conic fitting. DAI Research paper (1996)
15. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24** (1981) 381–395
16. Chen, S.E., Williams, L.: View interpolation for image synthesis. In: Proceedings of the 20th annual conference on Computer graphics and interactive techniques, ACM (1993) 279–288