

# A Vehicle-Mounted Multi-camera 3D Panoramic Imaging Algorithm Based on Ship-Shaped Model

Xin Wang, Chunyu Lin<sup>(✉)</sup>, Yi Gao, Yaru Li, Shikui Wei,  
and Yao Zhao

Institute of Information Science, Beijing Jiaotong University,  
No. 3 Shangyuncun, Haidian District, Beijing, People's Republic of China  
cylin@bjtu.edu.cn

**Abstract.** 3D panoramic driving system provides real-time monitoring of a vehicle's surrounding. It allows drivers to drive more safely without vision blind area. This system captures images of the surroundings by the cameras mounted on the vehicle, maps and stitches the collected images as textures to a 3D model. In this process, the key tasks are to construct a 3D surface model and to design an efficient and accurate texture mapping algorithm. This paper presents a ship-shaped 3D surface model with less distortion and better visual effects. Based on the ship-shaped model, a texture mapping algorithm is proposed, which can obtain the mapping relation between the corrected image and the 3D surface model indirectly by setting up a "virtual imaging plane". The texture mapping algorithm is accurate and runs fast. Finally, this paper uses an improved weighted average image fusion algorithm to eliminate the splicing traces. Experiments show that the proposed algorithm based on ship-shaped model has better 3D panoramic effect.

**Keywords:** Driver assistance systems · Ship-shaped model  
Panoramic imaging · Image fusion

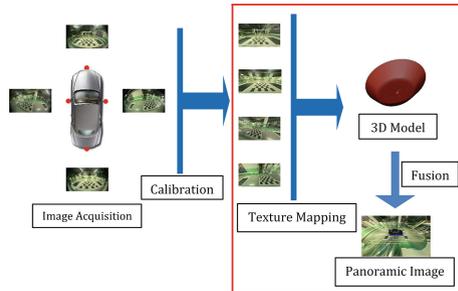
## 1 Introduction

Classical 2D panoramic driver assistance system provides a bird's-eye view. This system uses six fisheye cameras to synthesize a seamless bird's-eye view image of a vehicle's surrounding [1]. An improved bird's-eye view vision system using four fish-eye cameras is proposed in [2]. But this bird's-eye view vision system has some problems, such as the distortions of non-ground-level objects [1] and limited view scope. 3D panoramic imaging technology can make up the above deficiencies of 2D panoramic system. Many papers have proposed this 3D panoramic imaging technology, which models the surrounding of a vehicle, and maps the corrected images collected by the cameras as textures to the 3D model to generate 3D panoramic image [3–5]. They all propose different models and texture mapping algorithms, but none of them is perfect. For example, there are unnecessary distortion in the bowl model [3] proposed by Fujitsu and the texture mapping algorithm proposed in [5] is only suitable for the model similar to sphere.

Through the study of panorama modeling, image texture mapping and image fusion algorithm, this paper presents a vehicle-mounted multi-camera 3D panoramic imaging algorithm based on a ship-shaped model. In this algorithm, we establish a more reasonable 3D surface model, propose a new algorithm to obtain the texture mapping relation between the corrected images and 3D surface model, and adopt a simple and effective fusion algorithm to eliminate the splicing traces. The algorithm proposed in this paper has the following advantages. Firstly, the ship-shaped model has less distortion and better visual effect. Secondly, the texture mapping algorithm runs fast. Thirdly, the image stitching area is seamless.

## 2 Process Overview

In general, in order to capture images around a vehicle, four wide-angle cameras facing four different directions should be installed on the vehicle. Because of the distortion of the wide-angle cameras, calibration is needed to obtain intrinsic parameters and extrinsic parameters [6]. With the intrinsic and extrinsic parameters, the images captured by the wide-angle cameras can be corrected as undistorted images, which are called as corrected images in this paper. Because the wide-angle cameras have no sensors to record the depth of field, it is necessary to establish a virtual 3D surface model to simulate the three-dimensional driving environment. After the images captured by the four cameras is corrected, they are projected onto the 3D surface model according to the texture mapping relationship, and the 3D panoramic image is generated through image fusion. Therefore, the process of 3D panoramic imaging technology is roughly shown in Fig. 1.



**Fig. 1.** The process of 3D panoramic imaging technology

This paper will focus on 3D surface modeling, texture mapping algorithm, as well as image fusion. Hence, we assume that image acquisition and calibration are already ready.

### 3 3D Surface Modeling

3D surface modeling is to build a 3D surface to which the corrected images as texture is mapped to generate 3D panoramic image. Fujitsu first proposed a bowl model [3], while four different models are compared in [4]. A mesh three-dimensional model is presented in [5]. Through the research of the city road environment, this paper presents a ship-shaped model with low distortion and good visual effect.

#### 3.1 Ship-Shaped Surface Model

According to the actual driving environment, this paper presents a 3D surface model like a ship, as shown in Fig. 2. The bottom of the model is elliptical and it extends upward at a specified inclination.

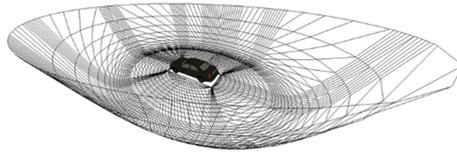


Fig. 2. Ship-shaped surface model

The general equation of the model is given as follows. Let  $(x, y, z)$  be a point on the surface model, then the equation for the bottom of the ship model is given in (1).

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} \leq 1 \quad \text{and} \quad z = 0 \tag{1}$$

Where  $a$  and  $b$  are the values of the semi-major axis and semi-minor axis of the ellipse, whose values are generally based on the length and width of the vehicle, the aspect ratio of the control panel in the driving system, and the actual road conditions.

The equation of the slope of the ship model is given in (2).

$$\frac{x^2}{(\sigma(z) + 1)^2 a^2} + \frac{y^2}{(\sigma(z) + 1)^2 b^2} = 1 \tag{2}$$

The equation  $\sigma(z)$  defines the slope of the model, which should be set elaborately to achieve the best visual effects. For example, suppose  $f(x) = 4x^2$  is the slope equation, which is the curve function of the section of 3D model on the  $xoz$  surface, as shown in Fig. 3.

The advantages of the model are as follows. There is a curvature between the bottom plane and the slope, so that the transition between the bottom and the slope is smooth. With elliptical structure increasing the projected area of the images captured by the left and right cameras on the model, it reduces the distortion and is more in line

with the human vision experience, especially when displaying the 3D object on left or right side of the car, it looks more real.

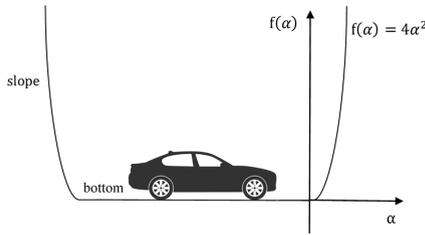


Fig. 3. The sectional view of ship-shaped model

### 4 The Texture Mapping Algorithm

Texture mapping maps the source image as texture onto a surface in 3D object space [7]. The main goal of the texture mapping is to determine the correspondence relation between the spatial coordinates of three-dimensional model and two-dimensional texture plane coordinates. A texture mapping algorithm based on the principle of equal-area is proposed in [5], which is applicable to the model similar with sphere. We present a texture mapping algorithm to obtain the mapping relation between the corrected images and the 3D surface model by setting up a virtual imaging plane. The algorithm is applicable to any shape of 3D surface model, including the ship model. Firstly, the pixel mapping relationship between the 3D surface model and the virtual imaging plane is set up. Then, the pixel mapping relationship between the virtual imaging plane and the corrected image is derived. Finally, the texture mapping relationship of the 3D surface model and the corrected images is obtained by combining the above two steps.

#### 4.1 The Mapping Relation Between the 3D Model and the Virtual Imaging Plane

As shown in Fig. 4, a virtual imaging plane is set between the camera and the 3D model. For any point P (x, y, z) on the 3D model, the corresponding point P'(x', y', z') can be derived from the perspective projection model [8].

Perspective projection is the process where the three-dimensional objects within the field of view of the camera are projected to the camera imaging plane. In general, the shape of the perspective projection model is a pyramid with the camera as the vertex, and the pyramid is cut by two planes. The plane near the camera is called near plane, and the far is called far plane. Only the objects between the two planes is visible to the camera. In the so-called perspective projection, the three-dimensional objects between the two planes are projected to the near plane. This projection relationship can be expressed by a four-dimensional matrix, called a perspective projection matrix. A commonly used perspective projection matrix is shown in (3).

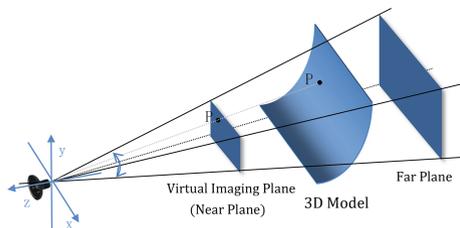


Fig. 4. Perspective model of 3D model

$$T = \begin{bmatrix} \frac{1}{\tan(\frac{\theta}{2}) \cdot \eta} & 0 & 0 & 0 \\ 0 & \frac{1}{\tan(\frac{\theta}{2})} & 0 & 0 \\ 0 & 0 & -\frac{f+n}{f-n} & -\frac{2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{bmatrix} \quad (3)$$

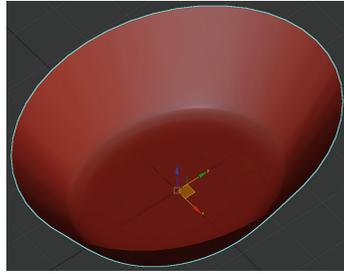
Where  $\theta$  is the maximum angle of the camera’s field of view,  $\eta$  is the aspect ratio of the projection plane (i.e., near plane),  $f$  is the distance between the far plane and the camera, and  $n$  is the distance between the near plane and the camera.

Taking an arbitrary camera on the car as example, a virtual imaging plane is set between the camera and the 3D model. The process of projecting the pixels from the 3D model to the virtual imaging plane can be regarded as a perspective projection. For any point on the 3D model, the corresponding point on the virtual imaging plane can be obtained by the perspective projection matrix.

Taking the front camera as example, establish three-dimensional coordinate system with the camera as the origin, as shown in Fig. 4. Assuming that the maximum angle of the camera’s field of view is  $\theta_0$ , the distance between the virtual imaging plane and the camera is  $n_0$ , the aspect ratio of the virtual imaging plane is  $\mu_0$ , the distance between the far plane and the camera is  $f_0$ , point  $P(x, y, z)$  is on the 3D model, and the corresponding point  $P'(x', y', z')$  is on the virtual imaging plane, then the correspondence between  $P$  and  $P'$  is given in (4).

$$\begin{bmatrix} x' \\ y' \\ z' \\ w' \end{bmatrix} = \begin{bmatrix} \frac{1}{\tan(\frac{\theta_0}{2}) \cdot \eta_0} & 0 & 0 & 0 \\ 0 & \frac{1}{\tan(\frac{\theta_0}{2})} & 0 & 0 \\ 0 & 0 & -\frac{f_0+n_0}{f_0-n_0} & -\frac{2f_0n_0}{f_0-n_0} \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (4)$$

Similarly, other three cameras also has the corresponding relationship as given in (4). Therefore, Eq. (4) can be extended as Eq. (5).



**Fig. 5.** Coordinate system  $O_{center}$

$$\begin{bmatrix} x'_i \\ y'_i \\ z'_i \\ w'_i \end{bmatrix} = \begin{bmatrix} \frac{1}{\tan(\frac{\theta_i}{2}) \cdot \eta_i} & 0 & 0 & 0 \\ 0 & \frac{1}{\tan(\frac{\theta_i}{2})} & 0 & 0 \\ 0 & 0 & -\frac{f_i + n_i}{f_i - n_i} & -\frac{2f_i n_i}{f_i - n_i} \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} \quad (5)$$

where  $i = 0, 1, 2, 3$ , is respectively denotes the front camera, rear camera, left camera and right camera.  $(x'_i, y'_i, z'_i)$  and  $(x_i, y_i, z_i)$  are the positions in the coordinate system with the camera corresponding to  $i$  as coordinate origin.

Equation (5) corresponds to four different coordinate systems. In order to unify the coordinate system, as shown in Fig. 5, we convert the points in the coordinate system denoted by  $O_{center}$  into those in the corresponding camera coordinate system.

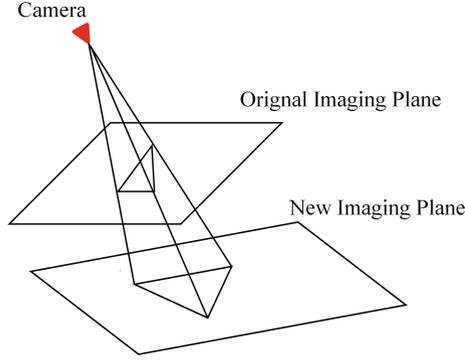
The process of converting the points in the coordinate system  $O_{center}$  to the corresponding point in the corresponding camera coordinate system is to make the points rotate  $\beta_i^o$  along the  $z$  axis, then rotate  $\beta_i^o$  along the  $x$  axis, and finally move along the translation vector denoted by  $\vec{F}_i = (X_i, Y_i, Z_i)$ .  $\alpha_i^o$  is the angle between the camera and the horizontal plane,  $\beta_i^o$  is the angle between the camera and  $x$  axis,  $(X_i, Y_i, Z_i)$  is the coordinate in the coordinate system  $O_{center}$ ,  $\alpha_i^o$  and  $(X_i, Y_i, Z_i)$  are related to the position and orientation of the camera on the car, the values of  $\beta_i^o$  are  $0^\circ, 180^\circ, 270^\circ$ , and  $90^\circ$ . The values of  $i$  is 0, 1, 2, 3 and they respectively correspond to the front camera, rear camera, left camera and right camera.

According to computer graphics knowledge [8], the above rotation relations can be expressed by the rotation matrices  $R_z(\beta_i^o)$  and  $R_x(\alpha_i^o)$ , and the translation relation can

be expressed by the translation matrix  $T_{(X_i, Y_i, Z_i)} \cdot R_z(\beta_i^o)$  is  $\begin{bmatrix} \cos \beta_i^o & \sin \beta_i^o & 0 & 0 \\ -\sin \beta_i^o & \cos \beta_i^o & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ ,

$R_x(\alpha_i^o)$  is  $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha_i^o & \sin \alpha_i^o & 0 \\ 0 & -\sin \alpha_i^o & \cos \alpha_i^o & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ , and  $T_{(X_i, Y_i, Z_i)}$  is  $\begin{bmatrix} 1 & 0 & 0 & X_i \\ 0 & 1 & 0 & Y_i \\ 0 & 0 & 1 & Z_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$ . Combining

Eq. (5) with the above rotation and translation matrices, the mapping relation between



**Fig. 6.** Perspective transformation

the pixel points on the 3D surface model and the pixel points on the virtual imaging plane is given in (6).

$$\begin{bmatrix} X'_i \\ Y'_i \\ Z'_i \\ W'_i \end{bmatrix} = R_Z(\beta_i^o)R_X(\alpha_i^o)T(X_i, Y_i, Z_i) \begin{bmatrix} \frac{1}{\tan(\frac{\theta_i}{2}) \cdot \eta_i} & 0 & 0 & 0 \\ 0 & \frac{1}{\tan(\frac{\theta_i}{2})} & 0 & 0 \\ 0 & 0 & -\frac{f_i + n_i}{f_i - n_i} & -\frac{2f_i n_i}{f_i - n_i} \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix} \quad (6)$$

#### 4.2 The Mapping Relation Between the Corrected Image and the Virtual Imaging Plane

The corrected image and the image on the virtual imaging plane can be viewed as images of the camera in two different planes as shown in Fig. 6, which is also known as perspective transformation [9].

In general, the pixel mapping relation of the perspective transformation can be expressed by the following formula (7)

$$\begin{cases} x = \frac{a_{11}u + a_{12}v + a_{13}}{a_{31}u + a_{32}v + 1} \\ y = \frac{a_{21}u + a_{22}v + a_{23}}{a_{31}u + a_{32}v + 1} \end{cases} \quad (7)$$

where  $(u, v)$  is the pixel coordinate in the original imaging plane and  $(x, y)$  is the pixel coordinate in the new imaging plane.

It can be seen from the Eq. (7) that the pixel mapping relationship of the perspective transformations contains eight unknown parameters. Therefore, four points on the original imaging plane and their corresponding points on the new imaging plane are required to calculate the unknown parameters.

As shown in Fig. 7, we make the position of the calibration boards fixed. Therefore, the position on the 3D model of the 16 feature points, which are on the calibration boards, can be determined when designing the 3D model. Assuming that the

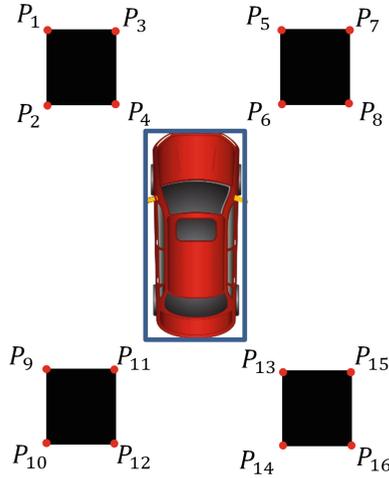


Fig. 7. Calibration board layout

coordinates of the feature point  $P_1$  on the 3D model are  $(x_1, y_1, z_1)$ , the coordinates of  $P_1$  on the virtual imaging plane of the front camera are  $(x'_1, y'_1)$  by Eq. (6). The pixel coordinates of  $P_1$  in the corrected image of the front camera can be obtained by feature point detection [10], which is  $(x''_1, y''_1)$ . The same procedure can be easily adapted to obtain the coordinates of these feature points which are  $P_2, P_7$  and  $P_8$  both on the virtual imaging plane and in the corrected image. Using these four points ( $P_1, P_2, P_7$  and  $P_8$ ), the pixel mapping relationship between the front corrected image and its virtual imaging plane can be obtained. The same procedure applies to the other three cameras.

### 4.3 Texture Mapping Between the Corrected Images and the 3D Model

Through the above two steps, the texture mapping relation between the 3D model and the corrected images can be obtained. When generating 3D panoramic image in real time, the delay will be large if calculating each pixel value on the 3D model through the mapping relation. In order to reduce delay, as shown in Fig. 8, the coordinates of the pixels on the 3D model and it in the corrected image can be stored in the memory in one-to-one corresponding order when the system initialized. When the 3D panoramic image is generated in real time, the corresponding texture coordinates of 3D model can be read from the memory where the texture coordinates are stored, and then the texture on the corrected images is mapped to the 3D surface model to generate a 3D panoramic image.

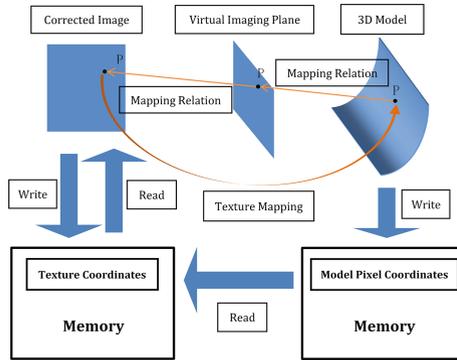


Fig. 8. The process of texture mapping

## 5 Image Fusion

If the panoramic image generated by texture mapping has no fusion processing, there will be a “ghosting” phenomenon in the overlapping areas, as shown in Fig. 9. Therefore, it is necessary to fuse the overlapping areas of the panoramic image.

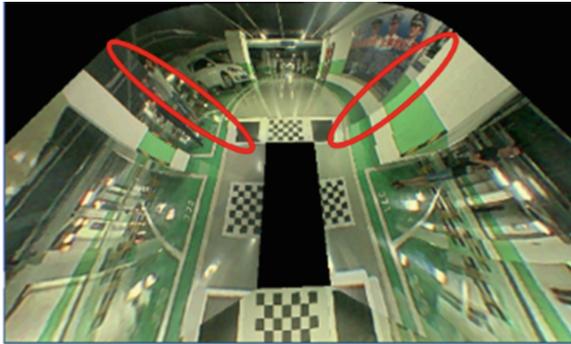


Fig. 9. The panoramic image before image fusion

In this paper, we improve the weighted average method [11] so that the weight of the texture of the corrected images is proportional to the distance from the texture coordinate to the mapping boundary of the corrected image. This method is simple and has good fusion effect.

In the example of the overlap between the front and the right, as shown in Fig. 10, the overlapping area has two boundary lines  $L$  and  $M$ ,  $L$  is the mapping boundary of the front corrected image,  $M$  is the mapping boundary of the right corrected image.  $W_{half}$  is half of the car width, and  $L_{half}$  is half of the car length. Here, the distance from the texture coordinate to the mapping boundary can be expressed by its angle relative to

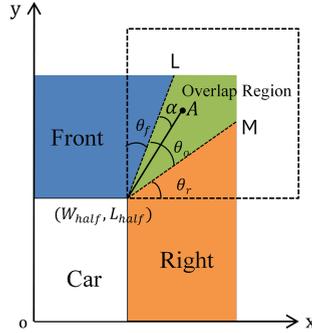


Fig. 10. Image fusion diagram

the mapping boundary. Therefore, according to the fusion algorithm, the texture of the point  $A(x, y, z)$  is  $C_A$ , which can be obtained by Eq. (8).

$$C_A = w_{front}(\alpha) \cdot C_{front} + w_{right}(1 - \alpha) \cdot C_{right} \tag{8}$$

Where  $C_{front}$  and  $C_{right}$  represent the texture of the corrected images,  $w_{front}$  and  $w_{right}$  are fusion weight,  $w_{front}(\alpha) = \frac{\alpha - \theta_r}{\theta_o}$ ,  $w_{right}(1 - \alpha) = 1 - w_{front}(\alpha)$ , and  $\alpha = \arctan\left(\frac{|y| - w_{half}}{|x| - l_{half}}\right)$ .

## 6 Experimental Results and Discussion

Using the fusion algorithm described in Sect. 5, the panoramic image obtained by the processing flow described in Sect. 2 is shown in Fig. 11.

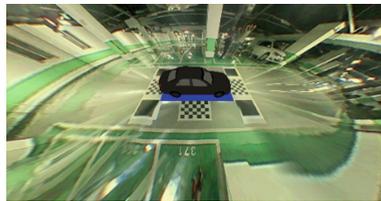


Fig. 11. Panoramic image

### 6.1 Effect of Image Fusion

Figure 12(a) is part of the fusion area that did not use the fusion algorithm, and Fig. 12(b) is part of the area that uses the fusion algorithm. It can be seen from the figure that the fusion algorithm adopted in this paper basically eliminates the seam of the stitching position.



**Fig. 12.** Comparison of the effect before and after using fusion algorithm

## 6.2 Comparison of the Effect of Different Models

At present, the earliest proposed 3D surface model is the bowl-shaped model proposed by Fujitsu, which is shown in [12]. Compared with the bowl-shaped model, the ship-shaped model proposed in this paper has the advantages of less distortion and better visual effect because of considering the driving environment and the visual experience.

At first, the ship-shaped structure of the 3D model proposed in this paper can eliminate unnecessary distortion. Paying attention to the lane line of the highway in Fig. 13, the lane line becomes a broken line in Fig. 13(a), while it is almost a straight line in Fig. 13(b).



**Fig. 13.** Comparison of the distortion between bowl-shaped model and ship-shaped model: (a) the image of the bowl-shaped model and (b) the image of the ship-shaped model.

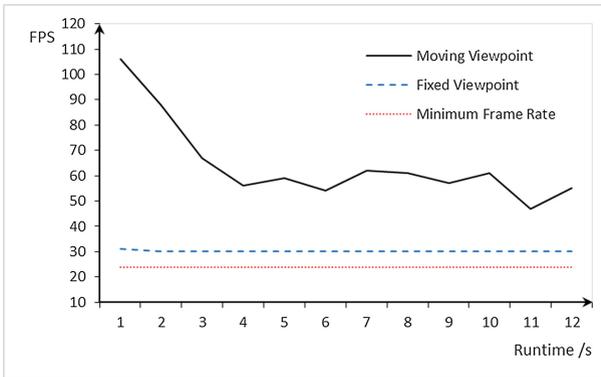
In addition, the ship-shaped model can simulate the driving environment better and has better visual effects. As shown in Fig. 14, the car on the road is projected to the bottom plane of the bowl model, making people feel that the other vehicle and the vehicle model are not on the same pavement. Considering the road environment, the objects on both sides of the road are projected onto the 3D surface of the ship-shaped model. Therefore, the ship model can better display 3D objects on both sides of the road, which reduces the driver's misjudgment.



**Fig. 14.** Comparison of the visual effect between bowl-shaped model and ship-shaped model: (a) the image of the bowl-shaped model and (b) the image of the ship-shaped model.

### 6.3 Performance of Texture Mapping

The above panoramic imaging algorithm is implemented by using the Linux operating system based on kernel version 3.10 and the hardware platform of the i.mx6Q series cortex-A9 architecture 4-core processor. On the experimental platform, the frame rate of the panoramic imaging algorithm is tested in both fixed viewpoint and moving viewpoint respectively, and both frame rates meet the minimum requirements, as is shown in Fig. 15. The experimental results show that the texture mapping algorithm proposed in this paper has high performance and can meet the requirements of real-time display of vehicle display equipment.



**Fig. 15.** Results of texture mapping performance test

## 7 Conclusion

This paper presents a 3D panoramic imaging algorithm. The contributions mainly includes three aspects: 3D surface modeling, texture mapping algorithm and image fusion. In the aspect of 3D modeling, this paper proposes a ship-shaped model that has

low distortion and better visual effect. In the aspect of texture mapping algorithm, this paper presents a method to calculate the texture mapping relation by setting up a virtual image plane between the camera and the 3D model. Because the mapping relation are stored in memory after initial run, the algorithm runs fast and meets the requirements of real-time imaging. In the aspect of image fusion, this paper adopts a special weighted average image fusion algorithm, which effectively eliminates the splicing traces of the fusion region. The experimental results prove the feasibility and effectiveness of the proposed algorithm.

**Acknowledgment.** This work was supported in part by National Natural Science Foundation of China (No. 61402034, Nos. 61210006 and 61501379), and the Fundamental Research Funds for the Central Universities (2017JBZ108).

## References

1. Liu, Y.-C., Lin, K.-Y., Chen, Y.-S.: Bird's-eye view vision system for vehicle surrounding monitoring. In: Sommer, G., Klette, R. (eds.) *RobVis 2008*. LNCS, vol. 4931, pp. 207–218. Springer, Heidelberg (2008). [https://doi.org/10.1007/978-3-540-78157-8\\_16](https://doi.org/10.1007/978-3-540-78157-8_16)
2. Yu, M., Ma, G.: 360 surround view system with parking guidance. *SAE Int. J. Commercial Veh.* **7**(2014-01-0157), 19–24 (2014)
3. Huebner, K., Natroshvili, K., Quast, J., et al.: Vehicle Surround View System: U.S. Patent Application 13/446,613[P], 13 April 2012
4. Yeh, Y.-T., Peng, C.-K., Chen, K.-W., Chen, Y.-S., Hung, Y.-P.: Driver assistance system providing an intuitive perspective view of vehicle surrounding. In: Jawahar, C.V., Shan, S. (eds.) *ACCV 2014*. LNCS, vol. 9009, pp. 403–417. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-16631-5\\_30](https://doi.org/10.1007/978-3-319-16631-5_30)
5. Liu, Z., Zhang, C., Huang, D.: 3D model and texture mapping algorithm for vehicle-mounted panorama system. *Comput. Eng. Design* **38**(1), 172–176 (2017). (in Chinese)
6. Zhang Z.: A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(11), 1330–1334 (2000)
7. Heckbert, P.S.: Survey of texture mapping. *IEEE Comput. Graphics Appl.* **6**(11), 56–67 (1986)
8. Angel, E.: Interactive computer graphics: a top-down approach with OpenGL, with OpenGL primer package, 2/E. In: *DBLP* (1997)
9. Bradski, G., Kaehler, A.: *Learning OpenCV: Computer Vision with the OpenCV library*. O'Reilly Media, Inc., Upper Saddle River (2008)
10. Jiang, D., Yi, J.: Comparison and study of classic feature point detection algorithm. In: *International Conference on Computer Science and Service System*, pp. 2307–2309. IEEE Computer Society (2012)
11. Rani, K., Sharma, R.: Study of different image fusion algorithm. *Int. J. Emerg. Technol. Adv. Eng.* **3**(5), 288–291 (2013)
12. Wrap-Around Video Imaging Technology - Fujitsu United States. <http://www.fujitsu.com/cn/about/resources/videos/fss-video/360.html>. Accessed 28 May 2017