

# Journal of Electronic Imaging

[SPIDigitalLibrary.org/jei](http://SPIDigitalLibrary.org/jei)

## **Panorama parking assistant system with improved particle swarm optimization method**

Ruzhong Cheng  
Yong Zhao  
Zhichao Li  
Weigang Jiang  
Xin'an Wang  
Yong Xu



# Panorama parking assistant system with improved particle swarm optimization method

Ruzhong Cheng  
Yong Zhao  
Zhichao Li  
Weigang Jiang  
Xin'an Wang  
Yong Xu

Mobile Video Networking Technology Research Center  
PKU Shenzhen  
Guangdong Province 518055, China  
E-mail: [yongzhao@pkusz.edu.cn](mailto:yongzhao@pkusz.edu.cn)

---

**Abstract.** A panorama parking assistant system (PPAS) for the automobile aftermarket together with a practical improved particle swarm optimization method (IPSO) are proposed in this paper. In the PPAS system, four fisheye cameras are installed in the vehicle with different views, and four channels of video frames captured by the cameras are processed as a 360-deg top-view image around the vehicle. Besides the embedded design of PPAS, the key problem for image distortion correction and mosaicking is the efficiency of parameter optimization in the process of camera calibration. In order to address this problem, an IPSO method is proposed. Compared with other parameter optimization methods, the proposed method allows a certain range of dynamic change for the intrinsic and extrinsic parameters, and can exploit only one reference image to complete all of the optimization; therefore, the efficiency of the whole camera calibration is increased. The PPAS is commercially available, and the IPSO method is a highly practical way to increase the efficiency of the installation and the calibration of PPAS in automobile 4S shops. © The Authors. Published by SPIE under a Creative Commons Attribution 3.0 Unported License. Distribution or reproduction of this work in whole or in part requires full attribution of the original publication, including its DOI. [DOI: [10.1117/1.JEI.22.4.041123](https://doi.org/10.1117/1.JEI.22.4.041123)]

---

## 1 Introduction

The automobile is an essential transportation means of modern life; it has become a routine means of transportation for many middle-class families. Meanwhile, there is a growing trend of accidents related to the vehicles, and parking accidents involving young children are one of the most serious accidents among them. Besides the inadequate experience of drivers, multiple occluded views from the driver's perspective are the major causes of parking accidents.

There are many occasions in which young children might be at the front or back of the vehicle without being noticed by drivers. While similar tragedies happen from time to time due to the occluded views of drivers, it is believed that some practical technologies can reduce these kinds of accidents. In particular, a driver assistance system that is targeted

to reduce such accidents and increase parking safety can be considered.

Protecting people's safety inside and outside the vehicle is the key function of a future intelligent transportation system (ITS).<sup>1</sup> In addition to some other functions of an ITS, such as a pedestrian detection system and some other systems that can dynamically capture vehicle surroundings,<sup>2</sup> the panorama parking assistant system (PPAS) provides safety functionality for parking purposes by intuitive dynamic top-view images. It combines video, audio, and radar signals to inform the driver if there are people or obstacles around the vehicle, especially for the occluded zone of the automobile. Therefore, the efficiency and ease of parking are increased, with parking accidents also being avoided.

### 1.1 State-of-the-Art Technology

Two kinds of parking assistant systems are prevailing in the market. One is parking distance control (PDC), which applies ultrasound transducers, and the other is the video assistant parking system (VAPS), which applies camera sensors.

PDC usually integrates ultrasound transducers into the rear bumpers of automobiles. It measures the distance to the nearest object behind the vehicle and outputs a progressive acoustic warning signal to inform the driver. PDC is able to detect obstacles within 2 m by using ultrasound transducers. However, the exact location, size, and type of obstacles cannot be determined. When the distance between obstacles and the vehicle is below half a meter, the performance of PDC is poor. Although PDC has been well developed by manufacturers in recent years, the main drawback still remains.

A VAPS, which mixes ultrasound transducers with camera sensors, was, therefore, designed and built to overcome the disadvantages of PDC. The system integrates a camera in the rear of the car. When the ultrasound sensor detects an obstacle that is close to the reversing vehicle, the driver can clearly identify what type of obstacle it is. VAPS is limited to visualizing the situation of the rear, and it does not completely solve the occlusion problems that happen at the sides and at the front of the vehicle during parking.

---

Paper 13185SS received Apr. 11, 2013; revised manuscript received Sep. 20, 2013; accepted for publication Sep. 27, 2013; published online Nov. 14, 2013.

Thus, the risk of having parking accidents still exists, even though it is reduced.

A panorama parking system called the around-view monitor and multiview camera system can overcome all the disadvantages of PDC and VAPS; they were first introduced by Nissan together with Xanavi and SONY, and they have been applied in several premium-line vehicle models.

Generally, all panorama parking systems make use of four wide-angle high-resolution cameras. These cameras are installed at the front, the rear, and both sides of a vehicle. Video frames from all these four cameras will be captured and processed. A top-view image will be shown to the driver after processing, allowing the driver to recognize the scenarios of all directions. Because of its high display accuracy that gives *in situ* views around the car, drivers can park confidently without looking around the car, just looking at the display screen. It not only solves the safety problem, but also makes people drive more easily, even in a complex situation in the parking area or garage. It provides unparalleled advantages compared with PDC and VAPS.

Present performance of panorama parking technologies still has the potential to improve. For instance, the integrity and smoothness of mosaicking images need to be refined by improved algorithms. More importantly, this technology is still very expensive, so it is only employed in original equipment manufacturer (OEM) markets to flagship and high-line vehicle models. It is hard to apply it broadly in normal cars. An after-market solution of this technology with proper price, high performance, and easy installation is necessary for utilization in budget cars, and this is the very purpose of this study.

## 1.2 Paper Outline

This paper focuses on the problem of designing an embedded PPAS, which can be applied to most passenger cars with a more accurate and efficient calibration method.

As one subsystem of the advanced driver assistant system for safety, the PPAS should also be designed for easy integration with other subsystems such as the fatigue driving detection system and pedestrian detection system. In this connection, the same system architecture will be employed to facilitate the functional integration.

The PPAS needs to be designed from software to hardware. Fisheye cameras are necessary to cover a proper wide range of views in four directions of the car. Four channels of video will be captured, processed, and stitched to form one smooth top-view image for display on the LCD screen of the automobile. Static or moving objects would be displayed, and the driver will be warned by an audio alert signal if these objects are too close to the car.

From the perspective of algorithm, there are two types of splicing in this paper's application: the first type is calibrating and optimizing the camera's parameters, so that the distorted images would be corrected and the prerequisites of routine image splicing methods become available. The second type is to use routine methods to splice the fisheye images corrected by the calibration and optimization methods, and the function of routine splicing methods such as region-based and feature-based methods is to match or splice two pictures' border. In fact, the routine splicing methods such as region-based and feature-based methods both require that the images have a certain overlapping area and the

deformation of the image cannot be too large. Furthermore, they also depend heavily on the position and angle of camera installation. In the case of this paper, all the images captured by fisheye cameras are highly distorted and the position or angle of the cameras would vary greatly when they are installed in different vehicles. So the whole splicing prerequisites in this paper are quite different from the routine splicing methods, such as panorama-photo splicing by iPhone's function.

In this paper, we mainly focus on the key problem of parameter calibration and optimization because it will decide the efficiency of installing PPAS, which is the key factor to judge the availability of PPAS's application in 4S shop. Additionally, by the parameter calibration and optimization, the four distorted images would be corrected and have one universal coordinate system, which means that after optimization, the four images could be spliced by coordinates alone.

The paper is, therefore, organized as follows. Section 2 will describe the proposed PPAS. Sections 3 and 4 will describe the general methods of calibration and distortion correction. An improved particle swarm optimization (IPSO) method to optimize calibration will also be described. Section 5 will outline the architecture of the system hardware. Section 6 will describe the implementation results of the system, and finally, Sec. 7 will give the final conclusion of this paper.

## 2 Panorama Parking Assistant System

The Market shop has a four-in-one function in the field, which includes vehicle sales (Sales), accessories supply (Spareparts), maintenance (Service), and other feedback (Survey), and it is the main channel to the after-sales market. Therefore, the purpose of this study is to build an embedded parking assistant system based on imaging technology, which should be convenient for installation and calibration in 4S shops for vehicle modification. It is expected to meet the following basic requirements.

The system can form an around-view image and display as a top view on a screen by using four cameras around the car; there is no blind spot in the synthesized top-view image, and the embedded system can be installed in normal family cars with the least installation and calibration efforts. The PPAS can be installed in any family car as long as it is calibrated for the specific car model. According to these requirements, the PPAS was designed as follows.

The PPAS has three main components. These components are digital signal processor (DSP)-embedded host, camera, and LCD display. The DSP is mounted inside the car and connected with the cameras. The LCD display is fixed on the central console so the driver can see all the surroundings on the LCD display. If the vehicle has its own LCD, the LCD can be replaced by its own set, and it can also reduce the cost of the PPAS. The four fisheye cameras are mounted around the automobile facing in four directions, as shown in Fig. 1. The round points in the figures are the positions of the cameras.

Images from the four fisheye cameras are captured and processed. The cameras' intrinsic and extrinsic parameters are evaluated, so a top-view image with world coordinates with reference to the ground plane is obtained by blending these four images together. The top-view image will be displayed on the screen in the central console. The display format is illustrated in Fig. 2.

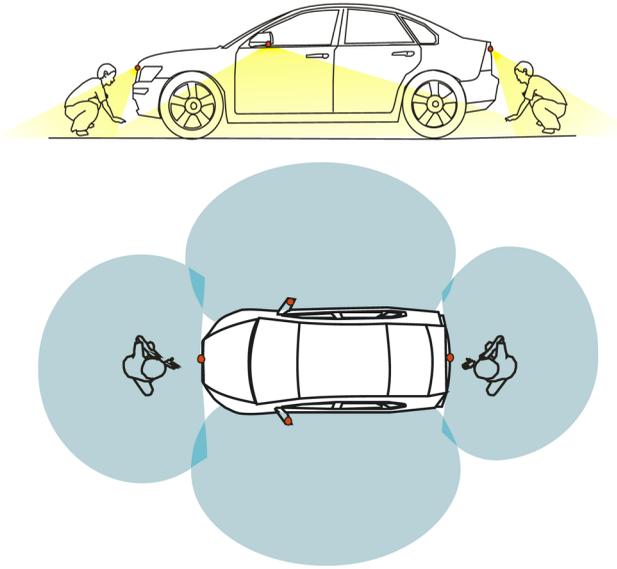


Fig. 1 Side view and top view of panorama parking assistant system (PPAS).

The left half of the screen shows the original image captured by the front camera or the rear camera. The front camera view will first be displayed when the automobile starts up, while the rear view will be activated when the vehicle is reversing. The right half shows the top-view image formed by the four calibrated and spliced images, and that area can also be shifted to the right-side view by pushing a shift button so as to see the blind spot on the other side to the driver. In the right part, there is a virtual car image at its center, and the driver can recognize the relative position around the car. All of these display arrangements can be redefined when necessary.

### 3 Camera Calibration

With respect to algorithms, two main procedures should be taken to finish the basic function of PPAS. The first procedure is camera calibration for fisheye distortion correction, and the second procedure is image mosaicking by parameter optimization.

The purpose of camera calibration is to establish a corresponding relationship between the three-dimensional (3-D) world coordinates and the two-dimensional (2-D) image coordinates so that the position in the 3-D world can be restored or estimated in the 2-D image.

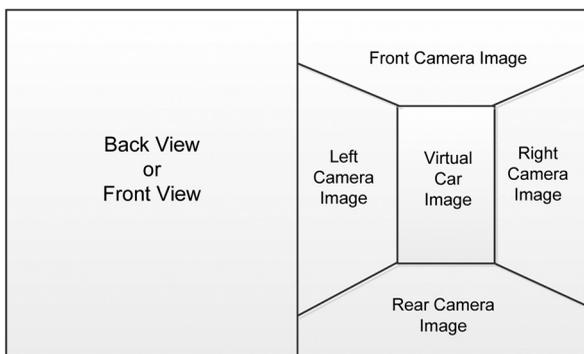


Fig. 2 Screen arrangement of PPAS.

Many camera calibration and distortion correction methods have been proposed in the last two decades.<sup>3-7</sup> The two-step method proposed by Tsai and Zhang<sup>8,9</sup> is representative and widely used, and this paper applied the two-step method to complete the parameter calibration and optimization. It mainly contains the following two steps: the first one is projection transformation for the linear solution of parameters, and the second is the nonlinear optimization of the linear calibration results. In this paper, the conventional method in the first step is applied to obtain the intrinsic parameters in this section, and an improved method will be applied in the second step to optimize the extrinsic parameter in Sec. 4.

#### 3.1 Projection Transformation

The relationship between the camera coordinate system and world coordinate system can be described by the rotation matrix  $R$  and the translation vector  $t$ . If the homogeneous coordinates of point  $P$  in the world coordinate system is  $(X_w, Y_w, Z_w, 1)^T$  and the camera coordinate system is  $(x, y, z, 1)^T$ , there will be the following relationship:

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = M_2 \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}, \quad (1)$$

where  $R$  is the orthogonal matrix, and  $t$  is the 3-D translation vector.  $M_2$  is a  $4 \times 4$  matrix.  $M_2$  is determined by the camera position related to the world coordinates, which are therefore named the extrinsic parameters of the camera.

Considering the relationship of the pixel units and the physical coordinate units, and using a pinhole imaging model and the perspective projection relations, it can be described as follows:

$$s \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}. \quad (2)$$

$(x, y, z)$  are the coordinates of point  $P$  in the camera coordinate system.  $f$  is the distance between the  $x$ - $y$  plane and the image plane, which is commonly referred to as the camera focal length;  $s$  is the scale factor.

$P$  and its projection coordinates  $p(u, v)$  can be expressed as follows:

$$\begin{aligned} s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} &= \begin{bmatrix} \frac{1}{dX} & 0 & u_0 \\ 0 & \frac{1}{dY} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} \alpha_x & 0 & u_0 & 0 \\ 0 & \alpha_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \\ &= M'_1 M_2 X_w, \end{aligned} \quad (3)$$

where  $(u, v)$  is one of the image coordinates in which the  $u$  and  $v$ -axis unit are pixels, and  $(u_0, v_0)$  is the origin of the coordinate system.

$\alpha_x$  and  $\alpha_y$  are the scale factors and known as the normalized focal length in the  $u$ -axis and  $v$ -axis.  $M_1$  is determined by  $\alpha_x, \alpha_y, u_0, v_0$ , which is associated with the cameras' intrinsic parameters.

It is generally assumed that the calibration plane is located on the plane of the world coordinate system. As mentioned in Eq. (1),  $R$  is a  $3 \times 3$  orthogonal matrix and  $r$  is the column vector of matrix  $R$ , and  $t$  is the translation vector; therefore, each point on the plane can be described as

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = M_1 \begin{bmatrix} r_1 & r_2 & r_3 & t \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ 0 \\ 1 \end{bmatrix} \\ = M_1 \begin{bmatrix} r_1 & r_2 & t \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ 1 \end{bmatrix}, \quad (4)$$

where

$$M_1 = \begin{bmatrix} \alpha_x & 0 & u_0 \\ 0 & \alpha_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The above equation shows the relationship between the point  $m = [u, v]^T$  of the image surface and  $M = [X_w, Y_w]^T$  on the plane.

Then a single mapping matrix  $H$  (Ref. 10) is established between the pixel point and the point on a plane template, as follows:

$$s\tilde{m} = M_1 \begin{bmatrix} r_1 & r_2 & t \end{bmatrix} \tilde{M} = H\tilde{M}, \quad (5)$$

where  $H = M_1[r_1, r_2, t]$ ,  $\tilde{m}$ , and  $\tilde{M}$  are the homogeneous coordinates of point  $m$  and  $M$ , and  $s$  is a nonzero scale factor. If knowing the four points in the plane template and knowing the corresponding points in the image coordinates,  $H$  can be estimated by the maximum likelihood criterion.

### 3.2 Solution for Intrinsic and Extrinsic Parameters

The constraint equation for intrinsic parameters is from Eq. (5). It is shown as follows:

$$[h_1 \ h_2 \ h_3] = M_1[r_1 \ r_2 \ t]. \quad (6)$$

Because  $R$  is the orthogonal matrix, two basic constraints of the intrinsic parameters can be deduced as

$$\begin{cases} h_1^T M_1^{-T} M_1^{-1} h_2 = 0 \\ h_1^T M_1^{-T} M_1^{-1} h_1 = h_2^T M_1^{-T} M_1^{-1} h_2 \end{cases} \quad (7)$$

In order to get a linear solution for camera intrinsic parameters, we assume

$$B = M_1^{-T} M_1^{-1}. \quad (8)$$

$B$  is a symmetric matrix and can be defined as vector  $b$ ,

$$b = [B_{11} \ B_{12} \ B_{22} \ B_{13} \ B_{23} \ B_{33}]^T. \quad (9)$$

So  $h_i^T B h_j = v_{ij}^T b$  is obtained; then revise the constraint equations into two vector equations as follows:

$$\begin{bmatrix} V_{12}^T \\ (V_{11} - V_{22})^T \end{bmatrix} b = 0, \quad (10)$$

$$v_{ij} = [h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, \\ h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3}]. \quad (11)$$

If there are  $n$  template plane images as parameters, use Eq. (10) to obtain the solutions. If  $n \geq 3$ , the eigenvector corresponding to a minimum eigenvalue of  $V$  is the solution of this equation.<sup>11</sup> Once vector  $b$  is obtained, the intrinsic parameter matrix  $M_1^{-1}$  can be obtained by the Cholesky decomposition.

Then the inverse matrix is the camera's intrinsic parameters matrix  $M_1^{-1}$ , which can deduce the solution of intrinsic parameters as follows:

$$\begin{cases} v_0 = \frac{(B_{12}B_{13} - B_{11}B_{23})}{(B_{11}B_{22} - B_{12}^2)} \\ \lambda = B_{33} - \frac{[B_{13}^2 + v_0(B_{12}B_{13} - B_{11}B_{23})]}{B_{11}} \\ \alpha_x = \sqrt{\frac{\lambda}{B_{11}}}, \alpha_y = \sqrt{\frac{\lambda B_{11}}{(B_{11}B_{22} - B_{12}^2)}} \\ c = -\frac{B_{12}\alpha_x^2\alpha_y}{\lambda}, u_0 = \frac{cv_0}{\alpha_x} - \frac{B_{13}\alpha_x^2}{\lambda} \end{cases} \quad (12)$$

The extrinsic parameters of each image that correspond to the plane template can be calculated by the following equations:

$$\begin{cases} r_1 = \lambda M_1^{-1} h_1 \\ r_2 = \lambda M_1^{-1} h_2 \\ r_3 = r_1 \times r_2 \\ t = \lambda M_1^{-1} h_3 \end{cases}, \quad (13)$$

where  $\lambda = \frac{1}{\|M_1^{-1}h_1\|} = \frac{1}{\|M_1^{-1}h_2\|}$ . They are just obtained from the rotation matrix  $R$  and the translation vector  $t$ .

### 3.3 Optimization of the Linear Calibration Results

Due to the structure and installation error of the lens, there would be some distortion in the image. In order to improve the accuracy of the calibration, considering the radial and tangential distortion of the lens, an equation was applied as follows:<sup>12,13</sup>

$$\begin{aligned} \hat{x} &= x(1 + k_1m^2 + k_2m^4) + 2p_1xy + p_2(m^2 + 2x^2) \\ \hat{y} &= y(1 + k_1m^2 + k_2m^4) + 2p_2xy + p_1(m^2 + 2x^2), \end{aligned} \quad (14)$$

where  $\hat{x}$  and  $\hat{y}$  are the ideal image coordinates calculated by a pinhole imaging model,  $(x, y)$  are the coordinates of the

actual image point,  $m^2 = x^2 + y^2$ ,  $k_1, k_2$  are the radial distortion parameters, and  $p_1, p_2$  are tangential distortion parameters.

Considering both the extrinsic and intrinsic parameters as unknown parameters and calculating them, a nonlinear equation can be obtained. Suppose there are  $n$  images that correspond to the template plane and  $m$  calibration points in the template plane, a cost function can be described as follows:

$$\min h = \sum_{i=1}^n \sum_{j=1}^m \|m_{ij} - m(A, k_1, k_2, p_1, p_2, R_i, t_i, M_j)\|^2, \quad (15)$$

where  $m_{ij}$  is the  $j$ 'th pixel of the  $i$ 'th image,  $R_i$  is the rotation matrix of the  $i$ 'th image,  $t_i$  is the translation vector of the  $i$ 'th image, and  $M_j$  is the spatial coordinates of the  $j$ 'th point.

$m(A, k_1, k_2, p_1, p_2, R_i, t_i, M_j)$  is a pixel coordinate obtained by known parameters,  $A$  is the matrix of intrinsic parameters, and  $k_1, k_2, p_1, p_2$  are distortion parameters. The optimal solution of this problem is to minimize the cost function.

The Levenberg-Marquadt algorithm is applied to solve the nonlinear least-squares problem. The initial estimated values are the results of the above linear solution.<sup>14,15</sup>

#### 4 Parameter Optimization for Image Mosaicking

The problem of image mosaicking can be considered as the task of optimizing the camera's intrinsic and extrinsic parameters. Once the parameters of the cameras have been calibrated and optimized, the four images can also be spliced together.

Compared with the intrinsic parameters, the extrinsic parameters could decide the final performance of image mosaicking. The intrinsic parameters can be obtained by the conventional method mentioned above, so the key problem of image blending lies in extrinsic parameter optimization.

The fisheye camera parameters to be optimized can be described by the equation as follows:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = f(\alpha_x, \alpha_y, u_0, v_0, r, t, k_1, k_2, p_1, p_2) \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}. \quad (16)$$

It is obtained by the equations mentioned in the paragraphs above. The original image of the world can be rebuilt by this equation. The parameters in function  $f$  can describe the features of the fisheye camera. The parameters are obtained by experiments. The rebuilt image is different from the original one.

As Eq. (15) described, the problem can be considered as finding the minimum difference of the two images by minimizing the cost function; therefore, the distorted image can be best restored like the real-world image. The whole procedure is shown as the following schematic shown in Fig. 3.

To minimize the cost function, PSO was applied to optimize the intrinsic and extrinsic parameters by using the two-step method.

#### 4.1 Particle Swarm Optimization Algorithm

PSO was proposed by Kennedy and Eberhart;<sup>16</sup> it is a stochastic optimization method based on swarm intelligence theory. It has been successfully applied in continuous optimization problems such as neural network training, voltage stability control, and the optimization of cutting parameters.

The main idea is to simulate the intelligent behavior of an individual particle in a swarm, just like an individual bird searching for food in a group or population. By group cooperation, the swarm can get maximum global optimization results.<sup>17,18</sup> In this paper the PSO method is applied to get the maximum optimized intrinsic or extrinsic global parameters of the fisheye camera so as to make the mosaicking function available by efficient calibration when installing PPAS in 4S shops.

The particle is the basic unit of the algorithm, and in the optimization problem, it is a candidate solution of the D-dimensional space. Assume the solution vector is D-dimensional and the total number of particles is  $n$ . When the algorithm iterates  $t$  times, the  $i$ 'th particle can be expressed as

$$x_i(t) = [x_{i1}(t), x_{i2}(t), \dots, x_{id}(t)], \quad (17)$$

where  $x_{ik}(t)$  represents the location of the  $i$ 'th particle in the  $k$ -dimensional solution space, and  $x_{ik}(t)$  is also the  $k$ 'th variable of  $i$ 'th candidate solution to be optimized.

Individual extreme  $p_i = [p_{i1}, p_{i2}, \dots, p_{id}]$  is the most optimal solution vector of a single particle from the initial search to the current iteration.

Neighborhood extreme  $l_i = [l_{i1}, l_{i2}, \dots, l_{id}]$  is the most optimal solution vector of a particle's corresponding neighborhood population from the initial search to the current iteration.

Particle velocity  $v_i(t) = [v_{i1}(t), v_{i2}(t), \dots, v_{id}(t)]$  represents a particle's changes in the location within the unit iteration number, that is, it is a particle's displacement of the solution variable in D-dimensional space, where  $v_{ik}(t)$  represents the velocity of the  $i$ 'th particle in the  $k$ -dimensional solution space.

The inertia weight  $w$  is used to control how much the speed of the previous iteration affects the current iterative speed. Generally, inertia weight is  $w \in [0, 1]$ . Shi and Eberhart found that a larger inertia weight is conducive to a global search in particle population, while a smaller inertia weight is inclined to a local search.<sup>19</sup> In the actual solving process of the optimization problem, the inertia weight is decreasing linearly with the iteration number  $w(t) = a \times w(t-1)$ . Therefore, in the initial search stage of this case, the search of the whole solution space can be carried out with high probability and can converge quickly to the optimal solution in a local area, where the local fine-tuning for particle population can be finished with the decrease of the inertia weight.

PSO first initializes a random particle population (random solution) and then searches the optimal solution by iteration. The particle updates its location and velocity through individual extremes and neighborhood extremes on every iteration.

The location of each particle changes according to the following equations.<sup>20,21</sup>

$$v_{id}(t+1) = w \cdot x_{id}(t) + c_1 R \quad \text{and} \quad (p_{id}(t) - x_{id}(t)) \\ + c_2 R \quad \text{and} \quad (l_{id}(t) - x_{id}(t)), \quad (18)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1), \quad (1 \leq i \leq n, 1 \leq d \leq D), \quad (19)$$

where  $c_1$  and  $c_2$  are learning factors, which are positive constants—generally  $c_1 = c_2 = 2$ ; Rand is a random number of  $[0,1]$ . Particles in the solution space keep tracing the extremes of the individual and neighborhood until meeting the stop conditions.

#### 4.2 Improved Particle Swarm Optimization Algorithm

The ordinary PSO algorithm is only an evolutionary mechanism, whose emphasis is more on groups to optimize or develop individuals. The traditional PSO methods have the advantage due to their good global convergence speed with a better solution; however, their disadvantage is that they easily fall into local optimum.

For the improvement of ordinary PSO, using PSO at the basis of a two-step method is applied by many people. First, the PSO method is used to do a global search, and then a second algorithm is applied which is good at processing locally. The solution obtained by PSO is regarded as the initial solution for further optimization. The problem is that after the global convergence, the solution is difficult to jump out of the region convergence. In conclusion, the conventional two-step method is better than routine PSO, but it still cannot improve the performance of PSO, essentially because it has been assumed that the PSO convergence region belongs to the optimal region. For example, there are three zones ( $A$ ,  $B$ , and  $C$ ) in the solution set— $A_{best}$ ,  $B_{best}$ , and  $C_{best}$  belong to  $A$ ,  $B$ , and  $C$ , respectively.  $C_{best}$  is better than  $B_{best}$ , and  $B_{best}$  is better than  $A_{best}$ ; if the PSO method could only find  $B_1$ , the second algorithm would definitely find  $B_{best}$ , yet it has not jumped out of  $B$  to find the best solution in the optimal  $C$  zone. By this way, all particles only rely on the good information, which means the particle moves to the best position in the local or global area. That would be premature aggregation and lost diversity and it would not be the best optimization.

Apart from this, the real working situation required the applied method to meet the following conditions:

1. Four cameras have been fixed on the automobile, and it is impractical to take multireference images for calibration.
2. The working environment varies, and the uneven light makes the grids unavailable to be detected on the reference and also affects the result.<sup>22</sup>

The method relying on multiple images or the edge-based method can hardly be applied in this situation. Therefore, to find a way to calibrate or optimize all the parameters with only one reference image and not be influenced by the working environment is important, because it will make calibration and optimization available in all kinds of work fields in the 4S shops.

An IPSO algorithm is, therefore, proposed to optimize the parameters, considering the above conditions.

The proposed IPSO method is able to break the convergence space for it has two mechanisms: toward the best and away from the worst. Both mechanisms are toward the right direction. The convergence rate of the algorithm maintains the same high efficiency, although it is not allowed to converge toward a single direction in which it will be trapped in the local solution, so as to find the global optimum.

When using a method relying on multiple images and edge-based method, the edges have to be determined manually if the edge cannot be detected. The IPSO allows a certain range of dynamic change for the parameters; it can exploit only one reference image to complete all the calibration or optimization without extra manual operation. That makes IPSO suitable in practical application in the workshop with light-changing conditions with a minimum of reference images.

If good and bad information are both taken into account in the algorithm to add a method of changing in velocity, the case will change when introducing the following equation:

$$v_{id}(t+1) = -w \times f[v_{id}(t), \theta] - c_1 \times R \quad \text{and} \\ \times [W_{id}(t) - x_{id}(t)] - c_2 \times R \quad \text{and} \\ \times [W_{gd}(t) - x_{id}(t)], \quad (20)$$

where  $W_{id}$  is the worst particle of the individual and  $W_{gd}$  is the worst particle of the global;  $f[v_{id}(t), \theta]$  represents the moving direction of  $v_{id}(t)$  being clockwise rotation, and the movement step remains unchanged.

Population density is introduced; if the density is low, it means the direction of evolution is good, so Eq. (18) is applied. If the density is high, it means the direction of the evolution is approaching the bad side, so Eq. (20) is applied. Diversity( $P$ ) is the measure of the density; the greater the density is, the lower the value of diversity( $P$ ) is. Its equation is as follows:<sup>23</sup>

$$\text{diversity}(P) = \frac{1}{|L||P|} \cdot \sum_{i=1}^{|P|} \sqrt{\sum_{j=1}^N (S_{ij} - \bar{S}_j)^2}, \quad (21)$$

where  $P$  represents population and  $|P|$  is the size of the population.

$|L|$  is the size of the search space,  $S \in R^N$ ,  $N$  is the number of dimensions,  $S_{ij}$  is the value of the  $i$ 'th particle in the  $n_j$ 'th dimension, and  $\bar{S}_j$  is the average value of a particle in the  $j$ 'th dimension.

In the PSO algorithm, the velocity  $v$  is used to promote the evolution of the particle population; however, this mechanism only works on specific particles, and when the particles tend to gather, the population can easily lose its vitality.

Increasing the uncertainty of the particle motion means the particle has a random nature. This can make the whole population maintain vitality. A variation will be introduced to the PSO, and Eq. (19) will be changed as follows:

$$x_{id}(t+1) = \begin{cases} x_{id}(t) + v_{id}(t+1), R \quad \text{and} \quad \geq r_0 \\ R \quad \text{and} \quad (1, D), R \quad \text{and} \quad < r_0 \end{cases}, \quad (22)$$

where  $r_0$  is the preset mutation probability,  $R$  and is a random number, and  $R$  and  $(1, D)$  randomly generates a position of the particle.

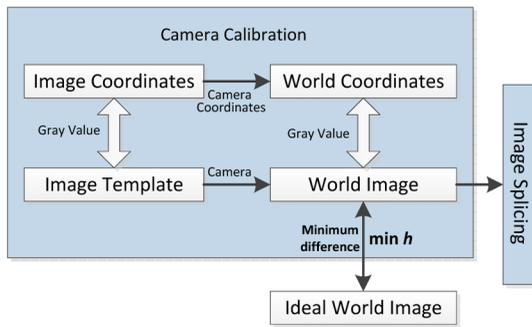


Fig. 3 The schematic procedure of PPAS.

The IPSO method has two phases: first, the particle is attracted by the best position, and second, it will be pushed away from the worst position. This mechanism means the particle moves to the good position or a “not bad” position. For the change of the particle reference, the swarm or group can leave the local optimum with the maximum probability. In IPSO, the mutation factor is introduced to enhance the vitality of the group.

From the process described above, it can be seen that the IPSO finds the optimized solution by iteration. This character makes it available to apply only one reference image to finish the whole optimization process without being influenced by the environment. That is the reason IPSO is proposed. The effect of the IPSO algorithm with the mutation operator is better than the traditional PSO algorithm, but the result is still not satisfactory, it still leaves room for improvement.

After parameter calibration and optimization, the four distorted images would be corrected and have one universal coordinate system. Therefore, the four images applied in this application could be spliced by coordinates alone. Because the scene in this paper is also comparatively

small and easy to mosaic by coordinates, some more complex region-based and feature-based splicing methods are not further compared after the process of calibration and optimization.

### 5 System Implementation

The main processor’s frequency of PPAS is ~600 MHz. The main components include DSP, DDR2, Nor flash, a clock, and reset circuit. When the power is on, the processing code will be loaded from Nor flash to DDR2. Nor flash is also used to save the parameters of the fisheye camera for calibration. The main logical control module is in charge of event detection (such as reverse wire-level change) and display control.

The image sensor of the fisheye camera’s minimum illumination can be as low as 0.01 Lux, with its operating temperature being between 40 and 105°C. The camera’s optical horizontal view angle is >165 deg. The system architecture for PPAS is shown in Fig. 4.

The video output port is connected to an LCD display of the on-board DVD system. Video frames captured by the four fisheye cameras are multiplexed into one video stream by a video decoder. The video processing front end of DSP demultiplexes the stream to four separate frames saved in an input buffer, and the DSP gets this frame data from the buffer and processes it. The resulting image will be saved in the output buffer and then displayed on a screen.

If the ultrasound transducers are mounted at the front or the rear of the car, they can be connected through general purpose input output (GPIO), by which the sound warning function can be created when there is a static or moving object at the front or rear of the car. This paper focused on the main module of image mosaicking, not creating the warning function in the hardware.

As Fig. 5 shows, the front camera is generally installed below or beside the brand logo, with the rear camera installed above the bumper. The left camera is installed in the bottom

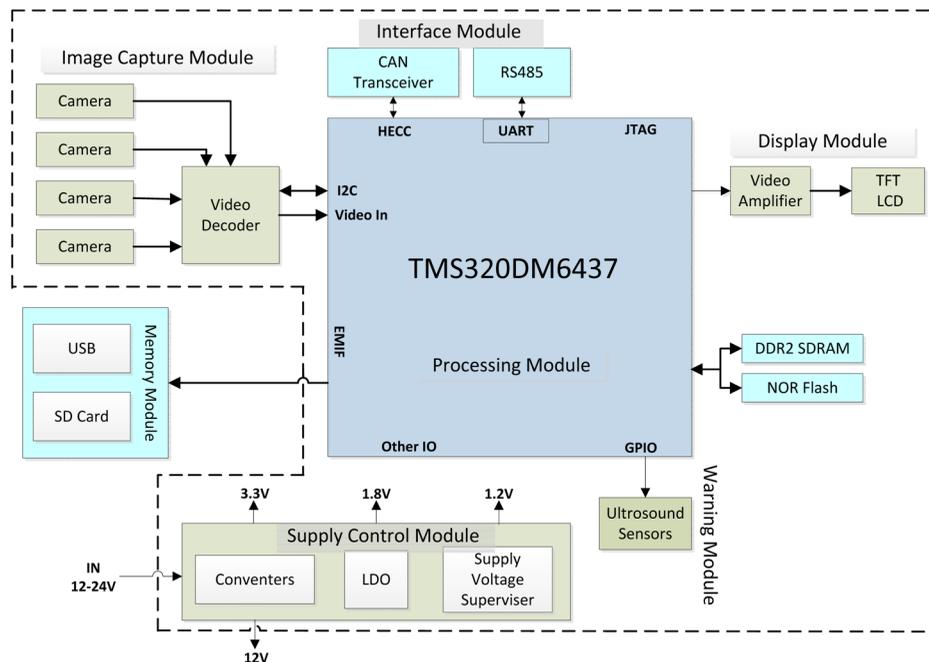


Fig. 4 System architecture for PPAS.



Fig. 5 PPAS installed in a Toyota Highlander.

of the rear-view mirrors, and the right camera is not shown in the picture while it is in the symmetrical position of the rear-view mirror according to the left. Figure 5 also shows the display screen on the panel, and the top-view image area in this prototype's screen is set in the left half of the display.

## 6 Experiment and Result

The whole procedure of PPAS includes image input, camera calibration, image process, and image output, and if the ultrasound transducers are installed, there is also an object warning period. Among these steps, the fisheye camera calibration and image mosaicking are the key phases, and the final result should also be tested in real scenes with an on-board system in the real experiment car. The results are as follows.

### 6.1 Result of Fisheye Camera Calibration

In the fisheye calibration step, Zhang's two-step method and polynomial coordinate transformation method are applied. The calibration needs at least three images with different shooting angles. In this study, nine chessboard-like images with  $8 \times 8$  grids shooting from different directions are used for the calibration. Forty-nine corners need to be detected. An improved

chess corner detection function is applied for the detection; it can get a better result when the fisheye distortion is bigger or the quality of the image is not good. Figure 6 shows the chess corners detected by this improved function.

Use the image with the detected chess corners to calibrate the fisheye camera. In order to increase the accuracy of the result, a recalibration step based on former calibration is necessary.

To test the accuracy, an inverse projection of the grid corner procedure is applied, as shown in Fig. 7, where the character + represents the detected corners and the circle character indicates the corner location by inverse projection. The figure shows that the calibrated locations of the corners are fit for the original locations.

The errors for the inverse projection of different images are shown in Fig. 8. The average range of errors in the distortion correction is generally within one pixel, thereby meeting the basic functional needs of the system.

Figure 9 is the result of the correction of the fisheye camera distortion after basic calibration.

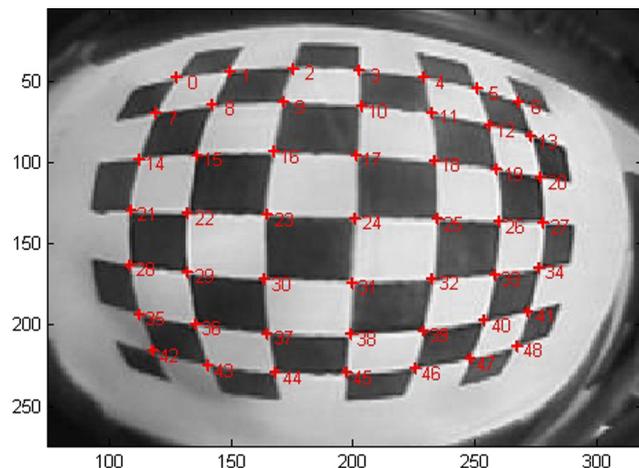


Fig. 6 The corners detected by the program.

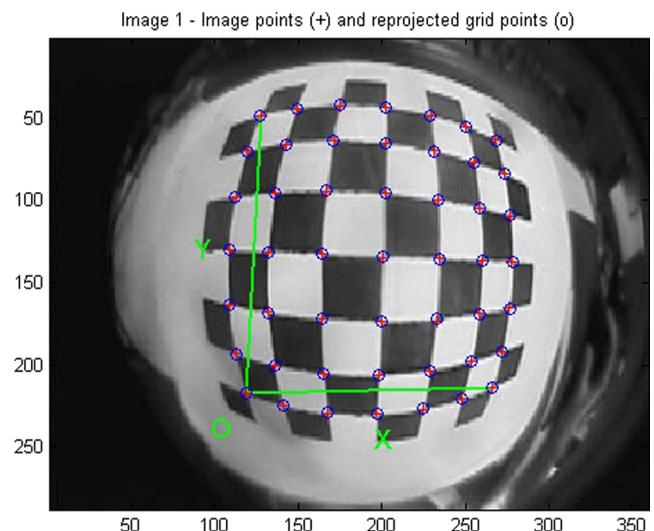


Fig. 7 Image points and reprojected grid points.

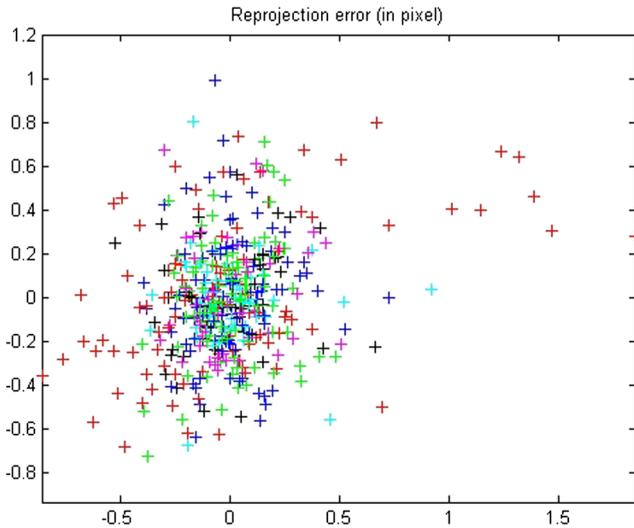


Fig. 8 The error map of inverse projection.

### 6.2 Result of Image Mosaicking

In this study, image mosaicking can be considered the problem of parameter optimization. When the parameter calibration of the four cameras around the vehicle is finished and the errors are minimized in global, the four images are blended.

Besides errors of intrinsic parameters due to manufacturing tolerance, the installation technique of different workers also contributes deviations to the intrinsic parameters. Therefore, the parameter calibration according to the standard reference parameters is necessary in IPSO optimization. It would increase the installation and calibration efficiency in the production line, meanwhile maintaining the same high accuracy.

Four templates are used in the experiments to analyze and compare the results of three algorithms—traditional PSO, IPSO, and PSO—with a mutation operator.

In this study, the angle of the fisheye camera lens is >165 deg. According to the result of experiments, the standard reference intrinsic parameters in this paper are regulated as



Fig. 9 The result of correction of the fisheye camera distortion after basic calibration.

Table 1 Comparison of three methods' success rate.

Method	1 (%)	2 (%)	3 (%)	4 (%)	5 (%)	6 (%)
Particle swarm optimization (PSO)	80	70	70	80	50	60
PSO with mutation operator	100	80	70	90	70	70
Improved PSO (IPSO) with mutation operator	100	100	100	100	90	100

$$(\alpha_x, \alpha_y, u_0, v_0, k_1, k_2, p_1, p_2) = (87.94, 79.87, 168.27, 133.76, 0.2119, 0.0966, -0.0131, -0.0066).$$

The four fisheye cameras have two sets of extrinsic parameters for their locations. One set corresponds to the front and the rear cameras, and the other set corresponds to the cameras on both sides of the vehicle.

The front/back cameras' extrinsic parameters are set as

$$(r_1, r_2, r_3, t_1, t_2, t_3) = (1.76, 1.53, -0.8, -95, 75, 35)$$

and the side cameras are set as

$$(r_1, r_2, r_3, t_1, t_2, t_3) = (1.96, 2.19, -0.40, 215, 85, 105).$$

With these reference parameters, the PPAS will be calibrated with a black and white chessboard-like map as a reference to get preferable optimized parameters.

Three workers were arranged to calibrate the side and the front/back cameras, and got a serial of experiment data to be evaluated. As Eq. (15) described, the optimization problem can be considered to minimize the difference of the rebuilt image and the original one, so the evaluation criteria can be defined as follows.

Using Eq. (15) to calculate the ratio of the minh and the total pixel of the template, if the ratio is <6%, it is defined as success. Calculate the ratio of the success number and the total calibration number. It is defined as the success rate.

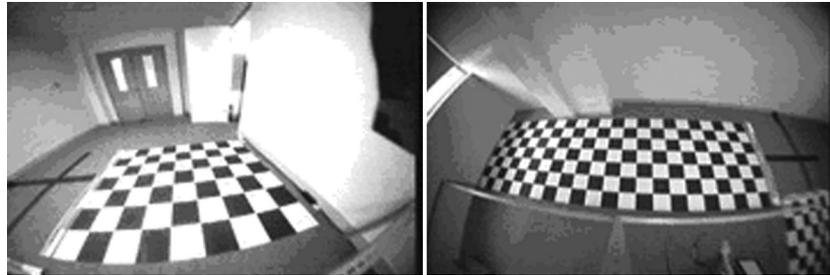
The success rate of these three methods is compared in Table 1. Eight pictures were taken for the experiments and each picture was performed for 10 tests for the experiments, and the number 1,2,3,4,5,6 is the number of six selected experimental pictures. From the table, it can be seen that the success rate of the IPSO with a mutation operator is better than the other two methods and that it meets the demands of the PPAS system.

Table 2 The average calibration time of IPSO.

Position	Left camera (s)	Front camera (s)	Right camera (s)	Back camera (s)
Worker A	62	249	49	301
Worker B	53	356	72	195
Worker C	38	322	51	266

**Table 3** Parameters optimized by IPSO.

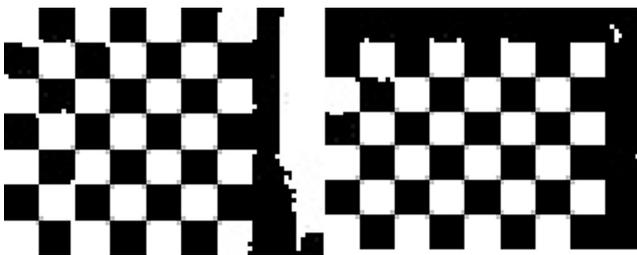
Parameter	$\alpha_x$	$\alpha_y$	$u_0$	$v_0$	$r_1$	$r_2$	$r_3$
Front camera	87.53	81.29	169.22	134.06	1.80	1.63	-0.72
Side camera	88.26	80.15	168.46	133.85	1.89	2.13	-0.47
Parameter	$t_1$	$t_2$	$t_3$	$k_1$	$k_2$	$p_1$	$p_2$
Front camera	-72	80	48	0.2107	0.0926	0.0219	0.0168
Side camera	198	85	92	0.2158	0.1023	0.0105	0.0147



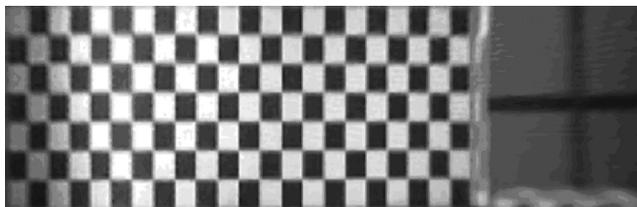
**Fig. 10** The original images.



**Fig. 11** Top-view image with initial external parameter.



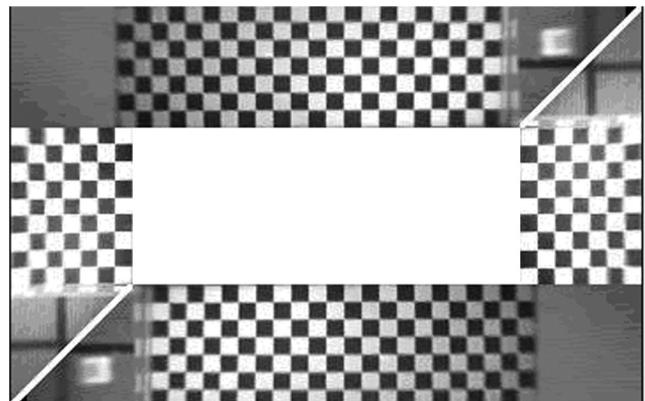
**Fig. 12** Local optimal solution obtained by improved particle swarm optimization.



**Fig. 13** The maximum top view using the optimized extrinsic parameters.

Besides the price factor, the convenience of the installation is another key problem for PPAS utilization in the mass market. The most time-consuming step in installation is camera calibration because it has to take into account all the errors accumulated in every manufacture and installation step to ensure the accuracy of mosaicking.

Table 2 shows the average calibration time of IPSO. The tests are based on three sets of experiments upon three different cars of the same type. From the table it can be seen that



**Fig. 14** Top view after mosaicking.



**Fig. 15** Top-view image from the first version of PPAS prototype (Video 1, MPEG, 11.3 MB) [URL: <http://dx.doi.org/10.1117/1.JEI.22.4.041123.1>].



**Fig. 16** Top-view image after mosaicking in final version of PPAS.

the average calibration time of each camera is  $<3$  min. Compared with manual calibration, the efficiency was increased more than twice.

Table 3 shows one set of optimized parameters of the front and the side fisheye cameras by the IPSO method; it is the result of one specific brand of car made by Dongfeng Nissan. Two original images captured by the fisheye cameras in the experiment are shown in Fig. 10.

The images with initial external parameters set by PSO are shown in Fig. 11.

The image optimized by IPSO is shown in Fig. 12, and it is the image of the local optimal solution. Figure 13 is

the maximum top view using the optimized extrinsic parameters.

After the calibration of the cameras, a more initial way to judge the effect of the calibration is to restore the images captured by the fisheye camera into a 2-D top-view image. The mosaicking effect can be judged by evaluation criteria mentioned above; if ratio  $L$  is  $<6\%$ , the result is preferred.

The ultimate top mosaicking view is shown in Fig. 14. The ratio  $L$  of the image in Fig. 17 is  $<6\%$ , and from the figure, it can be seen that the four restored images are well spliced. The deviation of  $+$  at the joints of the two neighbor images are very small. It is available enough for PPAS.

### 6.3 Real Car Test

The PPAS has been installed and tested in several types of cars to test its performance.

Figure 15 shows the performance of the first version of the PPAS prototype installed in a Nissan NV200. It was tested on the basketball ground outside the workshop to see the continuity of the curve after image mosaicking. The figure shows that the curve can almost be connected seamlessly. The test video is Video 1.

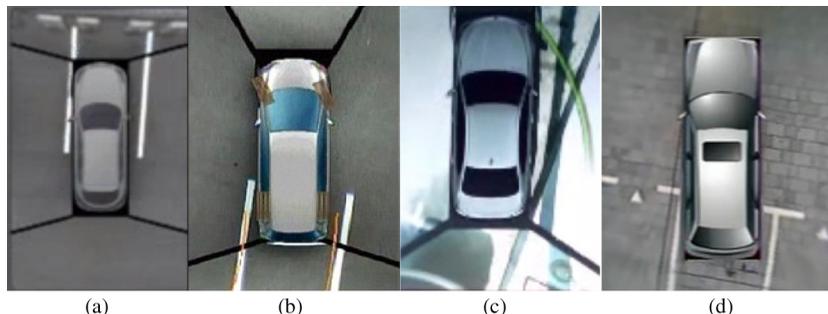
Figure 16 shows the performance of the final version of the PPAS. The system algorithm was upgraded in a seamless connection by applying the weighted smoothing method;<sup>24</sup> therefore the four images blocks mosaicked by coordinates were further made seamless in borders.

Figure 17 shows the mosaicking effect comparison of different brands of car. Figure 17(a) is the effect of Infiniti's around-view system, Fig. 17(b) is the mosaicking effect of the Honda multiview system, Fig. 17(c) is the effect of BMW 5 serial's parking assistant system, and Fig. 17(d) is the mosaicking effect of the proposed PPAS in this paper.

The PPAS was also installed in other types of automobile and can be widely used in the after-auto market for family cars. The wide application of a similar system would greatly improve parking safety, especially for children, which is the purpose of this study.

### 7 Conclusion

The design of a PPAS is proposed in this paper. The PPAS first uses four fisheye cameras to collect image information surrounding the car and then integrates them into one top-view image. This parking assistant system can avoid serious parking accidents and also make parking an easy and safe task, even for newly trained drivers. The detailed contributions of this study are as follows:



**Fig. 17** Top-view effect of different types of car.

1. Use a two-step method together with a polynomial coordinate transformation method to calibrate the fish-eye camera, testing it with the program through recalibration to increase the accuracy of the results and meet the basic requirements of the system design.
2. Introduce an IPSO algorithm to optimize parameters of the fisheye cameras in PPA. The local optimum value can be obtained with only one reference image.
3. Design an embedded on-board system and test it with experimental cars in real scenes. The performance of the PPA is stable. Much less time is needed to complete the system installation, so it is feasible to apply it in a 4S shop.
4. The PPA system can be widely used in various types of automobiles in the after-market with a comparatively cheaper cost.

### Acknowledgments

The authors would like to thank Zejun Wu, Yong Dai, Wei Chen, and Jianbo Zhou for their support in the implementation and field testing. The authors would also like to thank Safdao Technology Co. Ltd. and Zhengzhou Nissan Company for their help with calibrating the camera and providing an experimental car (NV200) for us to test the algorithm and the Panorama Parking Assistant System prototype.

### References

1. J. Zhang, F.-Y. Wang, and K. Wang, "Data-driven intelligent transportation systems: a survey," *IEEE Trans. Intell. Transp. Syst.* **12**(4), 1624–1639 (2011).
2. T. Gandhi and M. Trivedi, "Vehicle surround capture: survey of techniques and a novel vehicle blind spots," *IEEE Trans. Intell. Transp. Syst.* **7**(3), 293–308 (2006).
3. S. Maybank and O. Faugeras, "A theory of self-calibration of a moving camera," *Int. J. Comput. Vis.* **8**(2), 123–151 (1992).
4. Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(11), 1330–1334 (2000).
5. R. Hartley and S. B. Kang, "Parameter-free radial distortion correction with center of distortion estimation," *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(8), 1309–1321 (2007).
6. J. Park, S.-C. Byun, and B.-U. Lee, "Lens distortion correction using ideal image coordinates," *IEEE Trans. Consum. Electron.* **55**(3), 987–991 (2009).
7. R. G. von Gioi et al., "Lens distortion correction with a calibration harp," in *2011 18th IEEE Int. Conf. Image Processing*, pp. 617–620, IEEE, Brussels, Belgium (2011).
8. R. Tsai, "An efficient and accurate camera calibration technique for 3D machine vision," in *Proc. of Computer Vision and Pattern Recognition*, pp. 364–374, IEEE (1986).
9. H. Zhang and Y. Shiu, "Noise-insensitive algorithm for robotic hand/eye calibration with or without sensor orientation measurement," *IEEE Trans. Syst., Man, Cybern.* **23**(4), 1168–1175 (1993).
10. E. Mails and R. Cipolla, "Multi-view constraints between collineations: application to self-calibration from unknown planar structures," *Lec. Notes Comput. Sci.* **1843**, 610–624 (2000).
11. B. Triggs, "Auto calibration and the absolute quadric," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 609–614, IEEE, San Juan, Puerto Rico (1997).
12. S. Liao, P. Gao, and Y. Su, "A geometric rectification method for lens camera," *J. Image Graph.* **5**(7), 593–596 (2000).
13. H. Zhou and L. Wang, "A fast algorithm for rectification of optical lens image distortion," *J. Image Graph.* **8**(10), 1131–1135 (2003).
14. Y. Xu et al., "A two-phase test sample sparse representation method for use with face recognition," *IEEE Trans. Circuits Syst. Video Technol.* **21**(9), 1255–1262 (2011).
15. Y. Xu and Q. Zhu, "A simple and fast representation-based face recognition method," *Neural Comput. Appl.* **22**(7–8), 1543–1549 (2012).
16. J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. of IEEE Int. Conf. on Neural Networks*, Vol. 4, pp. 1942–1948, IEEE, Perth, WA (1995).
17. F. Pan et al., "Several characteristics analysis of particle swarm optimizer," *ACTA Automatica Sinica* **35**(7), 1010–1015 (2009).
18. X. Jin et al., "Convergence analysis of the particle swarm optimization based on stochastic process," *ACTA Automatica Sinica* **33**(12), 1263–1268 (2007).
19. Y. H. Shi and R. Eberhart, "A modified particle swarm optimizer," in *Proc. of IEEE Conf. on Evolutionary Computation*, pp. 69–73, IEEE, Anchorage, Alaska (1998).
20. I. W. Jiang, J. Xie, and Y. Zhang, "A particle swarm optimization algorithm based on diffusion-repulsion and application to portfolio selection," in *ISISE 08*, pp. 498–501, IEEE, Shanghai, China (2008).
21. J. X. Liu et al., "Template matching based on an improved PSO algorithm," in *Third International Symposium on Intelligent Information Technology Application Workshops, 2009, Nanchang, China*, pp. 292–296 (2009).
22. A. Stenkula, "Vehicle vicinity from above—a study of all-around environment displaying system for heavy vehicles," MS Thesis, Department of Computer Science, Royal Institute of Technology, Sweden (2009).
23. K. X. Liu, W. Jiang, and J. Xie, "A particle swarm optimization algorithm based on molecule diffusion," in *International Conference on Industrial Mechatronics and Automation 2009, Chengdu, China*, pp. 125–128 (2009).
24. Y. Wei et al., "Improved image mosaic algorithm based on feature point matching," *Infrared Technol.* **32**(5), 288–290 (2010).



**Ruzhong Cheng** received his MS degree in aerospace engineering and mechanics in 2003 from Harbin Institute of Technology (HIT). He had worked in the MEMS Center of HIT from 2004 to 2005. He is currently working toward his PhD degree in the Department of Electronic Engineering and Computer Science in Peking University Shenzhen Campus. His main research interest is computer vision.



**Yong Zhao** received his MS degree in electrical engineering from Northwestern Polytechnic University and PhD degree from Southeast University in 1989 and 1991, respectively. He worked for Honeywell Canada from 2000 to 2004. He joined the Department of Electronics Engineering, Peking University Shenzhen Campus in 2004, where he is currently an associate professor. His research interests mainly focus on the embedded application of intelligence image algorithms, such as scene target detection, extraction, tracking, recognition, and behavior analysis.



**Zhichao Li** received his BS degree in biomedical engineering from Huazhong University of Science and Technology in 2008. He received his MS degree in software engineering from Peking University in 2011. He has been with Safdao Technology Co. Ltd. since 2010 and engaged in the research of advanced driver assistant system (ADAS). His research interests include computer vision and embedded system design.



**Weigang Jiang** received his BS degree in computer science from Jinan University in 2008. He has worked for Safdao Technology Co. Ltd. since 2010 as an engineer for the project of ADAS. He is pursuing his MS degree at Huazhong University of Science and Technology at the same time. His current research interests include intelligent computing and computer vision.



**Xin'an Wang** received his MS and PhD degrees in microelectronics from Shanxi Microelectronics Institute in 1989 and 1992, respectively. He is currently a professor in the School of Electronics Engineering and Computer Science, Peking University. His research interests focus on the area of application-specified integrated-circuit design, IC design methodology, and embedded system design of electronics products.



**Yong Xu** received his BS and MS degrees from the Air Force Institute of Meteorology, Nanjing, China, in 1994 and 1997, respectively. He then received his PhD degree in pattern recognition and intelligence systems at the Nanjing University of Science and Technology, Nanjing, in 2005. He is currently an associate professor at Shenzhen Graduate School, HIT. He also acted as a research assistant at Hong Kong Polytechnic University, Kowloon, Hong Kong, from August 2007 to June 2008. His current research interests include pattern recognition, biometrics, video analysis, and machine learning.