Sensors and Materials, Vol. 33, No. 9 (2021) 3169–3186
MYU Tokyo

S & M 2679

Robust 3D Model Reconstruction Based on Continuous Point Cloud for Autonomous Vehicles

Hongwei Gao,[1,2] Jiahui Yu,[3*] Jian Sun,[1] Wei Yang,[1]
Yueqiu Jiang,[1] Lei Zhu,[4] and Zhaojie Ju[3**]

[1]School of Automation and Electrical Engineering, Shenyang Ligong University, Shenyang 110159, China
[2]China State Key Laboratory of Robotics, Shenyang Institute of Automation,
Chinese Academy of Sciences, Shenyang 110016, China
[3]School of Computing, University of Portsmouth, Portsmouth, PO1 3HE, UK
[4]College of Automation (Artificial Intelligence), Hangzhou Dianzi University, Hangzhou 310018, China

(Received December 22, 2020; accepted August 26, 2021)

Keywords: dense 3D point cloud, region growing, match optimization, monocular zoom stereo vision

Continuous point cloud stitching can reconstruct a 3D model and play an essential role in autonomous vehicles. However, most existing methods are based on binocular stereo vision, which increases space and material costs, and these systems also achieve poor matching accuracies and speeds. In this paper, a novel point cloud stitching method based on the monocular vision system is proposed to solve these problems. First, the calibration and parameter acquisition based on monocular vision are presented. Next, the region-growing algorithm in sparse matching and dense matching is redesigned to improve the matching density. Finally, an Iterative Closest Point (ICP)-based splicing method is proposed for monocular zoom stereo vision. The point cloud data are spliced by introducing the rotation matrix and translation factor obtained in the matching process. In the experiments, the proposed method is evaluated on two datasets: self-collected and public datasets. The results show that the proposed method achieves a higher matching accuracy than the binocular-based systems, and it also outperforms other recent approaches. In addition, the 3D model generated using this method has a wider viewing angle, a more precise outline, and more distinct layers than the state-of-the-art algorithms.

1. Introduction

Model generation technology has been widely used in many areas, such as panoramic display, navigation, and data visualization. Many continuous images collected by vehicle-mounted cameras need to be stitched to reconstruct the environment for autonomous vehicles. Both image matching and stereo point cloud stitching are key technologies. A 3D point cloud is defined as the collection of points on the surface of an object and represents the 3D model.(1)

In most studies, point cloud data are obtained by applying professional instruments and binocular stereo vision systems. The optical scanner can acquire point cloud data on a target surface in seconds. Mineo *et al.* and Berger *et al.* applied costly devices to obtain point cloud

*Corresponding author: e-mail: jiahui.yu@port.ac.uk
**Corresponding author: e-mail: zhaojie.ju@port.ac.uk
https://doi.org/10.18494/SAM.2021.3231

data for the target surface but got an unsatisfactory result. Multiple transformation models such as non-uniform rational B-splines (NURBS) surface models and computer aided design (CAD) models have obtained efficient point cloud data.[2,3] Point cloud data can also be obtained by methods based on binocular stereo vision. Yang proposed a bilateral filtering method suitable for hardware implementation to improve the speed of matching computation.[4] Einecke and Eggert proposed an anisotropic mean filtering method to improve the accuracy of matching.[5] Kowalczuk *et al.* proposed a dynamic programming method to improve the speed of matching.[6] However, these methods are complex, require high computational resources, and increase the cost of memory.

3D point cloud stitching is an important technology that enables an autonomous vehicle to understand its surroundings and tendency of motion. There are three reconstruction methods differing in the number of images required, namely, single-image-based, double-image-based, and image-sequence-based methods. First, a small amount of image information is used to reconstruct the 3D model in our proposed method, which significantly improves the speed of model operation. Schönberger *et al.* proposed a joint reconstruction and retrieval system that improves scalability and stores many scene details.[7] Huang *et al.* extended automatic object reconstruction by analyzing different objects and formulating optimization strategies.[8] Effective unsupervised deep networks were extended by Payne *et al.* and Eigen *et al.* to recognize and reconstruct 3D objects.[9,10] However, the features extracted from a single image are limited, leading to poor accuracy and incompleteness. Second, the parallax of multiple spatial points collected by two-view sensors is calculated to reconstruct the 3D model. The key technology is to match feature points and calibrate the sensors. Fraser proposed some self-calibration methods to obtain scene-independent camera calibration parameters.[11] A geometric calibration method to automatically estimate the intrinsic, extrinsic, and distortion parameters was proposed by Li *et al.*[12] Although double images provide more feature points than a single image for reconstruction, these feature points still cannot achieve a highly complete model.

In this paper, we propose a model based on an image sequence to improve the speed and accuracy of stitching 3D point clouds. A depth image sequence is used in model reconstruction in our model. The fusion of multiple sets of depth images can be used to obtain the complete 3D information of a model, and the gray values of each pixel can represent the different features.[13] The depth-image-based method was studied by Ju *et al.* to propose a continuous deep fusion model.[14] Hanqi *et al.* achieved a complete depth surface by introducing Bayesian mapping to constrain images from various perspectives.[15] The density of the point cloud is a key factor for reconstructing the 3D model, and a region-growing-based method can improve the density.[16] Tanskanen *et al.* proposed a monocular device to generate a 3D model with a high density suitable for outdoor and indoor scenes.[17] Lasang *et al.* obtained a dense 3D model by combining color and depth images.[18] Motivated by the recent success of the region-growing-based method, we used such a method in our model to achieve a high density and good reconstruction accuracy.

To improve the feature matching accuracy and the speed of reconstruction, we proposed a point cloud stitching method based on monocular stereo vision. The main contributions of our paper are summarized as follows. 1) From a detailed analysis of the imaging principle and characteristics of the monocular camera, an offline calibration is proposed to obtain stable

camera parameters at different focal lengths. 2) An image generation method based on the Iterative Closest Point (ICP) algorithm is proposed, including data simplification, matching, and image splicing. 3) The image-matching algorithm is improved by combining the region-growing theory.

This paper is organized as follows. In Sect. 2, we describe the camera calibration method. In Sects. 3 and 4, we introduce our proposed model in detail. In Sect. 5, we evaluate the proposed model, and in Sect. 6, we describe the limitations of our model and outline future work.

2. Related Work

2.1 Calibration method

Camera calibration involves finding the internal and external parameters of a camera in accordance with a given camera model and establishing the mapping method between the image pixel coordinate system and the spatial coordinate system, that is, the position of the 3D space point and the positional relationship of the image pixel points captured by the camera. In this work, we study the point cloud stitching method based on zoom monocular stereo vision. The internal and external parameters of the camera under different focal lengths need to be calibrated, and the calibrated projection matrix is used for subsequent point cloud calculation. We capture the zoom image using a manual zoom camera and calibrate the zoom camera using a 2D planar template and a MATLAB calibration box. The in-camera parameter model of the calibration box is expressed as

$$M = \begin{bmatrix} fc(1) & alpha\_c \cdot fc(1) & cc(1) \\ 0 & fc(2) & cc(2) \\ 0 & 0 & 1 \end{bmatrix}. \tag{1}$$

The parameter *fc* is a vector that represents the focal length of the horizontal pixel. We use a 1 × 2 matrix to express this parameter. *cc* is a vector expressed as a 1 × 2 matrix that represents the center of the image, the tilt factor *alpha_c* represents the tilt angle in the *kc* direction and the *y* direction, and *alpha_c* is 0. Also, the distortion factor *kc* is a 1 × 5 vector, and *kc* is taken as [1, 1, 1, 0, 0].

On the basis of the planar target, images from multiple viewpoints are acquired to complete the camera calibration. The method sets the coordinate system of the calibration plane plate used in the calibration in space coordinate system to $Z = 0$. We calculate the optimal solution of the camera parameters based on the linear model and solve the nonlinear solution by the maximum likelihood method. The equation for the imaging model is

$$s\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = M\begin{bmatrix} m & n & o & p \end{bmatrix}\begin{bmatrix} x_{wi} \\ y_{wi} \\ 0 \\ 1 \end{bmatrix} = M\begin{bmatrix} n & o & p \end{bmatrix}\begin{bmatrix} x_{wi} \\ y_{wi} \\ 1 \end{bmatrix}, \tag{2}$$

where s is the depth coefficient and $M = \begin{bmatrix} k_x & k_s & u \\ & k_y & v \\ 0 & 0 & 1 \end{bmatrix}$ is the parameter matrix in the camera. Let $H = M[n\ o\ p] = [h_1\ h_2\ h_3]$ be a homography matrix. H is obtained by maximum likelihood estimation and the algorithm in accordance with the constraints of the in-camera parameter matrix.

$$\begin{cases} h_1^T M^{-T} M^{-1} h_2 = 0 \\ h_1^T M^{-T} M^{-1} h_1 = h_2^T M^{-T} M^{-1} h_2 \end{cases}$$
$$B = M^{-T} M^{-1} = \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{12} & B_{22} & B_{23} \\ B_{13} & B_{23} & B_{33} \end{bmatrix} \tag{3}$$

In Eq. (3), the H matrix is obtained by assuming that the observed samples are independently and identically distributed. The inner parameter matrix is below.

$$\begin{cases} v_0 = \left(B_{12}B_{13} - B_{11}B_{23}\right) / \left(B_{11}B_{22} - B_{12}^2\right) \\ k_x = \sqrt{c / B_{11}} \\ k_y = \sqrt{cB_{11} / B_{11}B_{22} - B_{12}^2} \\ k_s = -B_{12}k_x^2 k_y / c \\ u_0 = k_s v_0 / k_y - B_{13}k_x^2 / c \end{cases} . \tag{4}$$

After finding the inner parameter matrix using the homography matrix $H = \lambda M[n\ o\ p] = [h_1\ h_2\ h_3]$ (λ is a constant factor), the camera external parameters are obtained as below.

$$\begin{cases} \lambda = 1 / \left\| M^{-1} h_1 \right\| = 1 / \left\| M^{-1} h_2 \right\| \\ n = \lambda M^{-1} h_1 \\ o = \lambda M^{-1} h_2 \\ a = n \times o \\ p = \lambda M^{-1} h_3 \end{cases} \tag{5}$$

2.2 Calculation of 3D coordinates

When the zoom camera captures images at different focal lengths, image pairs with different focal lengths can be considered positional relationships translated along the optical axis. The

thick lens model is deemed to be the ideal model for zoom lenses.[19] As shown in Fig. 1, the principal planes H_{oxy} and H_{ixy} are perpendicular to the optical axis and intersect at points H_{oxy} and H_{ixy}. The object distance H_{oxy} is the distance from the object to the main plane p_o, and the image distance H_{ixy} is the distance from the image plane to the main plane p_i. When zooming, the main planes H_{oxy} and H_{ixy} move along the optical axis. In addition, when H_o and H_i coincide, the thick lens model is transformed into a pinhole model, where C is the projection center in the pinhole model, $p_o = T_z$ (the distance from the object to the center of the projection), and $p_i = f$ (the distance from the image plane to the center of the projection).

According to the characteristics of the zoom image, the change in the focal length of the camera zoom is equal to the change in the object distance. However, owing to various factors, the change in the focal length is not equal to the change in the object distance. The thick lens model is more suitable than the pinhole model for zoom depth information recovery. In the thick lens model, the translation of the principal plane H_{oxy} determines the position at which the ray is incident, whereas the translation of the principal plane H_{ixy} determines the focal length. If H_o is a static reference point and the main plane H_{ixy} is shifted to coincide with the main plane H_{oxy}, we find that the change in the object distance Δt is the key to zoom ranging. By using an image with at least two different focal lengths, the depth calculation equation for the thick lens model is

$$Depth = \frac{\Delta t r f_2}{f_1 r_2 - f_2 r_1}. \tag{6}$$

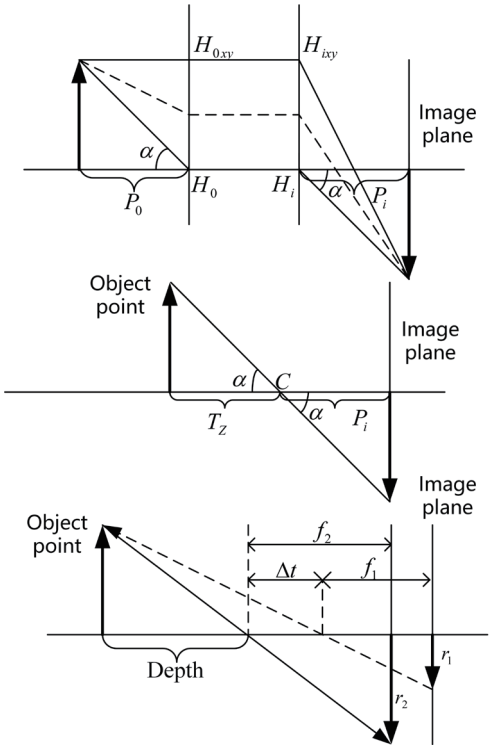


Fig. 1. Thick lens model for zoom depth estimation.

To accurately obtain the depth information of the object point, the 3D coordinate system of the spatial point is calculated with Eq. (7) using the calibration result of the zoom camera, where M is the internal parameter matrix of the camera, z_c is the depth of the point, and $\overline{u}$ is the image point coordinate in the image.

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = M^{-1} z_c \overline{u} \tag{7}$$

3. Proposed Dense Matching Algorithm

3.1 Image feature detection and tracking

For a monocular image under double the focal length, the position and angle do not change. A bifocal monocular vision mainly involves a change in the scale, and the main direction should remain unchanged. The scale is used to measure the view degree of the image. In the scale-invariant feature transform (SIFT) algorithm, the descriptors with similar degrees of blur are most likely to be matching points. However, the same target object is captured in a bifocal image pair compared with a bifocal monocular image. The degree of blurring is different because of the inherent noise. Therefore, to make the blurring degree of the two images similar, the smoothing scales are necessarily unequal and fixed. The zoom images are taken at focal lengths of F18 and F55, and the main directions and scale characteristics of the bifocal monocular image matching points are verified. The results show that the main trends of the bifocal monocular image are the same, and the ratio of the matching point scale is close to 3, which is the ratio of the focal length.

Let the number of matching points obtained by the SIFT algorithm be N, and the matching point scale ratio is $s(i)$, the main direction ratio is $o(i)$, and the value range of i is $[1, N]$. Therefore, the scale average ratio can be expressed using Eq. (8), and the main direction average ratio can be expressed using Eq. (9). Their standard deviations are given by Eqs. (10) and (11), respectively.

$$\mu_s = \frac{1}{N}\sum_{i=1}^{N} s(i) \tag{8}$$

$$\mu_o = \frac{1}{N}\sum_{i=1}^{N} o(i) \tag{9}$$

$$\sigma_s = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(s(i)-\mu_s\right)^2} \tag{10}$$

$$\sigma_o = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(o(i)-\mu_o\right)^2} \tag{11}$$

Let $s(i)$ and $o(i)$ have normal distributions. We remove anomalous matching points that do not satisfy the scale and main direction characteristics within the confidence interval. Let $s(i)$ have a confidence space of $[u_s - k_s\sigma_s, u_s + k_s\sigma_s]$ and k_s is the standard deviation factor. The confidence space of $o(i)$ is $[u_o - k_o\sigma_o, u_o + k_o\sigma_o]$ and k_o is the main direction ratio standard deviation factor. By adjusting the standard deviation factor, the confidence interval can be controlled. After the matching of feature attributes based on the SIFT algorithm, there is still mismatching. Therefore, on the basis of this, the polar constraint of the bifocal monocular image is used to remove the mismatch.

The pole-to-pole distance is called the polar line distance. According to the zoom image feature, the zoom image can be understood as a unique panning image, as shown in Fig. 2. The blue cube simulates the zoom image. The orange circle simulates the size of the target object at different focal lengths. The points of different colors in the blue cube simulate the matching points in the bifocal monocular image. In an ideal state, the extension of the line between the matching pairs of points in the blue cube intersects with a point O_1, which is called the pole. On the basis of the least-squares method, a pole is fitted by the polar line of the matching point, and the mismatched pair is removed using the pole distance. The calculation of the pole distance is given by Eq. (12), A_i, B_i, and C_i are the i pole line equations, and (x_0, y_0) are poles.

$$d_i = \frac{A_i x_0 + B_i y_0 + C_i}{\left| A_i^2 + B_i^2 \right|} \tag{12}$$

Let the i-pole distance be $d(i)$, $i \in [1, N]$. The average ratio of the polar line distances and the standard deviations are determined using Eqs. (13) and (14), respectively.

$$\mu_d = \frac{1}{N}\sum_{i=1}^{N} d(i) \tag{13}$$

$$\sigma_d = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(d(i) - \mu_d\right)^2} \tag{14}$$

The polar line distance obeys the normal distribution, and the equidistance interval is used to remove the mismatch point of the pole line distance. The confidence interval for the polar line distance is $[\mu_d - k_d\sigma_d, \mu_d + k_d\sigma_d]$, where k_d is the multiple of the standard deviation of the matching point line distance and is used to control the size of the confidence interval. Once the

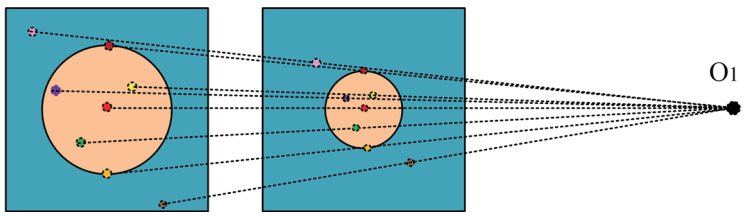


Fig. 2. (Color online) Bifocal image pole schematic.

mismatched points are removed using the SIFT feature attributes, k_d can take a smaller value, that is, the set limit distance is further reduced. d_{max} is the maximum value of the pole line distance, and the iterative operation satisfies

$$d_i^m < d_{max},\ i \in [1, N^m], \tag{15}$$

where d_i^m is the distance value of polar line i, and N^m represents the remaining matching points of the m operation. The value of d_{max} directly determines the number of iterations. The smaller the value of d_{max}, the more accurate is the pole fitting and the higher the accuracy of matching points.

3.2 Matching algorithm

The matching algorithm is an essential part of the 3D reconstruction. The dense matching processing is implemented with the region-growing algorithm, as shown in Fig. 3. It was first applied in the field of image segmentation. It uses the continuity of pixel arrangement, selects a small number of precise matching points as the starting point, and then propagates in the region. When the set conditions are met, the matching relationship is propagated to other points. The most important part of the region's growing matching algorithm is seed point selection and region growth.

The process of applying the region-growing dense matching method is described as follows. For seed point selection, first, the feature points are matched by the SIFT algorithm. Second, the

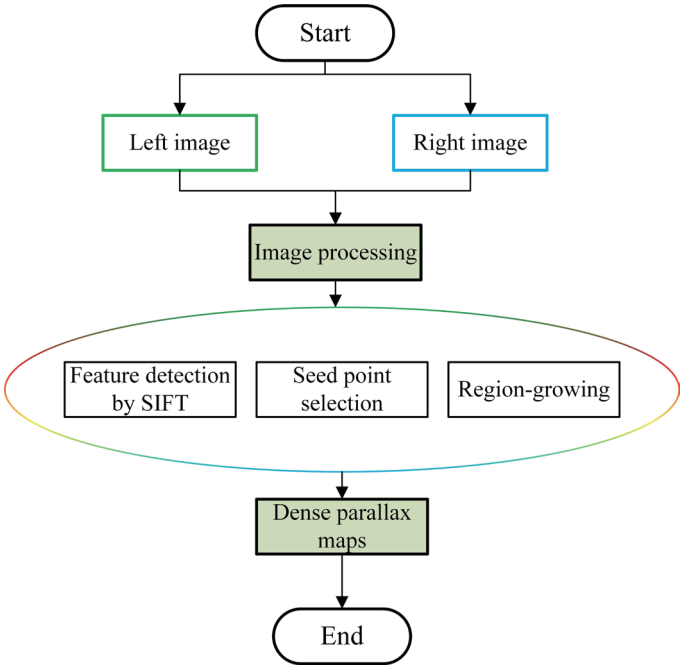


Fig. 3. (Color online) Region-growing dense matching flow.

Random Sample Consensus (RANSAC) algorithm is used to purify the initial matching results. Finally, the matching points with high reliability are selected as seed points.

The key of the region-growing algorithm is to select an exact matching point as the starting point and spread the matching to the entire region.[20] Therefore, the search range for N^m can be narrowed down within the smaller value of point P. Window sizes of 3 × 3 and 4 × 4 are usually selected, and the pixel similarity in the window is calculated. We use the difference of pixels in a window in a color image to calculate the pixel similarity. When the similarity reaches the maximum, the selected point is the matching point.[21] The strategy map for regional growth is shown in Fig. 4.

4. Proposed Point Cloud Stitching Method

4.1 Point cloud acquisition

The point cloud features are of two types: point cloud local feature description and point cloud topological feature description.[22] The extraction of local features depends on the storage format of point cloud data. In the case of many noise points, the reliability of the local feature of the point cloud is low. The topological feature description is constructed by using all points in the point cloud, and such features contain more information than local features. Our method uses the local feature description of the point cloud. Surface normal features are one of the essential properties of a geometric surface. Normal features can be extracted using the surface feature extraction technique. Extracting the normal of a point on the surface is equivalent to estimating the normal of the plane tangent to the surface, which can be summarized as the least-squares plane fitting estimation problem. The extraction of normal features is as follows.

For each point P_i, the covariance matrix is shown in Eq. (16). The number of points adjacent to point P_i is represented by k, $\overline{P}$ is the 3D centroid of the nearest neighbor, λ_j is the j feature value in the covariance matrix, and the corresponding j feature vector is $\vec{v}_J$.

$$\begin{aligned} C &= \frac{1}{k}\sum_{i=1}^{k}\left(P_i - P\right)\cdot\left(P_i - P\right)^T \\ C\cdot\vec{v}_J &= \lambda_i\cdot\vec{v}_i,\ j \in \{0, 1, 2\} \end{aligned} \tag{16}$$

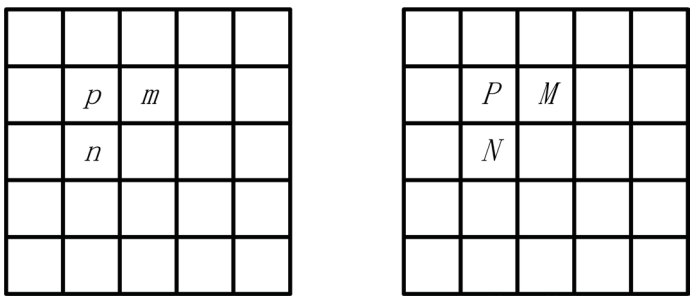


Fig. 4. Strategies for region growing.

The point V_P is known, and this thorny problem can be solved by using the viewpoint direction. Therefore, all the normals $\vec{n}_i$ are oriented, and the normal direction must be consistent with the viewpoint direction V_P, i.e., Eq. (17) must be satisfied.

$$\vec{n}_i \cdot \left(v_P - P_i\right) > 0 \tag{17}$$

4.2 Point cloud stitching

Point cloud sets *cloud* 1 and *cloud* 2 contains coincidence points. The coordinate system of all 3D points in *cloud* 1 is $O_1X_1Y_1Z_1$, and the coordinate system of all 3D points in *cloud* 2 is $O_2X_2Y_2Z_2$. By defining in this way, *cloud* 1 and *cloud* 2 can be spliced. Let the coordinates of a point in the coordinate system $O_1X_1Y_1Z_1$ be (X_1, Y_1, Z_1), and the coordinates in the coordinate system $O_2X_2Y_2Z_2$ be (X_2, Y_2, Z_2). Therefore, the conversion relationship between (X_2, Y_2, Z_2) and (X_1, Y_1, Z_1) is as shown in Eq. (18).

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix}^T = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix}^T + \begin{bmatrix} a_{41} \\ a_{42} \\ a_{43} \end{bmatrix}^T = R \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} + t \tag{18}$$

The steps for finding the optimal R and T using the algorithm for matching are as follows. After determining the corresponding points in the initial point set, it is assumed that the set of matching points is P in *cloud* 1, the number of points included in P is N_p, and the point cloud set in *cloud* 2 is X. The number of points in the set X is N_x, and $N_x = N_p$. We calculate the sum of squares of the smallest Euclidean distances of all the nearest point pairs using Eq. (19) and find the corresponding rotation matrix R and the translation matrix t. To find the minimum value of $f(q)$, we determine the point set N_p and the center of gravity of the point set N_x using Eqs. (20) and (21), respectively.

$$f(q) = \frac{1}{N_p} \sum_{i=1}^{N_p} \left\| x_i - R(q_R) p_i - q_t \right\| \tag{19}$$

$$u_p = \frac{1}{N_p} \sum_{i=1}^{N_p} p_i \tag{20}$$

$$u_x = \frac{1}{N_x} \sum_{i=1}^{N_x} x_i \tag{21}$$

The centers of gravity u_p and u_x are used to obtain the set N_p and the covariance matrix of set X using Eq. (22).

$$\begin{aligned}\sum px &= \frac{1}{N_p}\sum_{i=1}^{N_p}\left[\left(p_i - u_p\right)\left(x_i - u_x\right)^T\right] \\ &= \frac{1}{N_p}\sum_{i=1}^{N_p}\left[p_i x_i^T\right] - u_p u_x^T\end{aligned} \tag{22}$$

The covariance matrix can be used to construct the symmetric matrix given by Eq. (23), where the trace of the matrix Σpx is $tr(\Sigma P,x)$. $\Delta = [A_{23}, A_{31}, A_{12}]$ is the identity matrix, where $A_{ij} = (\Sigma P,x - \Sigma^T P,x)_{ij}$.

$$Q(\sum P,x) = \begin{bmatrix} tr(\sum P,x) & \Delta^T \\ \Delta & \sum P,x + \sum{}^T P,x - tr(\sum P,x) I_{3\times 3} \end{bmatrix} \tag{23}$$

The eigenvalue of the covariance matrix given by Eq. (23) is calculated, and the eigenvector $q_r = [q_0, q_1, q_2, q_3]$ is the eigenvector associated with the largest eigenvalue, and according to the eigenvector, the rotation matrix can be calculated as

$$R(q_R) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 - q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}. \tag{24}$$

The obtained $R(q_R)$ can be obtained from

$$q_t = u_x - R(q_R) u_p. \tag{25}$$

After the optimal rotation matrix R and translation factor t are obtained, the points in point set X are substituted into Eq. (22), and after the coordinate transformation, a new point set M is obtained. If the sum of the squares of the distances of point sets M and N_p is less than a given threshold, the iterative calculation is ended. Otherwise, M is taken as the point set N_x, and the above steps are repeated until the sum of the squares of the distances from point set M is less than the given threshold.

Let the coordinate system of all 3D points in *cloud* 1 be $O_1X_1Y_1Z_1$, the coordinate system of all 3D points in *cloud* 2 be $O_2X_2Y_2Z_2$, and there are certain overlapping areas of point cloud sets *cloud* 1 and *cloud* 2. To put it simply, the basic idea of splicing *cloud* 1 and *cloud* 2 is to convert the 3D point coordinates in the coordinate system $O_2X_2Y_2Z_2$ into the coordinate system $O_1X_1Y_1Z_1$ using the point cloud coordinates of the two-point cloud coincident regions. Let a point in the coincident region of *cloud* 1 and *cloud* 2 be (X_1, Y_1, Z_1) in the coordinate system $O_2X_2Y_2Z_2$ and (X_2, Y_2, Z_2) in the coordinate system $O_2X_2Y_2Z_2$. Therefore, the conversion relationship between (X_2, Y_2, Z_2) and (X_1, Y_1, Z_1) is as shown in Eq. (26).

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix}^T = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix}^T + \begin{bmatrix} a_{41} \\ a_{42} \\ a_{43} \end{bmatrix}^T = R \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} + t \tag{26}$$

R is a rotation matrix, and t is a translation factor. Therefore, all the 3D point coordinates in the coordinate system $O_1X_1Y_1Z_1$ can be converted into the coordinate system $O_1X_1Y_1Z_1$ as long as the optimal rotation matrix R and the translation factor t are estimated by 2D image matching, thereby completing the point cloud stitching to generate a panorama.

In the 3D reconstruction based on the bifocal monocular image, the rotation matrix R and the translation vector t in the matching process are obtained when performing image matching on the two-dimensional bifocal image. During the image acquisition, the bifocal image is acquired using the zoom camera, and although the Z value is changed, the focal length ratio is similar to the image scaling ratio. The large focal length in the test image at t_1 is reduced in accordance with the focal length ratio based on the image center. Similarly, the same processing is performed on the test image at time t_2. The bifocal image taken at time t_2 is obtained only by moving the rotation angle x, y of the monocular camera. The Z coordinate of the pixel obtained by the 3D reconstruction can be regarded as constant. After this processing, the R and t in the image matching process can be used to estimate the value of the point cloud stitching. Finally, the point cloud stitching is realized.

5. Experiment and Discussion

5.1 Experiment settings

The original image captured by the zoom camera has a resolution of 4758 × 3168, and each image has a size of 4.68 Mb. The square size of the flat template used in the experiment was 30 × 30 mm^2, and the image resolution after compression was 1176 × 784. In this study, the camera focal length variation range is F18–F55, and the images at the focal lengths of F18 and F55 are acquired.

5.2 Calibration

The unit pixel of the focal length is converted to millimeter, and the calibration results of the focal length are shown in Table 1. The calibration results of the main points are shown in Table 2. To determine the accuracy and stability of the calibration, the mean and variance are obtained, respectively. The results are shown in Table 3. This method is based on an image with a resolution of 1176 × 784. The principal point approximation can be understood as the center of the image. For the image dataset, the coordinates of the image are [588, 392]. As shown in Table 3, compared with the calibrated main point mean, at the F18 focal length, there is a deviation of about 10.6 in the X-axis direction and an error of about 3.93 in the Y-axis. Compared with the calibration result at the F55 focal length, there is a deviation of about 22 in the Y-axis

Table 1
Focal length calibration results.

Type	Focal length mean (mm)	Focal length standard deviation (mm)
F18	18.5345	0.0390
F55	53.0607	0.0450

Table 2
Calibration results of the main point.

F18 / Pixel	F55 / Pixel
[576.30557, 386.41879]	[562.88867, 381.68161]
[575.71693, 387.40678]	[566.34844, 375.48787]
[577.44345, 387.14419]	[564.70411, 384.61876]
[577.78978, 388.92651]	[567.35401, 387.00371]
[578.00621, 390.42648]	[567.60030, 387.02564]
[578.33154, 390.59528]	[565.93015, 383.57685]
[577.29677, 389.06431]	[566.85153, 382.37069]
[579.03188, 385.98752]	[566.60772, 376.67035]
[578.00546, 388.07354]	[567.58473, 392.26639]
[575.61311, 386.67946]	[564.05132, 384.40398]

Table 3
Statistical results of the main point.

Type	Main point mean	Main point standard deviation
F18	[577.3541, 388.0723]	[1.0785, 1.5475]
F55	[565.9921, 383.5106]	[1.5259, 4.6803]

Table 4
Calibration results of the distortion coefficient.

Type	*kc*(1)	*kc*(2)	*kc*(3)
F18	−0.1696	0.1450	−0.0014
F55	0.0624	0.3488	−0.0028

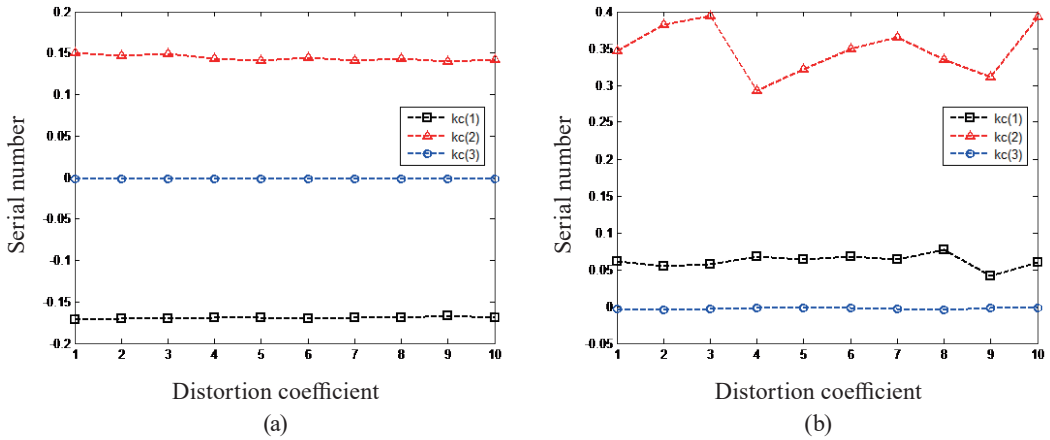


Fig. 5. (Color online) Calibration results of the distortion factor. (a) Curve of the distortion coefficient under F18. (b) Curve of the distortion coefficient under F55.

direction and an error of about 8.48 in the *X*-axis. Therefore, it can be concluded that there is an inherent error between the principal point and the ideal image center.

Figure 5 shows the calibration results and the distortion coefficient curves at the focal lengths of F18 and F55. The result shows that under F18, the curve tends to be stable, and under F55, the curve has an inevitable fluctuation. The distortion coefficients under the focal lengths of F18 and F55 are significantly different. Table 4 gives the mathematical expectation values for the components of the distortion coefficients that are useful for correcting the zoom image.

5.3 Sparse matching

Two sets of images are taken with focal length of F55 and F18 by using a zoom camera, and the initial matching result is shown in Fig. 6. The number of matching points is 168, the scale

confidence space is [2.390, 3.268], and the confidence space is [−11.544, 17.877]. On the basis of the SIFT feature attribute to remove the mismatch, the error matching experiment is performed, and the result is shown in Fig. 7. There is no obvious error matching point in the image, and the number of matching points is 165.

5.4 Dense matching

Image points cannot meet the requirements of subsequent 3D reconstruction point cloud data, so the dense matching of bifocal monocular images is essential. Both methods are used to obtain dense disparity maps, namely, region-matching algorithm and SIFT-based region-growing algorithm. The disparity map obtained using the similarity measure function SAD algorithm is shown in Fig. 8. The result of the region-growing algorithm is shown in Fig. 9.

Comparison of the results shows that, although the outline of the stone pier can be obtained using the SAD algorithm, the background is fuzzy and has no hierarchy. The parallax map obtained using the region-growing algorithm is better than that obtained using the SAD algorithm, and the stone pier contour is clearer and the background level is more obvious.

5.5 Point cloud stitching

We apply monocular stereo vision to obtain 3D point cloud data. Figure 10 shows the point cloud images at t_1 and t_2. The result of point cloud splicing is shown in Fig. 11. The results show that the target object is transparent, but the background is blurred. For the binocular stereo vision principle and the monocular stereo vision, the ICP-based point cloud splicing is also implemented, and the rotation matrix R and the translation factor T are obtained in the image matching. The result is shown in Fig. 12. Our results are more precise than those of state-of-the-art methods, and an efficient rough outline can be obtained, whereas the other effects are more ambiguous.

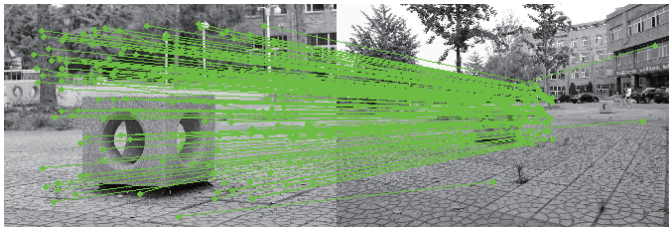


Fig. 6. (Color online) Image initial matching result.

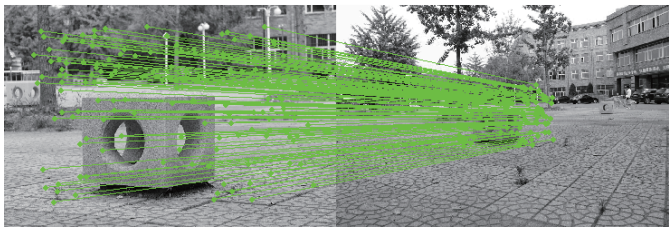
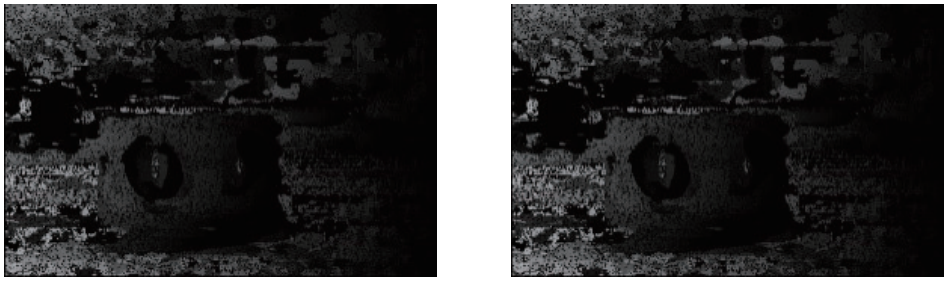
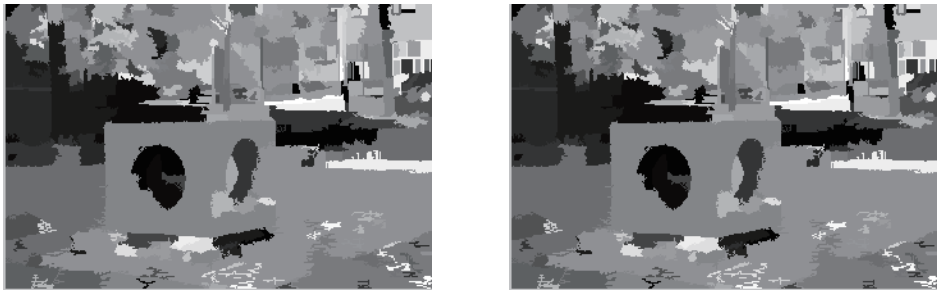


Fig. 7. (Color online) Polar distance-based result.



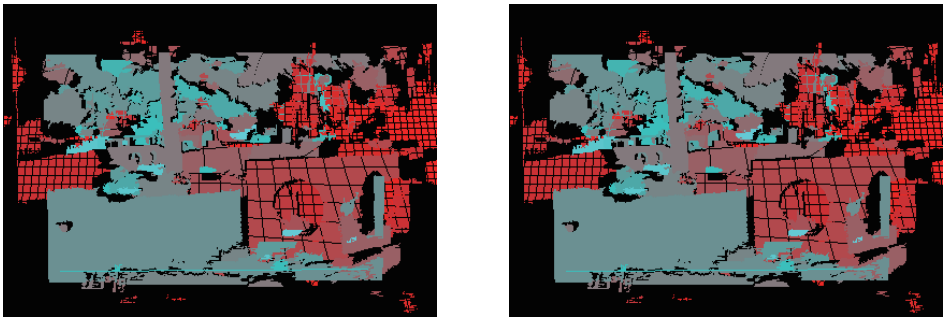
(a) (b)

Fig. 8. (Color online) Parallax map based on SAD algorithm. (a) Disparity map at time t_1. (b) Disparity map at time t_2.



(a) (b)

Fig. 9. (Color online) Disparity map based on region-growing algorithm. (a) Disparity map at time t_1. (b) Disparity map at time t_2.



(a) (b)

Fig. 10. (Color online) Point cloud images at t_1 and t_2. (a) Point cloud display at t_1. (b) Point cloud display at time t_2.

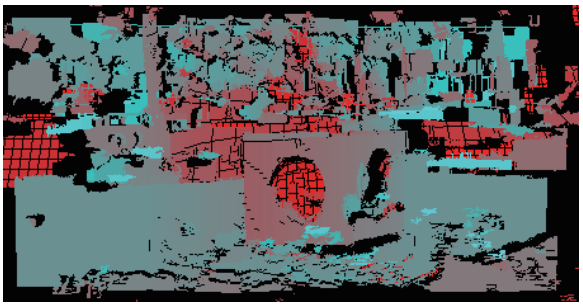


Fig. 11. (Color online) Image based on monocular stereo visual point cloud mosaic results.

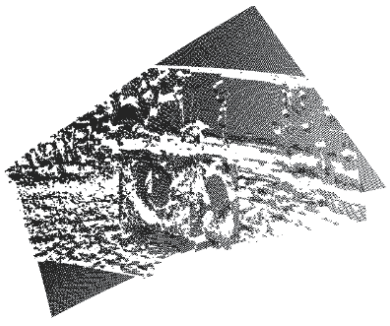


Fig. 12. Image based on binocular stereo visual point cloud mosaic results.

Table 5
Calibration results of the distortion coefficient.

Methods	Average time (s)	Number of points
ICP-based method	**0.395**	1370678
NDT-based model	1.201	5667982
Our proposed method	0.972	**6258985**

Furthermore, to comprehensively analyze the performance of the model, we thoroughly evaluate our proposed method on the public dataset, as shown in Table 5. The KITTI dataset consists of different data collected in many real applications and is the benchmark for autonomous vehicles. This study uses the same sequence collected at parking scenarios to test our proposed method and other state-of-the-art methods. We consider the following state-of-the-art methods: 1) an ICP-based method was proposed for 3D point cloud stitching by introducing the OCT model; 2) a repaid NDT-based model was proposed by using the NARF and FPFH methods.[23,24] As shown, although the ICP-based method achieved the highest speed compared with other algorithms, it had poor ability to complete the stitching. Our proposed method achieved the balance between speed and point cloud density because we had the most point clouds.

6. Conclusion and Future Work

In this paper, to reconstruct a 3D panorama model for autonomous vehicles, a novel point cloud stitching method based on monocular stereo vision is proposed. The main goal is to reduce costs and achieve better results than binocular-based systems. The offline calibration method is redesigned to analyze the internal and external parameters of the camera. According to the characteristics of the zoom image, the SIFT algorithm is redesigned to extract and match data features. Next, a dense matching method based on the bifocal monocular image is proposed, introducing the region-growing algorithm and the ICP algorithm to increase the matching density of the bifocal monocular image. The results show that the proposed method achieved an excellent balance in precision, speed, and completeness.

A high-speed visual acquisition system will be designed in future work, which will focus on improving the ability of 3D point cloud acquisition. This is because the vehicle and the sensor move at a high speed in most cases, leading to poor feature extraction.

Acknowledgments

The authors would like to acknowledge support from the following projects: LiaoNing Province Higher Education Innovative Talents Program Support Project (Grant No. LR2019058); LiaoNing Province Joint Open Fund for Key Scientific and Technological Innovation Bases; LiaoNing Revitalization Talents Program (Grant No. XLYC1902095); Shenyang Institute of Automation, State Key Laboratory of Robotics Foundation (Liaoning Province Key Technology Innovation Base Joint Open Fund); National Natural Science Foundation of China (Grant Nos. 52075530, 51575412, 51575338, U1609218 and 51575407); CAS Inter-disciplinary Innovation Team (Grant No. JCTD-2018-11); AiBle project co-financed by the European Regional Development Fund.

References

1 B. Yang, J. Kostková, J. Flusser, T. Suk, and R. Bujack: Pattern Recognit. **74** (2018) 110. https://doi.org/https://doi.org/10.1016/j.patcog.2017.09.004
2 C. Mineo, S. G. Pierce, P. I. Nicholson, and I. Cooper: J. Comput. Des. Eng. **4** (2017) 192. https://doi.org/https://doi.org/10.1016/j.jcde.2017.01.002
3 M. Berger, A. Tagliasacchi, L. M. Seversky, P. Alliez, G. Guennebaud, J. A. Levine, A. Sharf, and C. T. Silva: Comput. Graphics Forum **36** (2017) 301. https://doi.org/10.1111/cgf.12802
4 Q. Yang: IEEE Trans. Pattern Anal. Mach. Intell. **36** (2014) 1026. https://doi.org/10.1109/tpami.2013.186
5 N. Einecke and J. Eggert: Proc. Int. Conf. Computer Vision Theory and Applications-Volume 2: VISAPP (SCITEPRESS, 2013) 189–198.
6 J. Kowalczuk, E. T. Psota, and L. C. Perez: IEEE Trans. Circuits Syst. Video Technol. **23** (2013) 94. https://doi.org/10.1109/TCSVT.2012.2203200
7 J. L. Schönberger, F. Radenović, O. Chum, and J. Frahm: Proc. 2015 IEEE Conf. Computer Vision and Pattern Recognition (IEEE, 2015) 5126–5134.
8 Q. Huang, H. Wang, and V. Koltun: ACM Trans. Graphics **34** (2015) 1. https://doi.org/10.1145/2766890
9 B. Payne, J. Lay, and M. Hitz: Proc. 2014 ACM Southeast Regional Conf. (ACM SE, 2014) 1–5.
10 D. Eigen, C. Puhrsch, and R. Fergus: Depth Map Prediction from a Single Image using a Multi-Scale Deep Network **3** (2014). arXiv:1406.2283
11 C. S. Fraser: Photogramm. Eng. Remote Sens. **79** (2013) 381. https://doi.org/10.14358/pers.79.4.381
12 F. Li, H. Sekkati, J. Deglint, C. Scharfenberger, M. Lamm, D. Clausi, J. Zelek, and A. Wong: IEEE Trans. Comput. Imaging **3** (2017) 74. https://doi.org/10.1109/TCI.2017.2652844
13 J. Yu, H. Gao, W. Yang, Y. Jiang, W. Chin, N. Kubota, and Z. Ju: IEEE Access **8** (2020) 43243. https://doi.org/10.1109/ACCESS.2020.2977856
14 Z. Ju, X. Ji, J. Li, and H. Liu: IEEE Syst. J. **11** (2017) 1326. https://doi.org/10.1109/JSYST.2015.2468231
15 F. Hanqi, K. Dexing, and L. Jinhong: Proc. 2015 Int. Conf. Intelligent Systems Research and Mechatronics Engineering (Chair, 2015) 283–292.
16 Y. Yingyun, L. Qian, N. Lipi, and Z. Qin: 5th Int. Conf. Graphic and Image Processing (SPIE, 2014) 90690W: 1–7.
17 P. Tanskanen, K. Kolev, L. Meier, F. Camposeco, O. Saurer, and M. Pollefeys: Proc. 2013 IEEE Int. Conf. Computer Vision (IEEE, 2013) 65–72.
18 P. Lasang, S. M. Shen, and W. Kumwilaisak: Proc. 2014 IEEE 4th Int. Conf. Consumer Electronics Berlin (IEEE, 2014) 331–334.
19 V. Anthony, L. Heng, Z. Yang, T. Isaac, A. G.-M. Gustavo, X. Di, K. N. Daniel, C. Changchen, S. B. Nicolas, G.-T. Andres, J. Hae Won, R. Jacob, and B. Julie: Proc. Zoom Lenses V (SPIE, 2015) 95800L: 1–8.
20 S. Angelina, L. P. Suresh, and S. H. K. Veni: Proc. 2012 Int. Conf. Computing, Electronics and Electrical Technologies (IEEE, 2012) 970–974.
21 T. Takanashi and J. Shin: J. Converg. Inf. Technol **7** (2012) 152. https://doi.org/10.4156/jcit.vol7.issue16.18
22 J. Liu, J. Zhu, J. Yang, X. Meng, and H. Zhang: 8th Int. Conf. Digital Image Proc. (ICDIP, 2016) 100334D: 1–5.
23 X. Wang, Z. L. Zhao, A. G. Capps, and B. Hamann: Multimedia Tools Appl. **76** (2017) 6843. https://doi.org/10.1007/s11042-016-3302-9
24 C. Mai, L. Zheng, and M. Li: Trans. Chin. Soc. Agric. Eng. **31** (2015) 137. https://doi.org/10.11975/j.issn.1002-6819.2015.z2.021

About the Authors

Hongwei Gao received his Ph.D. degree in the fields of pattern recognition and intelligent systems from Shenyang Institute of Automation (SIA), Chinese Academy of Sciences (CAS) in 2007. He is with the School of Automation and Electrical Engineering, Shenyang Ligong University, Shenyang 110159, China, and also the State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China.
(ghw1978@sohu.com)

Jiahui Yu received his B.S. and M.S. degrees in intelligent systems from Shenyang Ligong University, China, in 2017 and 2019, respectively. Since 2019, he has been working towards a Ph.D. degree at the University of Portsmouth, U.K. He has published over 10 papers in journals and conference proceedings. His current research interests include machine intelligence, pattern recognition, and human–robot/computer interaction and collaboration. (jiahui.yu@port.ac.uk)

Jian Sun received his B.S. and M.S. degrees in control engineering from Shenyang Ligong University, China, in 2016 and 2019, respectively. Since 2019, he has been working towards a Ph.D. degree at Shenyang Ligong University, China. His current research interests include machine learning and digital image processing. (jiayou2017sj@163.com)

Wei Yang received his B.S. degree in automation and his M.S. degree in automation and electrical engineering (detection technology and automatic equipment) from Shenyang Ligong University, China, in 2016 and 2021, respectively. His research interests include deep learning, pattern recognition and digital image processing. (xcfyangwei@163.com)

Yueqiu Jiang received her B.S. and M.S. degrees in computer science from Shenyang Ligong University, China, in 1998 and 2001, respectively, and her Ph.D. degree in computer science from Northeast University, China, in 2004. She worked as a lecturer from March 2004 to August 2006 and as an associate professor from September 2006 to August 2010 in the School of Science, Shenyang Ligong University. Currently, she is a professor in the School of Information Science and Engineering, Shenyang Ligong University. Her research interests include image processing, multimedia applications, satellite communications, and signal processing. (yueqiujiang@sylu.edu.cn)

Lei Zhu is currently a professor of computer science at the College of Automation (Artificial Intelligence), Hangzhou Dianzi University, China. His current research interests include signal processing, machine learning, and computational intelligence. (zhulei@hdu.edu.cn)

Zhaojie Ju received his B.S. degree in automatic control and his M.S. degree in intelligent robotics from Huazhong University of Science and Technology, China, and his Ph.D. degree in intelligent robotics from the University of Portsmouth, U.K. He held research appointments at the University College London, London, U.K., before he started his independent academic position at the University of Portsmouth in 2012. He has authored or coauthored over 200 publications in journals, book chapters, and conference proceedings and received four Best Paper Awards and one Best AE Award in ICRA2018. His research interests include machine intelligence, pattern recognition and their applications on human motion analysis, multi-fingered robotic hand control, human–robot interaction and collaboration, and robot skill learning. He is an associate editor of several journals, such as IEEE Transactions on Cybernetics, IEEE Transactions on Cognitive and Developmental Systems, and Neurocomputing.
(zhaojie.ju@port.ac.uk)