The Journal of Engineering

The 2nd 2018 Asian Conference on Artificial Intelligence Technology (ACAIT 2018)

IET Journals
The Institution of Engineering and Technology

3D vision measurement for small devices based on consumer sensors

eISSN 2051-3305
Received on 19th July 2018
Accepted on 26th July 2018
E-First on 24th October 2018
doi: 10.1049/joe.2018.8330
www.ietdl.org

Qian Zhang[1], Juanhui Tu[2] ✉, Zhiyong Li[1], Hong Liu[2]

[1]College of Computer Science and Electronic Engineering, Hunan University, Changsha, People's Republic of China
[2]Key Laboratory of Machine Perception, Peking University, Shenzhen Graduate School, Shenzhen, People's Republic of China
✉ *E-mail: juanhuitu@pku.edu.cn*

Abstract: High-precision, low-cost three-dimensional (3D) space measurement and positioning technology is desperately needed in wide applications. This study analyses the key technologies in the recognition of the devices to achieve the requirement of device recognition and capture on the production line. A 3D measurement algorithm for the small devices based on the consumer-level sensors is proposed in this study. Histogram of gradients feature is used to classify the devices, and structure light is used to get the depth data of the devices. Object extraction and Euclidean cluster segmentation are used to analyse the depth data, in order to determine their positions and orientations. In the database built on iPhone X, the accuracy of category identification reached 0.97, and the measurement error of angle is small. The results show that the proposed method is feasible and can be applied to the recognition and position of the devices.

1 Introduction

With the rapid development of the industry, traditional two-dimensional (2D) technology is limited in accuracy and distance measurement, especially in complex object recognition, dimension measurement, and interactive applications. The demand for 3D vision technology becomes more and more requisite because high-precision, low-cost 3D space measurement, and positioning technology are desperately needed in many parts of the society. In the application of industry, this technology can not only classify the devices but also determine their positions, size and orientations in high precision.

Although the current solution has been better for positioning accuracy, such as laser scanning measurement systems, structure light 3D scanners, the cost has remained high. 3D visual technology also appears on mobile phones. Apple's iPhone X uses a structure light-based 3D machine vision solution on its front camera, which is captured by the camera by projecting specific light information onto the surface of the object. The changes of the optical signal caused by the object are used to calculate the position, depth and other information of the object so that the entire 3D space can be restored. Therefore, real-time 3D information acquisition is achieved by using consumer electronic devices.

To the best of our knowledge, there are no studies using consumer-level sensors to measure components on industrial production lines. Therefore, in this study, the consumer-level sensor is used to measure devices on industrial production lines. After acquiring 3D information of the small device, the distance and angle can be measured through the TrueDepth camera's structure light technology. First, we establish the sample to be detected, extract its histogram of oriented gradients (HOG) [1] features of the sample, and put it into the support vector machine (SVM) classifier for training. Then, the trained classifier is used for device classification. The target device can be extracted when its category is identified. Also, then we can find the plane of the object from the marked point and get its normal vector. Therefore, the angle between the object and the plane is obtained.

2 Related works

This study mainly focused on how to obtain the 3D information of the detected object from the consumer-level 3D vision system software platform and performs visual measurement on small devices.

In the current industrial measurement system, in order to achieve high-precision inspection and adjustment of the devices, precision measurement including 3D coordinates is performed, and the following systems are available [2].

The laser scanning measurement system [3] can quickly obtain the coordinates of the spatial position of each sample point on the surface of the object to obtain a set of points representing the entity, which is 'point cloud' [4, 5]. It has the advantage of directly reflecting the real-time, dynamic, and real morphological characteristics of the detected object. There are companies that produce 3D laser scanners at home and abroad, such as Trimble's Trimble GX200, which is currently a relatively advanced device. The point accuracy is up to ±6 mm at a distance of 50 m, ±12 mm at 100 m, and a distance accuracy of ±4 mm at 50 m and ±7 mm at 100 m. The angle accuracy can reach $\leq \pm 12''$ in both horizontal and vertical directions.

A structure light 3D scanner is a non-contact measuring device based on a line or surface structure light projection measurement principle. It has fast measurement speed, ability to collect surface data on a large scale, high accuracy, and unique calibration technology can make single- sided accuracy up to 4 μm. It has been widely used in the contour measurement of large-scale complex parts such as airplanes, automobiles, and ships.

These methods are a trend in industrial measurement, but they are all relatively expensive. 3D visual products also have been tried in the consumer market, such as Microsoft Kinect [6, 7] game accessories and Intel RealSense somatosensory accessories. However, because of the lack of good application scenarios and immature technology, the current consumer-level 3D visual market is cheerless.

At present, there are also 3D visuals on personal mobile phones. For example, iPhone X uses a structured light based on a 3D machine vision solution on its front camera. Huawei also introduced a Jupiter X, a speckle-structure light mobile phone accessory with a 'point cloud depth camera.' 'Point cloud depth camera' recognition accuracy reaches the sub-millimetre level to achieve high-precision and safety face recognition. It can achieve 3D face model and face recognition, as well as 3D facial expression control and 3D small object modelling.

Our main contributions are as follows: (i) a low-cost consumer-level sensor is used to collect 3D data of the object. (ii) The data we collect is used to achieve object recognition, distance and angle measurements. The results show that our method can be applied to the recognition and position of the devices.

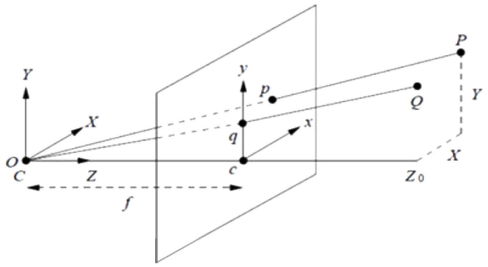


Fig. 1 *Spatial points [x, y, z] correspond to pixels [u, v, d]*

3 Methodology

3.1 Data collection

This paper uses the iPhone 3D vision sensor TrueDepth camera to collect RGB pictures and depth information. It can effectively capture the depth information of the target, and directly obtain the position of the target object relative to the camera.

Through the TrueDepth camera, the AVFrame class in the ARKit framework and the AVDepthData class in the AV-Foundation framework of the development framework of the Apple platform were used to obtain RGB images and depth information synchronously.

3.1.1 Convert RGB image and depth data to point cloud data: The correspondence between a space point [x, y, z] is shown in Fig. 1 [8], and its pixel coordinates [u, v, d] in the image (d refers to depth data) is as follows:

$$\begin{aligned} u &= \frac{x \cdot f_x}{z} + c_x, \\ v &= \frac{y \cdot f_y}{z} + c_y, \\ d &= z \cdot x, \end{aligned} \tag{1}$$

where f_x and f_y refer to the focal length of the camera on the x and y axes, c_x and c_y refer to the camera's aperture centre, s refers to the depth map scaling factor.

Given (u, v, d), the derivation of (x, y, z) is as follows:

$$\begin{aligned} z &= d/s, \\ x &= (u - c_x) \cdot z/f_x, \\ x &= (v - c_y) \cdot z/f_y. \end{aligned} \tag{2}$$

Given f_x, f_y, c_x, and c_y according to the above formula, a point cloud can be built.

3.2 Proposed algorithm

3.2.1 Object recognition: Feature extraction. In this study, our object detection based on the HOG algorithm uses a 64 × 64 pixel window to scan in a raster image on a frame of the image. The horizontal and vertical scanning steps are 8 pixels. The window is divided by an 8 × 8 pixel cell to form 8 × 8 = 64 cells. We treat four adjacent cells as a block, and one window contains ((64–16)/8 + 1) × ((64–16)/8 + 1) = 7 × 7 = 49 pixel blocks.

SVM [6, 9] is a popular linear classifier for classifiers based on statistical learning theory [7]. The idea of classification is given to a sample set containing positive and negative examples. The purpose of the SVM is to find a hyperplane. Samples are split based on positive and negative examples.

The two-category SVM multi-class classification methods mainly include 'one-versus-one' [10], 'one-versus-rest' [11, 12], directed acyclic graph-SVM [13] and error-correcting output codes [14]. 'one-versus-rest' is a widely used multi-class classify-cation method. For the 'one-versus-rest' algorithm, if there are k samples, then k binary classifiers need to be constructed, and each classifier is used to separate one type from the rest. During the training, one of them is a positive class, and the remaining $k-1$ is a negative class. In the judgment, the sequence of the samples to be tested is obtained through k binary classifiers to obtain k output values $f_i(x) = \text{sgn}(g_i(x))$, $i = 1, 2, \ldots, k$. If there is only one +1 in the decision result, the sample class to be detected is the positive class of the corresponding classifier. If there is more than one +1 in the decision result, i.e. the phenomenon of classification overlap occurs, the decision function value of the classifier whose output is +1 need to be compared. The positive class of the classifier with the largest value represents the class of the sample to be detected. If the judgment is −1, the sample is considered inseparable.

In this study, the 'one-versus-rest' SVM is used to classify and identify small devices based on HOG feature descriptors.

3.2.2 Object extraction: In this section, background subtraction [15] is used to obtain 3D data of the object.

After the camera is fixed, the average of the depth information of the previous frames is taken as the initial background image B before the device is placed.

After the background image depth information is subtracted from the current frame depth information, thresholding is performed to extract 3D data of the object.

Get the depth information of the current frame that is used to obtain the 3D data of the object.

3.2.3 Euclidean cluster segmentation: In this section, we describe how to use Euclidean clustering to segment the point clouds.

The point cloud segmentation is based on the characteristics of space, geometry and textures to divide the point cloud so that the point cloud within the same partition has similar characteristics. The effective segmentation of point clouds is the premise for feature extraction of target objects. Point cloud segmentation algorithms which are commonly used include random sampling consistency algorithms, region growth segmentation methods, and so on [16–18].

Random sampling consistency segmentation can only segment a model from a specific point cloud dataset and does not apply to cloud segmentation of site points with multiple point cloud clustering. The regional growth segmentation can segment the clustering of point clouds well, but the algorithm has a large time complexity and does not apply to real-time scenes. The segmentation algorithm based on the minimum spanning tree algorithm is very cumbersome. In the case of too many points in the point cloud dataset, the algorithm is inefficient. Considering the two factors of time complexity and segmentation effect, this study uses the Euclidean cluster segmentation algorithm to segment the point cloud data in the scene. The Euclidean cluster segmentation algorithm is relatively easy to understand, i.e. the points whose distance is within a certain threshold are classified as one type.

Examine m data points and define some kind of sparseness relationship between points to divide clusters. The Euclidean-style cluster segmentation defines the affinity property of the Euclidean distance [19]

$$d(p_i, q_i) = \sqrt{\sum_{k=1}^{n} (p_{ik} - q_{ik})^2}, \tag{3}$$

where p_i, $q_i \in P$, and P is a set of points.

Euclidean segmentation algorithm. The algorithm steps are as follows:

(i) Create a k-dimensional tree for the input point cloud data set as P.

(ii) Set an empty cluster C and a queue Q to store the set of points to be examined.

(iii) For any point p_i in point cloud P, perform the following steps:

Put p_i in the current queue Q.

For each point p_i belongs to Q: with p_i as the centre, a set of points whose distance is less than the threshold r is placed into class Q.

If all points in Q have been processed, place all of them in the clustering set and clear queue Q.

(iv) When all the points in the point cloud P have been processed and are in the clusters formed by the cluster set C, the algorithm ends.

3.2.4 Least-square method: In this section, we use the least squares method to obtain a plane's fitting plane.

Let M be the initial plane point cloud, assuming that the fitted plane equation is $P: Ax + By + Cz = 0$, this equation can uniquely represent the plane P, but the representation of P also has different forms, such as $Ax/2 + By/2 + Dz/2 + D/2 = 0$ can also mean P.

To standardise the expression of the space plane equations, it is assumed that the plane is not the origin of the coordinates. Here, the standard space plane equation is defined as (4), where x, y, and z are three coordinate axes, and a, b, and c are the three coefficients of the equation of the spatial planes, respectively

$$p(x, y, z) = ax + by + cz + 1 = 0. \tag{4}$$

Set point as $M\{(x_1, y_1, z_1), (x_2, y_2, z_2), \ldots,(x_n, y_n, z_n)\}$, its best fitting plane should be satisfied e.g. (5)

$$\sum_{i=1}^{n} [p(x, y, z) - p(x_i, y_i, z_i)]^2 = \min, \tag{5}$$

where $p(x, y, z) = 0$, then

$$f = \sum_{i=1}^{n} (ax_i + by_i + cz_i + 1)^2 = \min. \tag{6}$$

To make formula (6) set up, we need to take the partial derivative of a, b, c with respect to 0 and satisfy $(\partial f/\partial a) = 0, (\partial f/\partial b) = 0, (\partial f/\partial c) = 0$, e.g. (7)

$$\begin{aligned} a\sum_{i=0}^{n} x_i^2 + b\sum_{i=0}^{n} x_i y_i + c\sum_{i=0}^{n} z_i x_i &= -\sum_{i=0}^{n} x_i, \\ a\sum_{i=0}^{n} x_i y_i + b\sum_{i=0}^{n} y_i^2 + c\sum_{i=0}^{n} z_i y_i &= -\sum_{i=0}^{n} y_i, \\ a\sum_{i=0}^{n} x_i z_i + b\sum_{i=0}^{n} y_i z_i + c\sum_{i=0}^{n} z_i^2 &= -\sum_{i=0}^{n} z_i. \end{aligned} \tag{7}$$

Then

$$\boldsymbol{QX} = \boldsymbol{K}, \tag{8}$$

where

$$\boldsymbol{Q} = \begin{bmatrix} \sum_{i=0}^{n} x_i^2 & \sum_{i=0}^{n} x_i y_i & \sum_{i=0}^{n} z_i x_i \\ \sum_{i=0}^{n} x_i y_i & \sum_{i=0}^{n} y_i^2 & \sum_{i=0}^{n} z_i y_i \\ \sum_{i=0}^{n} x_i z_i & \sum_{i=0}^{n} y_i z_i & \sum_{i=0}^{n} z_i^2 \end{bmatrix},$$

$$\boldsymbol{X} = [a, b, c]^{\mathrm{T}}, \boldsymbol{K} = \begin{bmatrix} -\sum_{i=0}^{n} x_i \\ -\sum_{i=0}^{n} y_i \\ -\sum_{i=0}^{n} z_i \end{bmatrix},$$

The solution to the coefficient of the plane equation is as follows:

$$\boldsymbol{X} = \boldsymbol{Q}^{-1}\boldsymbol{K}. \tag{9}$$

Fig. 2 *Devices images used in this study*

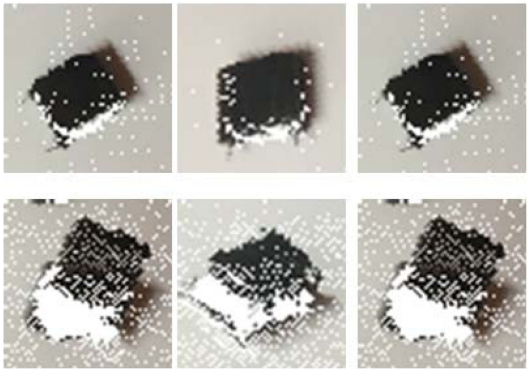


Fig. 3 *Depth data in different views*

The standard deviation of accuracy evaluation is shown as

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n} (\mathrm{d}i - \bar{d})^2}{n-1}}, \tag{10}$$

where di is the distance from a point to a fitting plane; $\bar{d}$ is the average distance from a point to the fitting plane.

For specific parts, four marked points of the plane are calibrated to obtain the plane of the device surface.

After the image is binarised, thresholding is performed to extract the calibration point. Then the contour and its centre can be found. Also, finally, the coordinates of the marker point can be obtained.

After filtering and eliminating the discrete points, the least-square method is used to obtain the fitting plane, and then we the background plane equation can be obtained.

4 Experiments and analysis

In this section, we firstly describe the datasets collected using iPhone X. Then we can evaluate the proposed method on our datasets, and finally report the experimental results and analysis.

4.1 Datasets and protocols

In our experiment, 520 device images with their depth data maps were collected using the TrueDepth camera of the iPhone X in different views or backgrounds. Also, experiments were carried out on the data shown in Figs. 2 and 3. 400 images were chosen as the training dataset, and 120 images as the test dataset from our database.

4.2 Experiment process

In this part, the datasets are used to train the SVM classifier. After the training, our SVM classifier is tested on the test datasets. The resulting confusion matrix is shown in Table 1. The correct recognition rate of device 1 is lower than that of other devices. Also, device 1 is easily misjudged as devices 2 and 3. From Table 1

Table 1 Classification accuracy confusion matrix

①	0.92	0.04	0.04	0.00	0.00	0.00
②	0.00	0.99	0.00	0.00	0.00	0.00
③	0.00	0.02	0.98	0.00	0.00	0.00
④	0.02	0.00	0.04	0.94	0.00	0.00
⑤	0.00	0.00	0.00	0.01	0.99	0.00
⑥	0.00	0.00	0.00	0.00	0.00	1.00
—	①	②	③	④	⑤	⑥

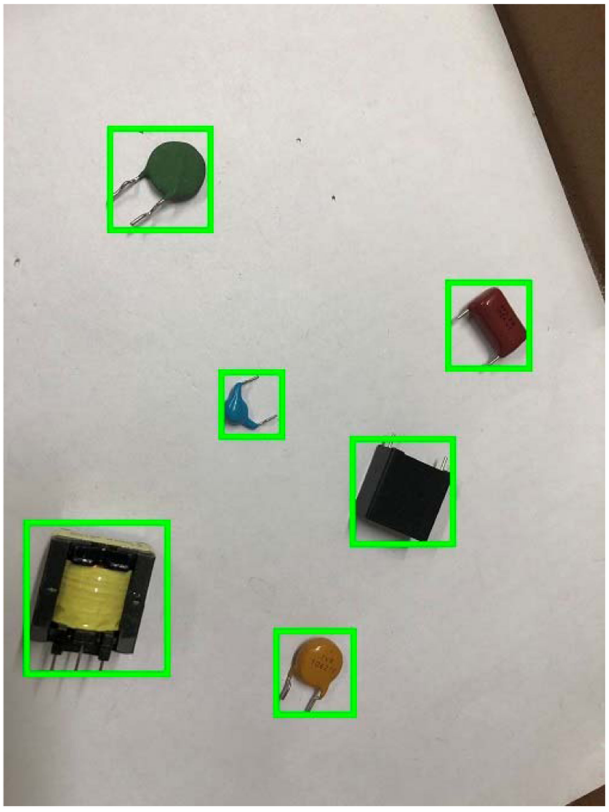


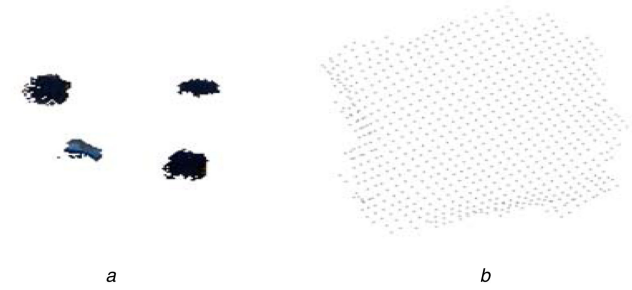
Fig. 4 *Object detected result*

Fig. 5 *Original image and threshold image*

we can find that the correct recognition rate of devices 3, 5, and 6 reaches 0.99. On the whole, the correct recognition rate of the data is 0.97. From Table 1 we can see that the accuracy is high enough to classify the devices, thus the classifier can be used to detect the object from the background. The object detected experiment result is shown in Fig. 4.

We mark the four points on the surface of the part, as is shown in Fig. 5. on the left, the location of the marked points can be obtained through our processing. After detecting the object from the visible light image, the depth data is used to calculate the angle and distance of the object. Each collected point cloud data contains 230,400 sampling points. However, the point cloud data we collect is usually of uneven density. In addition, errors in measurements can produce sparse outliers, making the effect worse. Therefore, the filter in the point cloud library needs to be used to filter the data.

A statistical analysis is performed on the neighbourhood of each point and points that do not meet certain criteria are pruned. Our sparse outlier removal method is based on the calculation of the distance distribution from point to point in the input data. For each



a *b*

Fig. 6 *Pointcloud data*
(a) Captured devices from the depth data map, ***(b)*** Segmented plane

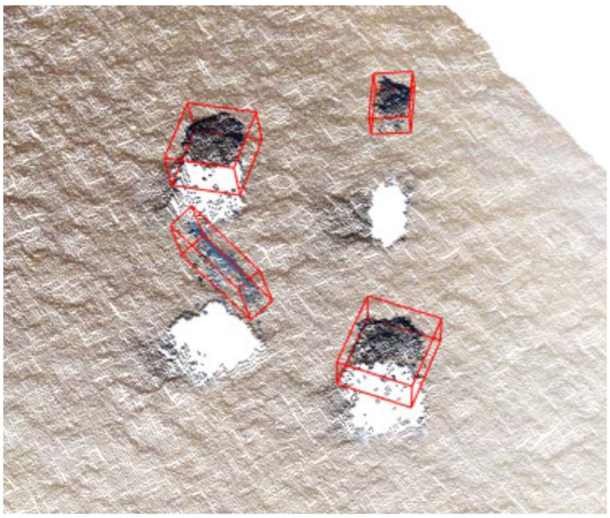


Fig. 7 *Devices' orientation by using the directed bounding box*

point, its average distance to all its neighbours is calculated. The hypothetical result is a Gaussian distribution whose shape is determined by the mean and standard deviation. Points, where the average distance is outside the standard range, can be defined as outliers and can be removed from the dataset. Also, the average distance defined by the global distance mean and variance. Only after the collected cloud data is filtered and denoised can we use the background subtraction method to perform target extraction. Therefore, before filtering the point cloud using the filter provided in the point cloud library, the original data is shown in Fig. 6*a*.

Then, the Euclidean cluster segmentation algorithm is used to segment the target cloud data after extraction to obtain multiple point cloud PCD files. The results obtained after visualisation are shown in Fig. 6*b*.

We calculate the normal vector of the plane from the point on the plane and get the angle between the device and the plane. Then, the bounding box of the object is calculated based on the normal vector of the object surface and display the result. Finally, each device distance and orientation is obtained. The final result is shown in Fig. 7. The experiment results are compared with the real data shown in Table 2. Comparing the above results, we can see that the consumer-level structure light proposed in this study can identify and measure the small devices.

Table 2 Comparison between the experiment data and the real data

device	Device normal vector	Plane normal vector	Theta	Ans	Real Ans
1	(0.002, 0.648, 5.018)	−(0.124, 0.921, 5.217)	3.0898	2.9681°	0°
2	(0.173, −3.08, 2.311)	−(0.124, 0.921, 5.217)	2.0403	63.1020°	60°
3	(0.457, 1.128, 5.688)	−(0.124, 0.921, 5.217)	3.0374	5.9710°	0°
4	(0.540, 0.381, 2.729)	−(0.124, 0.921, 5.217)	2.9678	9.9575°	15°

5 Conclusion

A 3D measurement algorithm for the micro devices based on the consumer-level sensors is proposed. We describe how to use the TrueDepth camera of the iPhone X to obtain the depth data of the device and design a process to classify and segment them using the depth data. Then some points are selected, after morphological processing, the distance and orientation information can be obtained. Experiments show that this method can perform well in the simple scenery. Future works will focus on verifying the effectiveness of our method in wider applications.

6 Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (nos. 61672215 and U1613209).

7 References

[1] Dalal, N., Triggs, B.: 'Histograms of oriented gradients for human detection'. Computer Vision and Pattern Recognition, San Diego, CA, USA, 2005, pp. 886–893

[2] Dongwei, L.: 'Development status of industrial measurement system at home and abroad', *Sci. Technol. Inf.*, 2011, (8), pp. 130–130

[3] Sun, C., Shi, H., Qiu, Y., *et al.*: 'Line-structured laser scanning measurement system for BGA lead coplanarity'. Asia Pacific Conf. on Circuits and Systems, Tianjin, China, December 2000, pp. 715–718

[4] Rusu, R.B., Cousins, S.: '3D is here: point cloud library (PCL)'. Int. Conf. on Robotics and Automation, Marseille, France, September 2011, pp. 1–4

[5] Brostow, G.J., Shotton, J., Fauqueur, J., *et al.*: 'Segmentation and recognition using structure from motion point clouds'. European Conf. on Computer Vision, Marseille, France, October 2008, pp. 44–57

[6] Cortes, C., Vapnik, V.: 'Support-vector networks', *Mach. Learn.*, 1995, **20**, (3), pp. 273–297

[7] Boser, B.E., Guyon, I.M., Vapnik, V.N.: 'Algorithm for optimal margin classifiers', *Comput. Learn. Theory*, 1992

[8] 'Camera models and imaging'. Available at http://www.comp.nus.edu.sg/~cs4243/lecture/camera.pdf, accessed 18 May 2018

[9] Vapnik, V.: 'The nature of statistical learning theory'. Conf. on Artificial Intelligence, Funchal, Madeira, Portugal, October 1995, pp. 988–999

[10] Hsu, C., Lin, C.: 'A comparison of methods for multi-class support vector machines', *IEEE Trans. Neural Netw.*, 2002, **13**, (2), pp. 415–425

[11] Schölkopf, B.: 'A comparison on methods for multi-class support vector machines', *IEEE Trans. Neural Netw.*, 2008, **13**, (2), pp. 415–425

[12] Kressel, U.H.G.: 'Pairwise classification and support vector machines', in Burges, C.J.C., Scholkopf, B., Smola, A.J. (Eds.): '*Advances in kernel methods*' (MIT Press, Cambridge, 1999), pp. 547–553

[13] Platt, J., Cristianini, N., Shawetaylor, J., *et al.*: 'Large margin DAGs for multiclass classification'. Neural Information Processing Systems, Vancouver, British Columbia, Canada, 2000, pp. 547–553

[14] Lingfeng, N.: 'Parallel algorithm for training multi-class proximal support vector machines', *Appl. Math. Comput.*, 2011, **217**, (12), pp. 5328–5337

[15] Kim, K., Chalidabhongse, T.H., Harwood, D., *et al.*: 'Real-time foreground-background segmentation using code-book model', *Real-Time Imaging*, 2005, **11**, (3), pp. 172–185

[16] Woo, H., Kang, E., Wang, S., *et al.*: 'A new segmentation method for point cloud data', *Int. J. Mach. Tools Manuf.*, 2002, **42**, (2), pp. 67–178

[17] Schnabel, R., Wahl, R., Klein, R., *et al.*: 'Efficient RANSAC for point-cloud shape detection', *Comput. Graph. Forum*, 2007, **26**, (2), pp. 214–226

[18] Preetha, M.M.S.J., Suresh, L.P., Bosco, M.J.: 'Image segmentation using seeded region growing'. 2012 Int. Conf. on Computing, Electronics and Electrical Technologies (ICCEET), Nagercoil, Tamil Nadu, India, March 2012, pp. 576–583

[19] Assem, I., Dupont, G.: 'Friezes and a construction of the Euclidean cluster variables', *J. Pure Appl. Algebra*, 2010, **215**, (2), pp. 2322–2340