http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (21): 6123-6129, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.6123.6129

Research on Robot Information Collection System Based on 3D Laser Radar

[1]Wei Chen, [1]Zhongqiang Tang and [3]Houliang Qian
[1,3]Industrial Center, Nanjing Institute of Technology, Jiangsu, Nanjing, 211167, China
[2]School of Automation, Nanjing Institute of Technology, Nanjing, 211167, China

Abstract: Precision of three-dimensional space plays a vital role for the robot to perform tasks accurately. This study designs a camera image acquisition system by using rotating linear laser beam. First, the control of actuator is realized by serial communication and the 2D image is captured from lines to surface, then denoise processing calibration is carried out by using Open CV. By using Irrlicht3D engine, the point cloud data is to be rendered to convert the 2D images to the 3D effect. Robot's collection on external image is achieved through the study of Open CV learning that combined with VC2008.

Key words: Robot, 3D laser radar, information collection

INTRODUCTION

On many occasions, the mobile robot needs to accurately perceive the surrounding environment, not only to avoid obstacles, but also in order to get the accurate information of the surrounding environment, for example, to draw a plane electronic map of the surrounding environment and thus to determine the robot's location. In terms of the majority of mobile robots, "Simultaneous Localization and Mapping (SLAM)" is a very important research topic (Bi *et al*., 2009). For these applications, it's difficult for ultrasonic sonar and infrared ranging sensor to complete the job. There are two problems in sonar: (1) The distance is limited and the larger environment can not be detected fully; (2) Due to crosstalk and the mutual interference of multiple reflections, it would seriously affect the measurement accuracy. The effective distance that infrared ranging sensor is inadequate. The ideal sensor is a kind of Laser Radar using laser scanning range finder sensor which is called Laser Range Finder. However, the Laser Radar is expensive which is of large size, heavy weight with the unit price at above 5000 dollars to 6000 dollars, so that, is not affordable as an ordinary robot (Xu and Zou, 2008). In addition, the working principle of the laser radar is to calculate the distance according to the time difference between the transmitting and receiving of the laser pulse. As the minimum time interval which can be distinguished by photo sensor is limited, therefore, the measurement accuracy can not be less than 5mm under normal circumstances and the accuracy does not decrease with the reduce of measurement distance. There is an urgent need for a high-precision, low-cost, portable substitute product. This study is done which is subject to above application.

This study presents a research on a new type robot 3D ranging radar information collection system based on computer technology, the mobile and proposes a new type of field fast calibration method. The system not only achieves three-dimensional reconstruction of the surrounding environment in functions and the equipment of the system also has features as light weight, easy portability, low cost which has a broad space for development (Gao and Xie, 2005).

SYSTEM DESIGN

This system is intended to design a low-cost 3D laser radar information collection system, specific studies are as follows:

- The principle of laser range finder
- Image acquisition and denoise processing by using Open CV
- The external environment interference with its elimination method (outside light interference)
- Control based on the actuator system
- Program design based on point cloud data processing and computer imaging technology

The system uses a linear laser that scans target object in line each time, rotates the actuator by using Arduino, thus can collect 180-degree image information, this study also analyzes the laser image obtained at certain time to get actual distance of linear laser and render the received point cloud data by Irrlicht 3D engine to achieve the 3D effect. The system can be divided into two functional modules, one is lower computer collection and transmission module, the other is the host computer processing and display module (Zhang, 2005).

Corresponding Author: Wei Chen, Industrial Center, Nanjing Institute of Technology, Nanjing, 211167, China

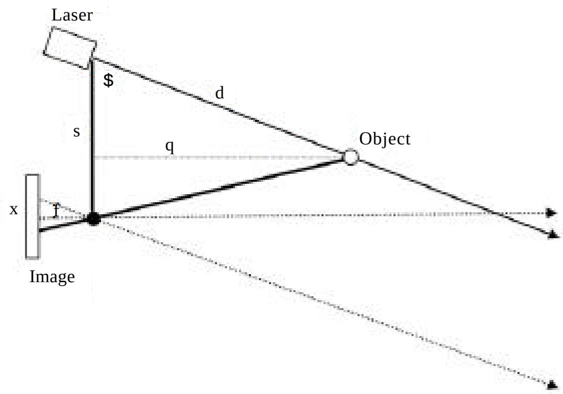

Fig. 1: Principle of triangulation range

Principle of laser range finder: The principle of triangulation range by using single-point laser is as shown in Fig. 1:

The equation of distance of objects from the laser:

$$q = fs/x \quad (1)$$

$$d=q/\sin(\beta) \quad (2)$$

where, x is the only variable need to obtained in the measurement whose meaning is the distance of the imaging of laser spot in the tested object in the camera sensor (CMOS) to one side edge. The distance can be obtained by searching and calculating the pixel coordinates of the laser spot center in the camera screen.

If (1-1) can be rewritten as x = fs/q and according to q, the derivation can be drawn:

$$dq/dx = -q\hat{}2/fs \quad (3)$$

The meaning of (1-3) is, when every time variable x jumps, we can get the relationship between the jump of distance value q and actual distance through our triangulation range equation. As can be seen, when the tested distance is far, each time the pixel from the camera move a unit, the distance value will be increased substantially. In other words: The accuracy and resolution of the triangulation range with get worse with increasing distance (Xiao, 2008). Therefore, to determine the indicators you want to achieve, you need to be clear about: The maximum distance of the ranging (Chen and Liu, 2008).

For linear laser ranging problems, it can be transformed into the calculation of the previous single-point laser range. In terms of the laser line obtained, the algorithm will calculate follow the laser spot X coordinate value pX based on Y-axis and try to seek the distance by the corresponding algorithm. To simplify the problem, the problem of distance of each laser spot in camera photosensitive parallel plane should be given priority (Zhi *et al*., 2012).

As is shown in Fig. 2, the distant plane is the target tested plane with a purple laser spot on it. The near plane is imaging plane of the camera's light-sensitive and after the turnover, he can be seen as a cross-section of the pyramid of target plane to the camera imaging center.

Point P1 in Fig. 2 is at the center of height of the projection image of the camera, in accordance with the definition of the pinhole camera, the distance between the point on the imaging projection P1'and the center of the camera Camera Center should be the camera focal length f.

In Fig. 2, set the distance from P2 projection point P2'to the camera center is f', perpendicular distance of P2 to the baseline d' can be drawn by the following equation:

$$d' = f'baseline/x \quad (4)$$

It is easy to know, f' can be obtained through f:

$$f' = f/\cos(\arctan((P2'.y-P1'.y)/f)) \quad (5)$$

P2'y and P1' y are the actual height of point P2', P1' in imaging element and they can be draw when each pixel coordinate pY is multiplied by the height in pixels. When we get the perpendicular distance d', we should turn it into the actual distance D and you need to know the angle theta between P2-RotationCenter and Baseline. The angle can be calculated by the angle beta of infrared machine and the Baseline based on solid geometry.

After calculating the coordinates of any point of the laser spot on the parallel plane, the problem can be generalized. For any laser projection point in 3D space, you can construct a parallel plane where the point is in and then use the above algorithm to solve.

An array dist [n] will be produced for each ranging sampling. Of which, the dist [i] is the distance of the laser spot at different heights of pixel coordinates for the corresponding images. The value of n is 240 for using camera with resolution of 320×240.

If 3D Scanning of 180 degrees with stepping 1 degree is conducted, then the array of point cloud can be got with a resolution of 180×240. If using a 0.3° step scan in 180 degrees, you get the 600x240 array of point cloud.

Determination and solution of pixel coordinate of the laser point: How to calculate the coordinates of the points of laser from the camera screen is discussed here. Specifically, the following problems should be solved:

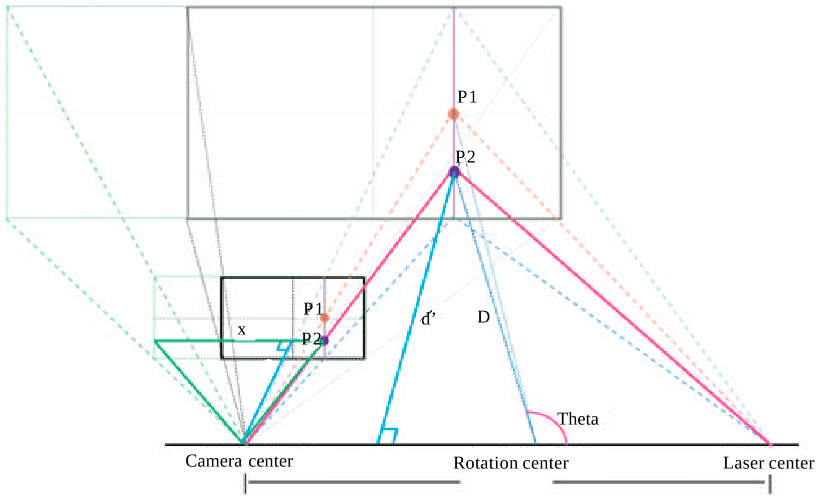

Fig. 2: Figure of the principle of laser range finder

Identify and determine the laser spot and eliminate interference.

Determine the precise location of the laser spot center.

To solve the later problem, the easiest way is to directly find the brightest pixel coordinates in the photoelectricity. However, informed by the preceding equation, pX value thus obtained is an integer, the calculated q will have a relatively large jump. How to make pX become more precise sub-pixel level is introduced here.

For this problem, academia has a lot of research which introduced some kinds of sub-pixel laser spot positioning algorithm as well as the analysis of their strengths and weaknesses. In simple terms, it can be considered that a two-dimensional gauss function can be obtained based on the laser spot on a sample.

Then, the center of the laser spot can be estimated through simple fitting or linear interpolation, a kind of method to find centroid. This production uses the simple centroid method for the center of sub-pixel laser, identifying and calculating the coordinates of the center of laser spot from the white fluorescent lamp picture.

Laser radar uses single point laser scanning, therefore, if you want to measure the 3D data, the use of linear laser supplement is required. The picture captured through the use of a red Straight line laser is shown in Fig. 3.

The solving process of linear laser is similar to that of the point-like laser, the difference of which is to locate the center of the laser spot respectively according to each line (or each column) of the images.

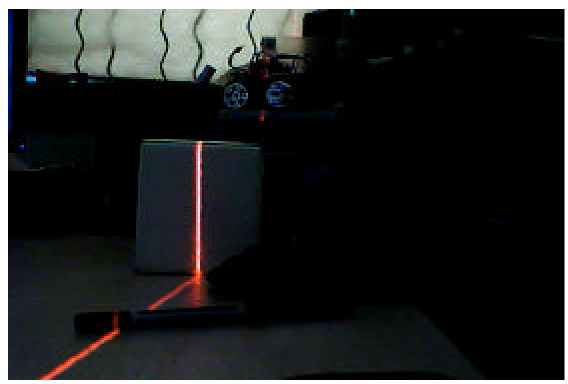

Fig. 3: Picture captured through a red straight line laser

Architectural design of system hardware: The laser used in system is 808nm hyphen infrared with 10mm diameter matched 808 nm low-pass filter. The comparison image of the laser spot after using the optical filter is shown in Fig. 4.

Due to the use of infrared laser, the camera is refitted, that is, by removing the infrared filters in camera lens and fitting with 808 nm infrared low-pass filter, the camera will be able to receive infrared laser emitted from the infrared laser.

The overall structure of the system is shown in Fig. 5.

In the actuator control section, the system uses Arduino actuator library. Arduino Nano is a miniature version of the Arduino USB interface, the biggest difference of which is that there is no power outlet and the USB interface is a Mini-B type socket. Arduino Nano has

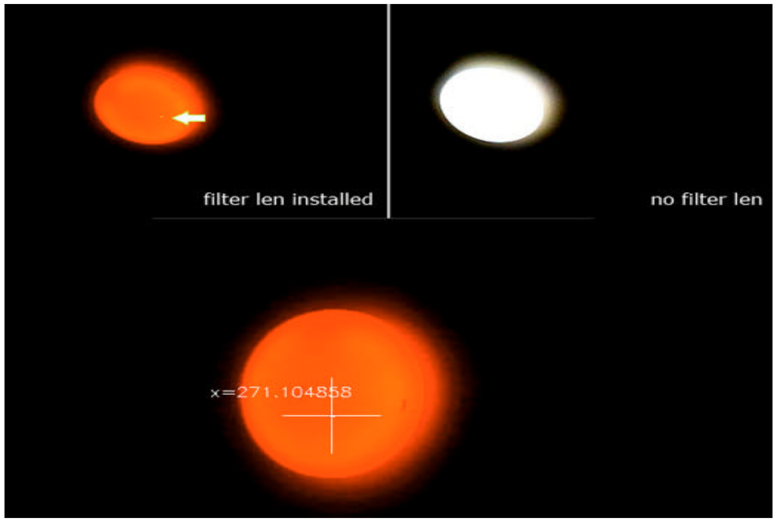

Fig. 4: Comparison image of laser spot after using optical filter

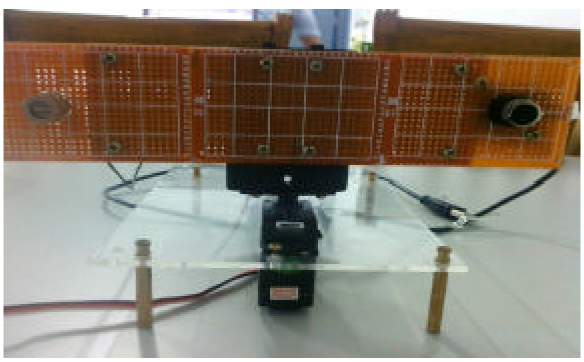

Fig. 5: Overall structure of system

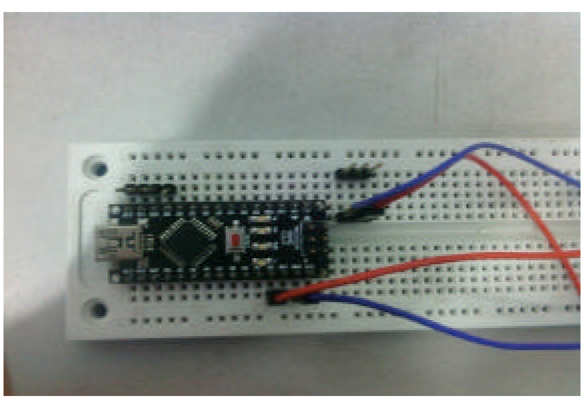

Fig. 6: Arduino control module

a very small size and can be directly inserted in the bread board. The processor core is the ATmega168 (Nano2.x) and ATmega328 (Nano3.0) and it also has 14 digital input/output port (6 of which can be used as PWM outputs), 8 analog inputs, a 16 MHz crystal oscillator, a mini-B USB port, an ICSP header and a reset button. ATmega328 built-in UART can conduct serial communication through the digital port 0 (RX) and 1 (TX) with the outside. Arduino control module is as shown in Fig. 6.

The following short program is the control part of actuator in the Arduino, calling the function of Arduino actuator library:

```
#include <Servo.h>
// create servo object to control a servo
Servo myservo;
// analog pin used to connect the potentiometer
int potpin = 0;
// variable to read the value from the analog pin
int val;
void setup ()
{
// attaches the servo on pin 9 to the servo object
myservo.attach (9);
}
void loop ()
{
// reads the value of the potentiometer (value
between 0 and 1023)
val = analogRead (potpin);
// scale it to use it with the servo (value between 0 and 180)
val = map (val, 0, 1023, 0, 179);
// sets the servo position according to the scaled value
myservo.write (val);
// waits for the servo to get there
delay (15);
}
```

As Arduino has a great flexibility, there is a lot of convenience in the design. Serial communication can only be carried out in control program to control the actuator to rotate to the corresponding angle.

IMPLEMENTATION OF 3D INFORMATION

Serial communication: The system uses the RS232 protocol to communicate, the serial send and receive bytes by bit (bit). Although it's slower than parallel communication by Byte (byte), the serial port can use a line to send data and receive data with the other line simultaneously. It is very simple and can achieve long-distance communications. As the serial communication is asynchronous, the port can send data in one line and receive data in another line. The other lines are used to handshake, but are not necessary. The most important parameters of the serial communication are baud rate, data bits, stop bits and parity check. For the two ports to communicate, these parameters must match.

Image acquisition and calibration processing: This system use Open CV's camera processed function to get the image.

```
// Query frame
cFrame = camera.QueryFrame ();
grayFrame= cvCreateImage (cvSize (cFrame.width, cFrame.height), IPL_DEPTH_8U, 1);
// Convert to grayFrame
cvCvtColor (cFrame, grayFrame, CV_BGR2GRAY);
// Gaussian Blur
(cvSmooth ( grayFrame, grayFrame, CV_GAUSSIAN, 3, 3 );
// Create a video player window
cvNamedWindow ("camera");
// Show the video to the specified location.
ShowImage ( TheImage, IDC_ShowImg );
// Release image resources
cvReleaseImage ( &grayFrame );
// Close camera
(camera.CloseCamera ();
// Destroy the window
(cvDestroyWindow ("camera");
```

In the corrective part of the camera, the design uses the printed Chessboard pattern and shot different images in different distances and locations respectively as shown in Fig. 7.

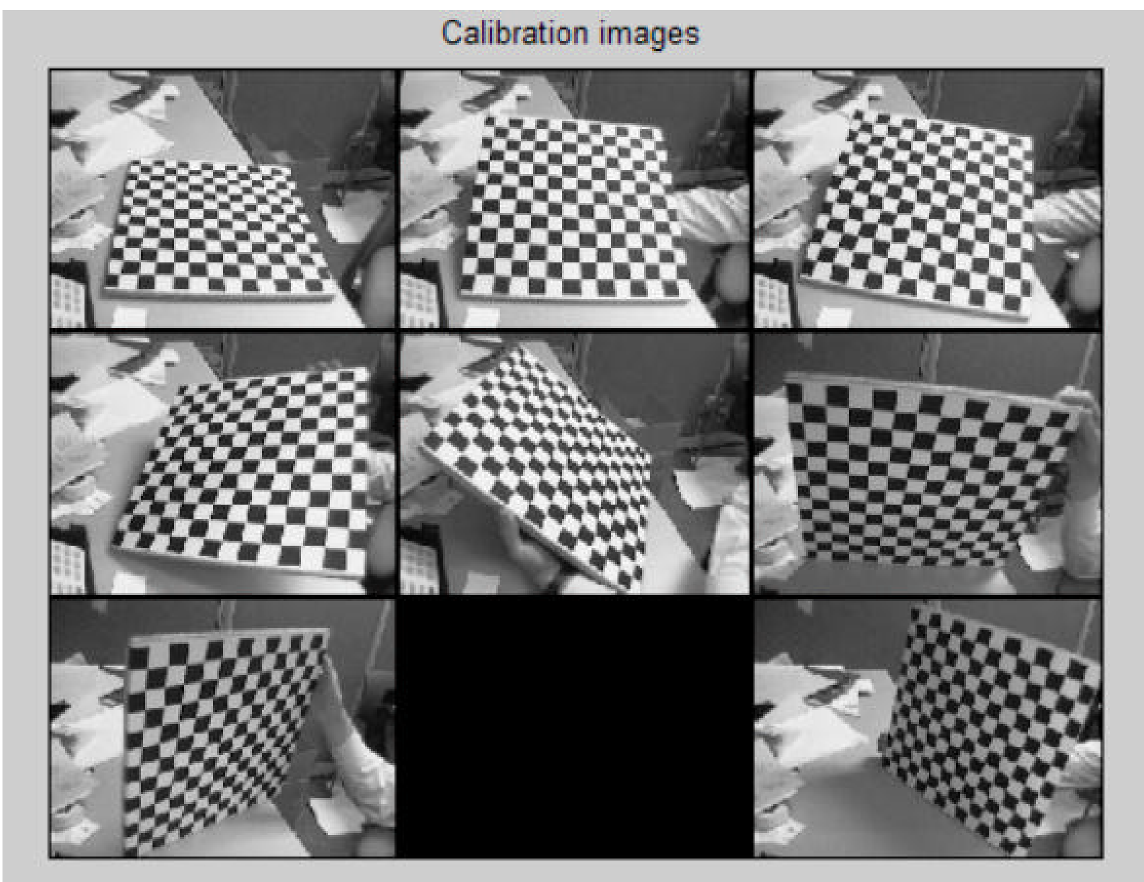

Fig. 7: Board chart filmed in different locations

Figure 8 shows the inner corner identified in calibration which is one of a board chart.

After calibration on above nine board charts, the corresponding calibration parameters will be obtained. OpenCV provides a direct-used calibration algorithm, namely, to input original image and the distortion mapping from the function cvCalibrateCamera2 () to generate the corrected image. Design can either use a one-time function cvUndistort2 () to use the algorithm to complete all matters, or can use a pair of functions including cvInitUndistortMap () and cvRemap () to deal with the matter more efficiently.

The basic approach is to calculate the distortion mapping first and then to correct the image. The function cvInitUndistortMap () is used to calculate the distortion mapping and the function cvRemap () shows applying the mapping in any image [18-19].

The function cvInitUndistortMap () calculates the distortion mapping which associates each point of the image with its innuendo position. The first two variables are the matrix of camera intrinsic parameters and mapping parameters, all of which are from the expected form of the function cvCalibrateCamera2. The generated distortion mapping is indicated by two independent 32-bit single-channel matrixes: The first one gives x as the mapped values of point and the second gives the y value. Results from cvInitUndistortMap () can be passed directly to the function cvRemap ().

```
_intrinsic_cam= (CvMat*) cvLoad (camerainfofile);
_intrinsic_distort= (CvMat*) cvLoad (cameradi
stortfile);
_cam_map_x = cvCreateMat (cy, cx, CV_32F);
_cam_map_y = cvCreateMat (cy, cx, CV_32F);
//Calculate the distortion mapping
cvInitUndistortMap ( _intrinsic_cam, _intrinsic_distort, _cam_map_x,
_cam_map_y);
// Read the calibration parameters and complete the delete of distorted
image
cvRemap (input, output, _cam_map_x, _cam_map_y);
The point cloud data obtained by scanning is stored in a format in
Files:
dnmpfile = fopen ("dnmp_pt.asc", "w");
fprintf (dnmpfile,"%d %d %.2f %.2f %.2f\n"
,currentpt.col,currentpt.row, px, py, pz );
```

3D rendering: The design uses the Irrlicht open 3D engine to simplify the implementation of this part, Irrlicht Engine is a high-performance, cross-platform, open source 3D engine which is characterized by fast running, scalable, thread-safe.

The steps ofIrrlicht Engine rendering required are the following:

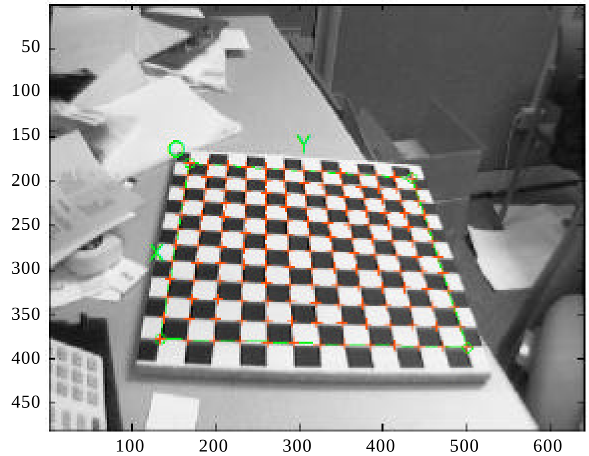

Fig. 8: Inner corner identified in calibration

```
Create a device
Get the pointers of scene manager, GUI environment, video device and use
them to do the render control
Then conduct DrawAll in beginScene and endScene
Release the device.
The most important function of the engine function is "CreateDevice". The
root object of everything Irrlicht engines do is this Irrlicht Device.
// Create Irrlicht Device
device = createDevice ( video::EDT_SOFTWARE, dimension2d<u32>
(640, 480), 16,
        false, false, false, 0);
The following are the equipment that need to be rendered and the
environment:
// Set the window caption
device->setWindowCaption (L"Hello World! - Irrlicht Engine Demo");
// Get the pointers of video device, scene manager and GUI environment and
stored
driver = device->getVideoDriver ();
smgr = device->getSceneManager ();
env = device->getGUIEnvironment ();
mm = smgr->getMeshManipulator ();
// Add the node of camera and lights
ICameraSceneNode* camera =
smgr->addCameraSceneNode (0,core::vector3df (0,-40,0), core::vector3df
(0,0,0) );
ILightSceneNode* lit = smgr->addLightSceneNode (camera,vector3df (0,30,-
20),video::SColorf (0.3,0.3,0.3),10);
We should drop the Irr equipment when finish, because previous video
device, scene manager, GUI environment is only for collection, not for the
"Create", so we only need to delete the Irr equipment.
// Drop the device
Device->drop ();
```

EXPERIMENTAL RESULTS AND ANALYSIS

The laser ranging system used in experiment has collected a two vertical desktop images and realizes the 3D reconstruction by Matlab programming.

Analysis of plane verticality: The inner product of two plane verticality is zero. Equations of the two planes are as follows:

$$0.0025x-0.0007y+0.0017z-1 = 0$$
$$0.0007x+0.0014y-0.0004z-1 = 0$$

When normal vectors of two planes normalized, inner product is:

$$\bullet = 0.003827649$$

Two plane included angles α = arccos () = 1.7785rad, about 89•6 535 degrees.

Absolute error:

$$E1 = 90-89\bullet 6535 = 0.3475$$

Relative error:

$$E2 = 0.3475/90\times 100\% = 0.38\%$$

Analysis of plane dispersion: This study uses the average value of standard deviation of discrete point and the plane distance to measure the dispersion of the plane. Through the experiment, the standard deviations of two plane discrete points are σ1 = 3.0965 mm and σ2 = 4.3966 mm; distance average of d1 = 2.3515 mm and d2 = 3.1453 mm.

Analysis of system measurement error: Set measuring distance as D and use 4σ/D values to measure the measurement error of the system. For this experiment, measuring distance is about 500 mm, the standard deviations of discrete points are 3.0965 mm and 4.3966 mm separately and the system measurement errors are:

$$\varepsilon 1 = 4\sigma/D = 4\times 3.0965500\times 100\% = 2.48\% \quad (6)$$

$$\varepsilon 2 = 4\sigma/D = 4\times 4.3966500\times 100\% = 3.52\% \quad (7)$$

CONCLUSION

This study studies a 3D laser radar information collection system and the control of actuator is realized by serial telecommunication, the 2D images is captured from lines to surface, then denoise processing calibration is carried out by using matlab and Open CV and by using Irrlicht3D engine, the point cloud data is to be rendered to convert the 2D images to the 3D effect. Robot's collection on external image is achieved through the study of Open CV learning that combined with VC2008. This study reduces the accuracy to realize low-cost production to meet the planed requirements and provide a steady run, of course, there is still much room for improvement of this system, the accuracy can be improved on this basis.

ACKNOWLEDGMENT

The authors are highly thankful for the financial support of Science Fund Project from Nanjing Institute of Technology through Grant Nos CKJ2011013 and QKJB2011025.

REFERENCES

Bi, D.X., F.T. Liu and Q. Xue, 2009. Field calibration method of structured light vision sensor based on laser cross line. J. Sci. Instrument, 30: 1697-1701.

Chen, S.Y. and S. Liu, 2008. Computer Vision Technology Based on Open CV. Science Press, China.

Gao, F.G. and S.R. Xie, 2005. Robot Censoring Technology and its Application. Chemical Industry Press, China.

Xiao, N.F., 2008. Intelligent Robots. South China University of Technology Press, China.

Xu, D. and W. Zou, 2008. The Perception, Positioning Control of Indoor Mobile Robot. Science and Technology Press, China.

Zhang, G.J., 2005. Machine Vision. Science and Technology Press, China.

Zhi, W., Q.S. Luo and X.D. Su, 2012. A novel mobile robot path planning method based on ACO algorithm using new ants meeting judgment strategy. Int. J. Digital Content Technol. Appl., 6: 302-309.