Real-time Calibration and Registration Method for Indoor Scene with Joint Depth and Color Camera

Fengquan Zhang[1,2*], Tingshen Lei[1], Jinhong Li[1], Xingquan Cai[1], Xuqiang Shao[2,3], Jian Chang[4], Feng Tian[2,5]

1. Beijing Key Laboratory on Integration and Analysis of Large-scale Stream Data, North China University of Technology, Beijing, China
2.State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China
3. School of Control and Computer Engineering, North China Electric Power University, Baoding, China
4. National Centre for Computer Animation, Bournemouth University, Poole, UK
5. School of Computer and Information Technology, Northeast Petroleum University, Daqing, China
Email：fqzhang@ncut.edu.cn

Abstract:

Traditional vision registration technologies require the design of precise markers or rich texture information captured from the video scenes, and the vision-based methods have high computational complexity while the hardware-based registration technologies lack accuracy. Therefore, in this paper, we propose a novel registration method that takes advantages of RGB-D camera to obtain the depth information in real-time, and a binocular system using the Time of Flight (ToF) camera and a commercial color camera is constructed to realize the three-dimensional registration technique. First, we calibrate the binocular system to get their position relationships. The systematic errors are fitted and corrected by with the method of B-spline curve. In order to reduce the anomaly and random noise, an elimination algorithm and an improved bilateral filtering algorithm are proposed to optimize the depth map. For the real-time requirement of the system, it is further accelerated by parallel computing with CUDA. Then the Camshift-Based Tracking Algorithm is applied to capture the real object registered in the video stream. In addition, the position and orientation of the object is tracked according to the correspondence between the color image and the 3D data. Finally, some experiments are implemented and compared using our binocular system. Experimental results are showed to demonstrate the feasibility and effectiveness of our method.

Keywords: binocular stereoscopic vision; image tracking; depth map; filtering, registration; calibration;

1. Introduction

Camera calibration is a procedure to compute the relationship between the coordinate in real-world and the image coordinate of a camera. It is an important step in computer vision fields and its precision largely influences the quality of different computer vision applications such as three-dimensional modeling [1,2], depth estimation [3,4], and augment reality [5,6]. The technique of 3D registration is also widely used in scene modeling [7,8], image tracking [9,10] and vision registration fields [11,12]. Recently, the RGB-D camera is a relatively new type of depth sensor, which can capture 3D data with a good frame rate, and it has a high focus in many fields of computer vision [13,14]. Time of flight (ToF) and structure light (SL) are two styles of RGB-D sensors. However, there are still many limitations such as the low resolution of the captured depth image, and there are too many abnormal points and noises in the captured depth data.

In this paper, we construct a binocular system using a ToF camera and a commercial color camera to complete the joint calibration and registration technology for the indoor scene application of augment reality, as shown in Figure 1. Because of the real-time capture of the 3D scenes information, we can simplify the realization of occlusions between real objects and virtual objects to realize the human computer interactions. A new calibration model is presented for all cameras parameters. Then we calibrate the ToF camera to remove the abnormal points in the depth map and smooth the noises. An elimination algorithm and an improved bilateral filtering algorithm is proposed to optimize the depth map. In 3D registration, we need to convert the depth map to the view of the color camera. After the conversion, the 3D point cloud data is obtained with color information from the view of the color camera. In order to demonstrate the stableness and

preciseness of joint calibration method, some applications are implemented in our binocular system.

Figure1. Joint acquisition system. SR-4000(left), DH-51(middle) and our binocular system (right).

The methodology for jointly calibrating and registration with color and depth camera is shown in the Figure 2. It begins from stereo data captured by the color and depth camera. In the calibration stage, corner detection for a usual 9×7 checkerboard is enforced to get the image points for captured image. The Harris corner detection algorithm is developed on the basis of the Moravec algorithm, and the systematic errors of ToF are fitted and corrected by the method of B-spline curve. Then a positional relationship of the two cameras is established. In the registration stage, a bilateral filtering algorithm is designed to optimize the depth map and reduce random noise, and the Camshift-based tracking algorithm is applied to capture the real object registered in the video stream. Then the pose of the camera is tracked according to the color image and the depth data, and virtual object can be well registered in the right position using our binocular system.

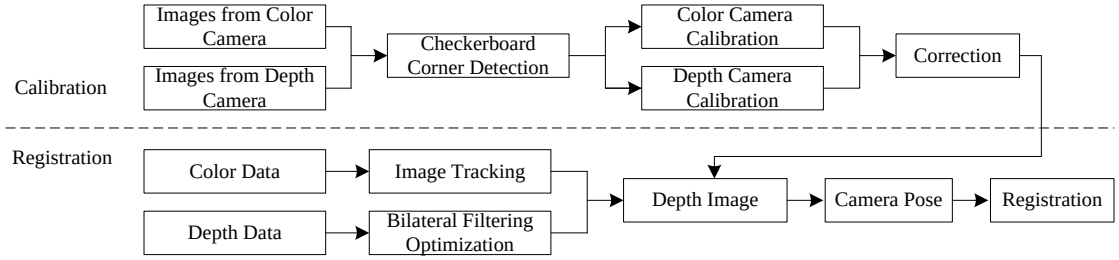


Figure 2. Process flow of calibration and registration with joint depth and color camera

2. Related work

Binocular stereoscopic vision was presented by Ishiguro in 1992 [15]. For more than twenty years, it has become a mature method with the improvement of feature registration algorithms and the development of camera calibration. The traditional registration technology can be divided into two categories, dynamic and static registration method. For the dynamic registration technology, there were tracking-based registration and vision-based registration technology. The main purpose of the former method was to record the position of the camera in the world coordinates and the direction of the optical axis to ensure the continuity of both virtual model and the real world to achieve accurate 3D registration [16,17]. The latter mainly calculated the position and orientation of the camera in the world coordinates by a given image or multiple images [18]. There are two widely applied methods of vision registration: one is to firstly calibrate the camera, analysis the reasonable positions from the captured video images, and then calculates the relative position of the camera. The second is to use affine transformation of the feature points for 3D registration. Camera calibration is to calculate the camera's intrinsic parameters, and then use the camera parameters and the images to calculate the direction and position of the camera, which is a transformation from a 2D geometric imaging plane to the 3D real scene in fact [19,20]. In order to get rid of the markers, the pose estimation method based on the natural texture features in the target scene was widely studied [21-23]. Such methods need no markers in the scene. But they

need to track the natural feature points in the image, and then deduce the movement of the camera to estimate the pose.

There are two main acquisition methods of the depth map, active depth acquisition method and passive depth acquisition technology. The active technology acquires depth information by emitting energy beams (laser, electromagnetic waves, and ultrasonic) to the target and detect echoes. The passive technology does not need to manually set the emitting source, but use the 2D images of the scenes in the natural light to reconstruct 3D information. With the development of the optical imaging technology, a variety of high-precision optical measuring instruments have been successfully developed, which provides accurate access to the depth information. The most commonly used active depth acquisition methods were time of flight (ToF) method [24,25] and the structural light (SL) method [26]. Some off-the-shelf devices are based on the above two concepts. Kinect v1, ASUS Xtion Pro Live and Structure Sensor were based on the SL idea, but Kinect v2 was based on the ToF concept [27-30]. The calibration of depth camera has been widely applied since the release of the Kinect v1. Different calibration methods based on depth camera have been researched by various organizations[31-33]. Paper [34] proposed a calibration method for the depth sensor, which used the disparity data from IR camera and RGB image to calibrate the external and internal parameters. An empirical model was designed to decrease the distortions of IR sensor [35], which was useful for some depth camera. But it had some limitations in automation and accuracy. Zhang [36] presented a classic method of calibration which used the maximum likelihood estimation to obtain internal parameters. However, its drawback is distortion of parameters for projectors and cameras, which are not compensated or estimated.

There are two problems in current depth camera. One is the implement of distortion along with the camera during the calibration procedure. Another is the adjust of systematic errors producing from the inaccuracy of in-factory calibration. This paper addressed these problems using a two-step calibration step to get all of the geometric parameters of depth camera. The experimental design are discussed with different test models and method comparisons in the end.

3. Calibration for the system of binocular stereoscopic vision

3.1 The principle of ToF camera

ToF camera is a device that provides an active light source. It emits modulated infrared lights to the surrounding environment. There is a specific sensor inside of the camera to collect the reflected back infrared lights. The emitted infrared signal will be reflected by the objects in the environment, and the CCD chip inside the camera will collect the reflected signals. After sampling on each pixel, mixing the reflected infrared signal and the modulated internal reference signal, a correlation function can be obtained as shown in Equation 1. The phase difference of the infrared signal can be calculated by the correlation function, and consequently we can obtain the distance between the ToF camera and the target object.

$$c(\tau) = s \otimes g = \lim_{T\to\infty} \int_{-T/2}^{T/2} s(t) \cdot g(t+\tau) dt \tag{1}$$

where $g(t)$ is the modulated reference signal, $s(t)$ is the infrared signal that is reflected and captured by the camera. τ is the phase difference of the periodic variation of the reference signal inside the ToF camera. The waveform of the infrared signal emitted from the ToF camera is usualy a signal, so $g(t)$ and $s(t)$ can be defined as:

$$g(t) = \cos(wt) \tag{2}$$

$$s(t) = b + a\cos(wt + \phi) \tag{3}$$

where ω is the modulation frequency, a is the amplitude of the incident signal, b is the correlated offset of the signal, and ϕ is the phase difference associated with the distance from the ToF camera to the target object. The correlation function obtained in the case of a sinusoidal signal is described in Equation 4:

$$c(\tau) = \frac{a}{2}\cos(\omega\tau + \phi) + b \tag{4}$$

Four consecutive images with their phase difference τ as $A_i = c(i \cdot \frac{\pi}{2}), i = 0,1,2,3$ can

determine the correlation function $c(\tau)$ respectively. The demodulation of the correlation function is accomplished by sampling the correlation function $c(\tau)$. Its parameters can be obtained by the following equations:

$$\phi = \arctan\left(\frac{A_3 - A_1}{A_0 - A_2}\right) \tag{5}$$

$$I = \frac{A_0 + A_1 + A_2 + A_3}{4} \tag{6}$$

$$a = \frac{\sqrt{(A_3 - A_1)^2 + (A_0 - A_2)^2}}{2} \tag{7}$$

Where I is the intensity of the incident infrared light. We can obtain the depth of the object by $d = \frac{c}{4\pi\omega}\phi$, where ϕ is known and c is the speed of light $c \approx 3 \times 10^8\ m/s$.

After the depth value is obtained, with the intrinsic parameters of the ToF camera, we can obtain the target object's 3D coordinates based on Equation 8-10.

$$z = r \cdot \frac{f}{\sqrt{f^2 + (X_c d_x)^2 + (Y_c d_y)^2}} \tag{8}$$

$$x = z \cdot \frac{X_c d_x}{f} \tag{9}$$

$$y = z \cdot \frac{Y_c d_y}{f} \tag{10}$$

where f is the focal length of the camera, d_x and d_y are the actual length of a pixel in the direction of x and y, respectively. X_c , Y_c are the normalized coordinates of the pixels relative to the optical center.

However, as a new device, the ToF camera has many aspects that need to be improved, such as low resolution of the depth maps, system error, depth data inaccuracy caused by spatial depth discontinuity, motion blur and so on.

3.2 **Calibration for the binocular system**

3.2.1 Calibration for commercial cameras

We use the traditional method to calibrate the commercial color cameras, which requires scorner detection to obtain the corresponding relationship between image pixels and points on the calibration board before calibration. The Harris corner detection algorithm is developed on the basis of the Moravec algorithm [37]. It is a point feature extraction operator based on the signal autocorrelation function.

This method uses a local detection window, and examines the average energy change in the window when it slightly moves in all directions. When the energy change exceeds a predefined threshold, it is considered that the central pixel of the window is a corner. The change of the grayscale in the window is calculated by the differential operator, which makes the corner detection rotational invariant. Suppose the grayscale value of a pixel (*x, y*) is *f (x, y)*, the grayscale change in the window can be described as Equation 11:

$$E_{u,v}(x, y) = \sum_{u,v} W_{u,v}[f(x+u, y+v) - f(x, y)]^2 \tag{11}$$

$$= \sum_{u,v} W_{u,v}[x\frac{\partial f}{\partial x} + y\frac{\partial f}{\partial y} + \Delta(x^2 + y^2)]^2$$

where *u, v*, and *W* are the coefficient of the Gaussian window at position (*u, v*). In order to improve the anti-noise ability, we use the following Gauss window [38] to apply Gaussian smoothing to the image:

$$W_{u,v} = \exp[-\frac{1}{2}(u^2 + v^2) / \delta^2] \tag{12}$$

where *W* is the function of Gaussian window.

3.2.2 ToF Camera Calibration

The ToF camera used in this paper is the SR4000 with the resolution of 176×144, so we use the traditional black and white checkerboard to calibrate the camera. This method needs to collect grayscale images of the checker board. Although the ToF camera is not able to obtain grayscale images, it can generate intensity maps that record the intensity of the reflected infrared light. Since the checkerboard has only black and white colors, the black area absorbs most of the light, and the white area reflects most light back. So flection intensity map is equivalent to a grayscale image for checkboards.

Figure 3 left is an intensity map obtained by the ToF camera. In the background where no infrared light is reflected is black. The closer an object is to the camera, the brighter it is. And the intensity map of the checkerboard is the same as the grayscale image generated by a normal camera.

The intensity map obtained by a ToF camera has low resolution, and there are noises. So it is difficult to accurately detect corners when calibrating them with a calibration board. The images obtained from a ToF camera is much different from images obtained from pinhole imaging systems. Both the grayscale images and the intensity maps have a certain degree of distortion. Therefore, we need to preprocess the intensity map before calibration. The intensity map is upsampled to improve its resolution from 176×144 to 528×432 as shown in Figure 3 right.

To model the system error of the ToF cameras, one method is to assume that the system error is linear [39], which is proved wrong in many studies. The distribution of the system error is periodic, similar to the sine function. In this paper, we use cubic B-spline curve [40] to model and correct the system error. It can well fit the system error of the ToF cameras. The B-spline recursion is defined as in Equation 13:

$$P(u) = \sum_{k=0}^{n} p_k B_{k,d}(u), \quad u_{\min} \le u \le u_{\max}, \quad 2 \le d \le n+1 \tag{13}$$

where p_k is a group of input with $n + 1$ control points. d is the order parameter, and the blending function $B_{k,d}$ is a polynomial with order of d-1.

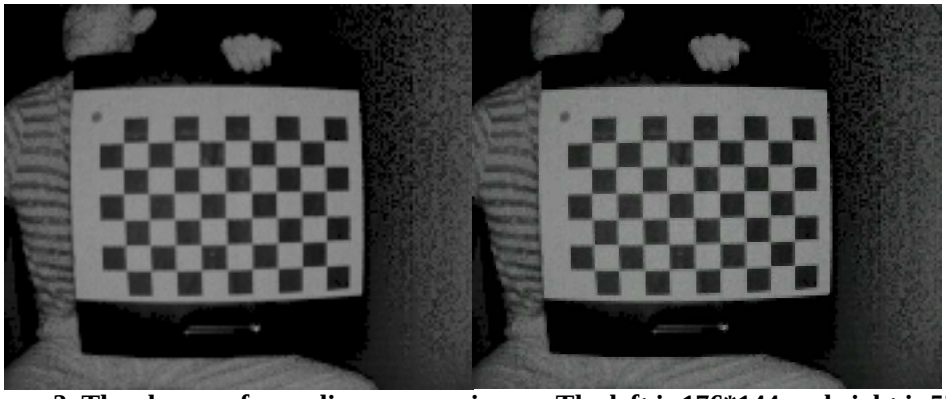


Figure 3. The change of sampling on gray image, The left is 176*144 and right is 528*432.

$$B_{k,1}(u) = \begin{cases} 1 & , u_k \le u \le u_{k+\_1} \\ 0 & , \text{ otherwise} \end{cases} \tag{14}$$

$$B_{k,d}(u) = \frac{u - u_k}{u_{k+d-1} - u_k} B_{k,d-1}(u) + \frac{u_{k+d} - u}{u_{k+d} - u_{k+1}} B_{d+1,d-1}(u) \tag{15}$$

where $\{u_0, u_1, \ldots, u_{n+k}\}$ is a non-decreasing sequence of nodes, and k is the order of the curve. We compute the blending function with its uniform way.

We take pictures of the checkerboard placed in front of the camera every 5cm starting from the position 0.6m away from the camera. And then we calculate the distance between the

checkerboard and the color camera, which is converted to the view of the ToF camera according to the positional relationship between the two cameras in the binocular system. While taking photos of the checkerboard, the ToF camera also measures the distance between the checkerboard and the ToF camera. Therefore, we can figure out the system error curve using the cubic B-spline interpolation.

3.2.3 Calibration between the ToF camera and the 2D color camera

The calibration between the ToF camera and the normal color camera is the key issue for all algorithms that is based on the depth acquisition using the ToF cameras. Because the ToF cameras can cause distortions as we mentioned earlier, we interpolate the intensity map to improve the accuracy of corner detection to correct the distortion.

In this paper, we use the checkerboard to calibrate based on the binocular stereoscopic method, and get the extrinsic parameters of the two cameras, R_{tof} , T_{tof} and R_c , T_c , using the Matlab Calibration Toolbox. The parameters represent the orientation and position of the two cameras in the world coordinate system, i.e., the coordinate system defined by the checkerboard, respectively. $[w]$ represents the world coordinate system defined on the checkerboard, $[tof]$ represents the visual coordinate system on the ToF camera, and $[c]$ represents the visual coordinate system on the normal camera. So we have

$$[tof] = [w]M_{tof}, \quad [c] = [w]M_c \tag{16}$$

where

$$M_{tof} = \begin{pmatrix} R_{tof} & -R_{tof} \cdot T_{tof} \\ 0 & 1 \end{pmatrix}, M_c = \begin{pmatrix} R_c & -R_c \cdot T_c \\ 0 & 1 \end{pmatrix} \tag{17}$$

Thus, the normal camera can be expressed in the ToF camera coordinate system as:

$$[c] = [tof]\begin{pmatrix} R_{tof} & -R_{tof} \cdot T_{tof} \\ 0 & 1 \end{pmatrix}^{-1}\begin{pmatrix} R_c & -R_c \cdot T_c \\ 0 & 1 \end{pmatrix} \tag{18}$$

According to Equation18 we can determine the positional relationship of the two cameras.

4. ToF depth map processing

4.1 Detection of abnormal points

Because the resolution of the ToF camera is only 176×144, and there are lots of noises in the obtained depth data, it is impractical to use the raw data obtained from the ToF camera directly. The low resolution will lead to inaccuracy of the captured depth information at positions with sudden depth changes in the scene. Abnormal points are irrelevant information which may degrade the performance of the subsequent filtering algorithm, so they must be removed. In this paper, we take advantages of the two cameras that both color and depth information can be acquired at the same time to remove the abnormal points.

Firstly, we utilize the intensity map generated by the reflected infrared lights to remove the abnormal points. The intensity of each pixel can reflect the reliability of the depth information. Too strong intensity indicates an excessively saturated pixel, which should be removed. Too weak intensity indicates that the received signal is not strong enough to calculate reliable depth information. Therefore, we only preserve the pixels with intensity in the range of [20%，80%].

Secondly, we evaluate the reliability of every pixel. The change of color is usually synchronised with the change of depth [41]. So if a huge difference of depth value is detected in the area of similar color, it indicates the probability of anomalies of the depth value in this area. In contrast, we allow big change of depth value where color is also of large difference. According to this feature we use Equation 19 to evaluate the reliability of each pixel. Several times of iteration of Equation 19 can determine how a reliable pixel is.

$$s_i = \sum_{j \in N_i} s_j \exp\left(-\frac{1}{\sigma_d} w_{ij}(d_i - d_j)^2\right) \tag{19}$$

where s represents the reliability of a pixel. It is a weighted value of the reliabilities of the *i*-neighborhood pixels. d is the depth value of the pixel. w_{ij} is defined as in Equation 20:

$$w_{ij} = \exp\left(-\frac{1}{\sigma_c}\left\|c_i - c_j\right\|_2^2\right) \tag{20}$$

where c can be obtained by $c = \frac{1}{3}(R+G+B)$, which indicates the intensity of the color calculated from the color image. When the calculated value stabilizes after several times of iteration, the depth value is lower than a specified threshold that should be removed.

4.2 Optimization of the depth maps using an improved Bilateral Filtering Algorithm

After the abnormal points removed, some noises that can degrade the accuracy of the subsequent 3D registration may still exist. Thus, the depth map needs to be denoised. We take advantage of the feature that continuous depth corresponds to similar color to extend the bilateral filter algorithm, which filters the depth data using color information. The extended bilateral filtering algorithm is shown in Equation 21 and 22:

$$d_p = \frac{1}{w_p}\sum_{q\in N_p} G_{\sigma_s}(\|p-q\|)G_{\sigma_d}(|d_p - d_q|)G_{\sigma_I}(|I_p - I_q|)d_q \tag{21}$$

$$w_p = \sum_{q\in N_p} G_{\sigma_s}(\|p-q\|)G_{\sigma_d}(|d_p - d_q|)G_{\sigma_I}(|I_p - I_q|) \tag{22}$$

where w_p is the normalization factor. Because the ToF cameras cannot acquire color information of the scenes, we use the corresponding color information provided by the 3D data in the weighted calculation. σ_s determines the size of the neighborhood in the filtering, σ_I determines the range of valid color values, and σ_d determines the range of valid depth values. The improved bilateral filtering algorithm can well preserve the boundaries of 3D data, and effectively remove the internal random noises. The details of the algorithms are as follows:

Pseudo code of the Bilateral Filtering Algorithm 1

for each pixel *p* in N_p

（1） Initialization: d_p=0, $w_p = 0$

（2） For each pixel *q* in N_p

(a) $w = G_{\sigma_s}(\|p-q\|)G_{\sigma_d}(|d_p - d_q|)G_{\sigma_I}(|I_p - I_q|)$

(b) $d_p += wd_q$

(c) $w_p += w$

（3） Normalization: $d_p = d_p / w_p$

end for

4.3 Bilateral filtering algorithm acceleration

The improved bilateral filtering algorithm calculates a weighted average of the depth values of the current pixel based on the depth and color values of its neighborhood pixels. This method can well preserve the boundaries of 3D data, while removing noises insides the object. The resolution of the ToF camera is 176×144. Because the depth value of every pixel is related to the depth values of its neighborhood, we need to store the captured depth and color information from each frame in the global memory, so that the neighboring depth and color information of the current pixel can be accessed by every computing thread to enable parallel computing. We divide the depth map to 11×9 blocks, each contains 16×16 threads. Each thread computes the depth

value of one pixel. Thus, the parallel computing can accelerate the computation of the depth data for each frame.

Because the data are stored in the global memory, the index of each pixel is global. However, the indices of the blocks and threads are local. Therefore, we use Equation 23 and 24 to get the global index of a pixel using the local indices of the blocks and the threads.

$$row=blockIdx.y*blockDim.y+threadIdx.y \tag{23}$$

$$col=blockIdx.x*blockDim.x+threadIdx.x \tag{24}$$

where row and col represent the abscissa and ordinate of the to be processed pixel, respectively.

4.4 Generation of the depth maps in the view of the color camera

We obtain the color information and the depth information with two different cameras respectively. However, the two cameras have different views, so we need to convert the captured depth data into the view from the color camera if we want to use them in augmented reality.

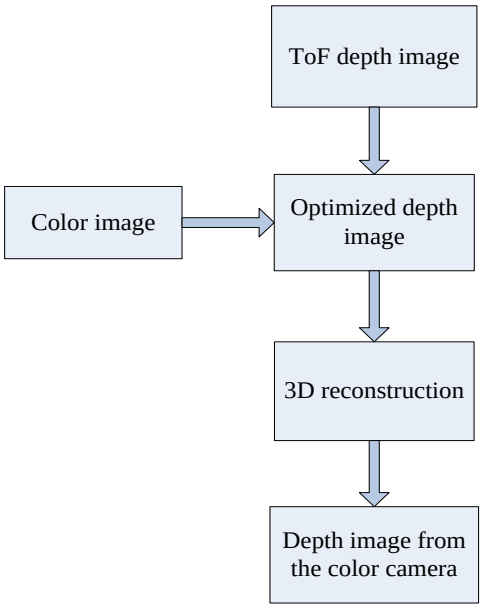


Figure 4. Generation process of the depth map in the view of the color camera

As shown in Figure 4, we assume each pixel of the optimized depth map captured by the ToF camera is corresponding to a point in the physical space. According to the previously calibrated intrinsic parameters of the ToF camera, we can obtain the 3D coordinate of each pixel in the depth map. If we establish a world frame on the ToF camera, as the position of the color camera in the world frame is already obtained from the binocular calibration, the calibration of the intrinsic parameters of the color camera is also completed. We project the reconstructed 3D data to the 2D image plane of the color camera, so that the relationship between the color image and the 3D space is established. Because of the difference between the views of the two cameras, only partial 3D data are obtained in the color image.

5. 3D registration based on depth maps

5.1 Determining the areas to be registered using video tracking technology

This paper uses the video tracking technology to dynamically track the area superimposed by the virtual object in the image [42]. The tracking algorithm consists of a pre-processing stage and a real-time stage. In the pre-processing stage, to accurately recognize and track colors, the RGB video image needs to be converted to a more stable color space that is adaptive to the subsequent histogram calculation. In the real-time stage, the color histogram and probability distribution of the tracked color are needs to be solved to realize tracking in each frame of images. The tracking algorithm in this paper is based on the Camshift algorithm.

This algorithm firstly computes a histogram of hues of all colors in the window by sampling the hue channel (H). And then calculate the probability of a pixel being used as the target pixel by searching the histogram model of each pixel in the video for the matching histogram. Video images can be converted to a probability distribution map of colors using this method, namely, histogram back projection. For ease of display, the color probability map is converted to an 8-bit grayscale projection. The pixel value with probability of 1 is set to 255, the pixel value with probability of 0 is set to 0, and the other pixels are converted to their corresponding gray values. So the brighter pixels in the gray scale projection indicates that the pixel has higher possibility to be the target pixel.

The idea of the Camshift window search algorithm is as follows:

(1) Select a search window *W* of size *S* from the color probability distribution map;

(2) Define (*x,y*) as a pixel in the search window, *I(x,y)* as the pixel value at the position of (*x,y*) in the projection.The zero-order matrix M_{00} and the first order matrix M_{01}, M_{10} of the search window are shown in Equation 25:

$$M_{00}=\sum_x\sum_y I(x,y),\ \ M_{01}=\sum_x\sum_y xI(x,y),\ \ M_{10}=\sum_x\sum_y yI(x,y) \tag{25}$$

(3) Based on Equation 25, we can obtain the centroid position $(x_c, y_c)$of the search window as:

$$x_c=\frac{M_{01}}{M_{00}}, y_c=\frac{M_{10}}{M_{00}} \tag{26}$$

(4) Reset the size of the search window S to the function of the color probability distribution in the search window above.

(5) Iterate steps (2)(3) and (4) until the convergence condition is met, i.e., the change of the centroid position is less than a predefined threshold, or the maximum of iteration times has reached. And then continue to the next frame.

At the end of the calculation of the current frame, the position and size of the next search window of the next frame are set by the centroid position and the zero order matrixM_{00} obtained from the current frame. And then we can calculate the second order matrix of the search window Z_{20}、Z_{02} and Z_{11}.

$$Z_{20}=\sum_x\sum_y x^2I(x,y),\ \ Z_{02}=\sum_x\sum_y y^2I(x,y),\ \ Z_{11}=\sum_x\sum_y xyI(x,y) \tag{27}$$

The direction angle of the long axis of the target that is searched for in this frame is:

$$\theta=\frac{1}{2}\arctan\left[\frac{2(Z_{11}/Z_{00}-x_cy_c)}{(Z_{20}/Z_{00}-x^2{}_c)-(Z_{02}/Z_{00}-y^2{}_c)}\right] \tag{28}$$

assume that:

$$a=Z_{20}/Z_{00}-x_c^2,\ \ b=Z_{11}/Z_{00}-x_cy_c,\ \ c=Z_{02}/Z_{00}-y_c^2 \tag{29}$$

so the length of long axis and the short axis of the target can be calculated by Equation 29 respectively.

$$I=\sqrt{\frac{(a+c)+\sqrt{b^2+(a-c)^2}}{2}},\ \ W=\sqrt{\frac{(a+c)-\sqrt{b^2+(a-c)^2}}{2}} \tag{30}$$

Repeating the aforementioned processes will realize the continuous calculation of the search window and the continuous tracking of the search target.

5.2 Calculate the camera's position and orientation based on the depth map

After the tracking area in a color image determined, this area can always be tracked and identified, regardless of how the camera is moving. Its color information and 3D data can be obtained from the images we acquired in the previous section. The 3D data is based on the coordinate system with the ToF camera as the origin, whose position and orientation are fixed relative to the coordinate system. However, the camera’s position and orientation determined by the 3D registration technology are relative to the markers. Therefore, we need to figure out the position and orientation of the tracking area relative to the camera, i.e., the position and orientation of the tracking area relative to the world coordinates defined on the ToF camera.

The relative position of the camera and the tracking area can be obtained directly from the

depth map. Considering that the resolution of the ToF camera is only 176×144, but the resolution of the color image is 800×600, so after the 3D data is projected into the image space of the color camera, some of the areas have no corresponding 3D data in the image space. In order to find the locations more accurately, if the 3D data of a certain area is too little, this area will be discard; otherwise if the 3D data of a certain area is dense enough, it will be retained. In this paper, the resolution of the color image is 800×600, where the tracking areas are divided into small squares with size of 5×5. We count the valid points inside the square, i.e., the number of pixels with 3D data, and calculate the position of the square in the world coordinate system. If the total number of valid points inside a square is greater than a threshold, the square is considered to be valid. Finally, the average position of all valid squares of a tracking area is obtained, which is the 3D position of the tracking area in the scene.

After the 3D position of the tracking area is determined, we need to find out the orientation of the camera relative to the markers. Assume that the tracking area in the 3D scene is a plane, or we select a tracking area as far as possible to be a plane. Therefore, we can fit the point cloud data of the tracking area into a plane in the 3D space using the least squares method. The normal vector of the plane in the camera's coordinate system is considered as the orientation of the camera relative to the tracking object.

Specifically, we denote the plane to be fitted as $Ax+By+Cz+1=0$. Suppose there are n valid points in a tracking area, the fitting equation can be expressed in the following matrices.

$$\text{i.e., } \begin{bmatrix} x_1 & y_1 & z_1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & z_n \end{bmatrix}\begin{bmatrix} A \\ B \\ C \end{bmatrix}=\begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix}$$

Pre-Multiply both sides by

$$\begin{bmatrix} x_1 & y_1 & z_1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & z_n \end{bmatrix}^T, \begin{bmatrix} x_1 & y_1 & z_1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & z_n \end{bmatrix}^T\begin{bmatrix} x_1 & y_1 & z_1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & z_n \end{bmatrix}\begin{bmatrix} A \\ B \\ C \end{bmatrix}=\begin{bmatrix} x_1 & y_1 & z_1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & z_n \end{bmatrix}^T\begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix}$$

Simplify and get

$$\begin{bmatrix} x_1 & \cdots & x_n \\ y_1 & \cdots & y_n \\ z_1 & \cdots & z_n \end{bmatrix}\begin{bmatrix} x_1 & y_1 & z_1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & z_n \end{bmatrix}\begin{bmatrix} A \\ B \\ C \end{bmatrix}=\begin{bmatrix} x_1 & \cdots & x_n \\ y_1 & \cdots & y_n \\ z_1 & \cdots & z_n \end{bmatrix}\begin{bmatrix} -1 \\ -1 \\ -1 \end{bmatrix}$$

$$\begin{bmatrix} A \\ B \\ C \end{bmatrix}=\begin{bmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i z_i \\ \sum x_i y_i & \sum y_i^2 & \sum y_i z_i \\ \sum x_i z_i & \sum y_i z_i & \sum z^2{}_i \end{bmatrix}^{-1}\begin{bmatrix} -\sum x_i \\ -\sum y_i \\ -\sum z_i \end{bmatrix} \quad (31)$$

By solving the coefficients A, B, and C, the normal vector of the fitting plane （A，B，C） can be obtained. In the coordinate system of the ToF camera, the fitted value of C can be greater than zero, less than or equal to zero. We chose the normal vector that is always facing the camera, so that the value of C is less or equal to zero. If the value of C is greater than zero, the normal vector will be (-A，-B，-C). We define the initial normal vector of the plane is (0, 0, -1), so the rotation axis of the plane is $\vec{n}=(A,B,C)\times(0,0,-1)$, and the rotation angle is $\theta=\arccos\frac{(A,B,C)\cdot(0,0,-1)}{\sqrt{A^2+B^2+C^2}}$. So far we have figured out where the virtual object should be placed and its orientation. The 3D registration technology has been completed.

6. Results and discussion

The experiment of this paper is built with Window 7 operating system, Intel(R) Core(TM) 4*E7300, 4G memory, GTX480, a ToF camera Mesa Imaging SR4000, and a commercial color

camera Daheng HV51CMOS. A 9×6 black and white checkerboard of 2cm side length was applied for calibration.

6.1 Calibration experiment for the TOF camera

We took intensity images of the checkerboard with different positions and orientations, some chosen images are shown in Figure 5. After upsampling the intensity maps, we could get the calibration result of the intrinsic parameters of the ToF camera using the Matlab Calibration Toolbox. Table 1 shows three sets of results of the calibration.

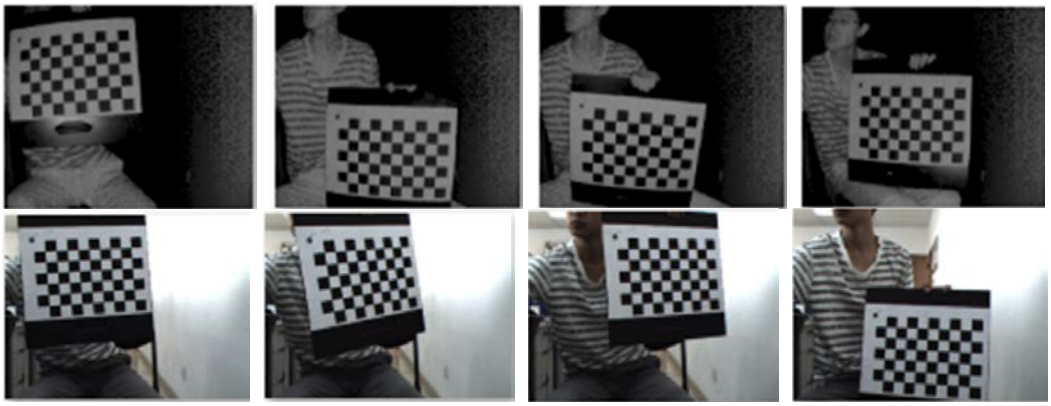


Figure 5. Sample images captured by our system. The intensity maps of the checker board taken by the ToF camera (top row), The images of the checkerboard taken by a commercial color camera (bottom row).

The results of our experiment are shown in Table1.

Table1. Successful checkerboard recognitions for our method, OpenCV, OCamCalib[44]

Camera Setup	Total images	Our method	OpenCV	OCamCalib
Our binocular system	80	75	25	61
IDS uEye setup [45]	80	80	79	80
GoPro setup [46]	80	80	71	77

As shown in the Table1, our method 75 out of 80 checkerboards are recognized, but OpenCV only 25 and OCamCalib can detect 61. Our method is not only found but also recognized highly accurately. And the found points are used in estimating the intrinsic and extrinsic parameters.

Figure 6 illustrates the location and orientation of the black and white checkerboard. It can be seen that the checkerboard is placed in the vicinity of 1m because clearer intensity maps can be obtained at this area based on the exposure parameters of the ToF camera we set. We placed the checkerboard in different positions and orientations for the accuracy of the calibration.

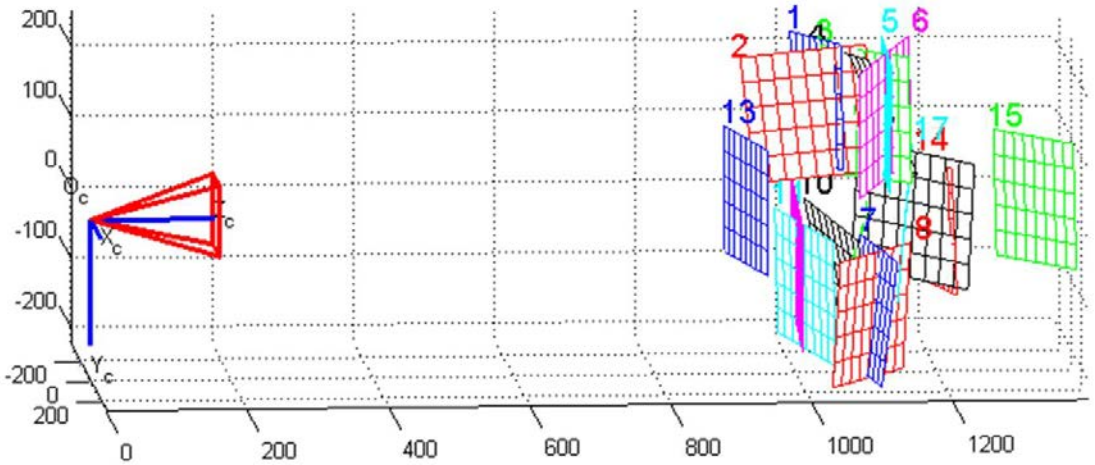
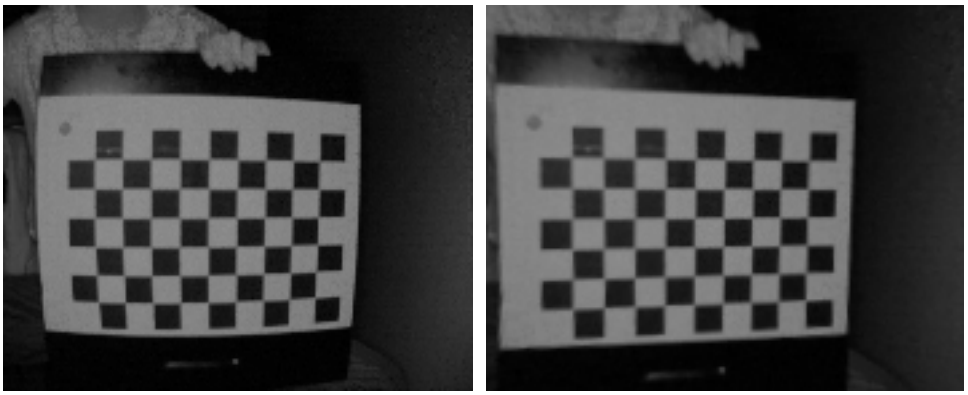


Figure 6.

Table 2 The result of the intrinsic parameters of the ToF calibration

Parameters	Value		
Focal length (f_x, f_y)	254.60，253.99	248.96，248.62	250.17，249.51
Optical center (c_x, c_y)	90.07，78.54	88.95，77.01	90.50，78.80
Radial distortion (r_1, r_2)	-0.9031，0.7380	-0.8719，0.6609	-0.8770，0.6704
Tangential distortion (t_1, t_2)	-0.0239，-0.0006	-0.0192，0.001	-0.0220 -0.0022

It is obvious in Table 2 that there is a certain fluctuation in the calibration results of the intrinsic parameters for the ToF camera, and the distortion parameters of the images taken by the ToF camera are relatively large. We need to correct the distortions of the intensity images before the binocular stereoscopic calibration. Figure 7 shows the intensity images before and after the distortion correction based on the distortion parameters. It is clear that the image is severely distorted in Figure 7(a), and the image has been greatly improved after correction in Figure 7(b).



（a）before distortion correction（b）after distortion correction

Figure 7. A comparison between the intensity maps before and after distortion correction

In order to verify the calibration results of the intrinsic parameters of the ToF camera, we back project the 3D checkerboard's corners in the physical space to the image space of the camera using the obtained intrinsic parameters. The back projection result is shown in Figure 8.

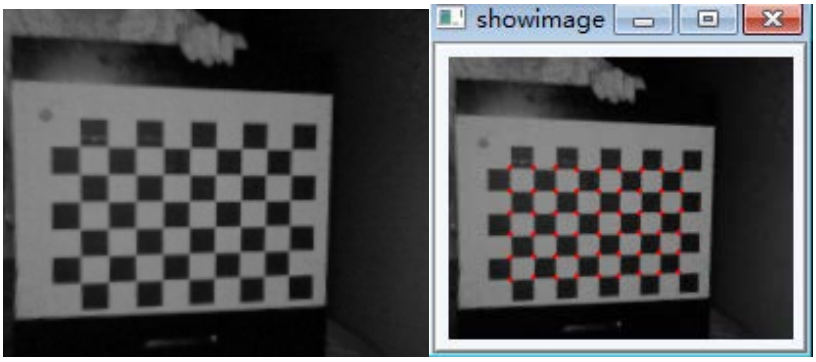


Figure 8. The back projection result

It is clear that the 3D checkerboard's corners are correctly projected back to the image, indicating that the calibration result is accurate.

In order to correct the depth of the ToF camera, we usea vision-based method to obtain the reference data. However, due to the low resolution of the ToF cameras, the accuracy of the

obtained reference data can be unsatisfying. Especially when the calibration object is too far from the ToF camera, the image of the object will be blurry that the corners are difficult to identify. In this paper, we solve this problem using a binocular system that consists of a commercial color camera and a ToF camera. Because the color camera can provide images of high resolution and the vision methods based on color camerasare more mature, we firstly obtain the reference data using the color camera, and then convert the reference data to the view of the ToF camera according to the positional relationship bwetween the two cameras. To obtain this positional relationship, the two cameras need to take pictures of the same calibration object in the physical space at the same time, and then the positional relationship can be calculated based on the position of the two cameras and the checkerboard. The binocular system can be calibrated using the 17 images taken by each camera, and the binocular calibration results is shown as follows:

$$\begin{bmatrix} 1.0000 & 0.0045 & -0.0055 & -59.2204 \\ -0.0043 & 0.9994 & 0.0357 & -15.9121 \\ 0.0056 & -0.0357 & 0.9993 & 30.2864 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix}$$

In order to obtain the reference data for correcting the system errors of the depth camera, we take color pictures of the checkerboard which is placed every 5cm from the position that is 0.6m away from the binocular system, and measured the distance between the cameras to the checkerboard using the ToF camera. The distance from the color camera to the checkerboard can be obtained using vision-based methods. And the distance from the ToF camera to the checkerboard can be calculated using the relative positional relationship between the two cameras, which will be used as reference data. After all the reference data and measured data are obtained, we get the system error curve of the ToF camera using uniform cubic B-spline interpolation as shown in Figure 9.

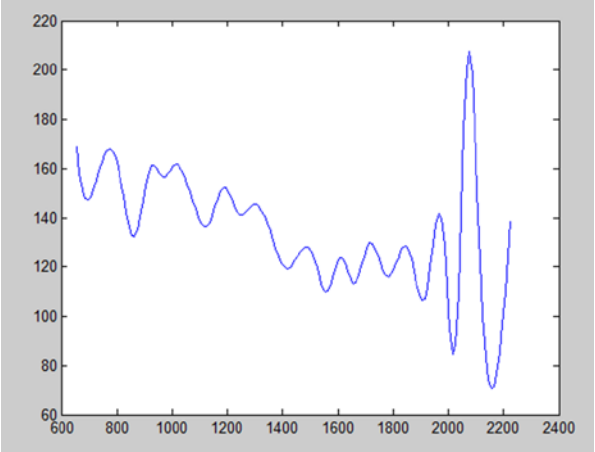


Figure 9 The interpolated system error curve of the ToF camera

It is clear that the system error curve is oscillatory, so the assumption that the system error is linear is wrong, and it is difficult to express the system error in polynomials.

Figure 10 shows the distribution of the reprojection error of the ToF camera. It appears relatively scattered because of the impact from the exposure time, illumination, and the reflectivity of the measured object. In the practical, using of the system requires to correct the errors according to the actual environment and camera parameters.

To further illustrate the accuracy of our method, we test whether increasing the number of checkerboards would enhance the accuracy of calibration. We examine between 3-17 patterns, as shown in Figure 11. It is easy to see that the increasing the number of patterns lead to smaller standard deviations (STDs) of the errors, thus it can get better accuracy. Therefore, we recommend no less than10 planes to obtain enough accuracy when calibration.

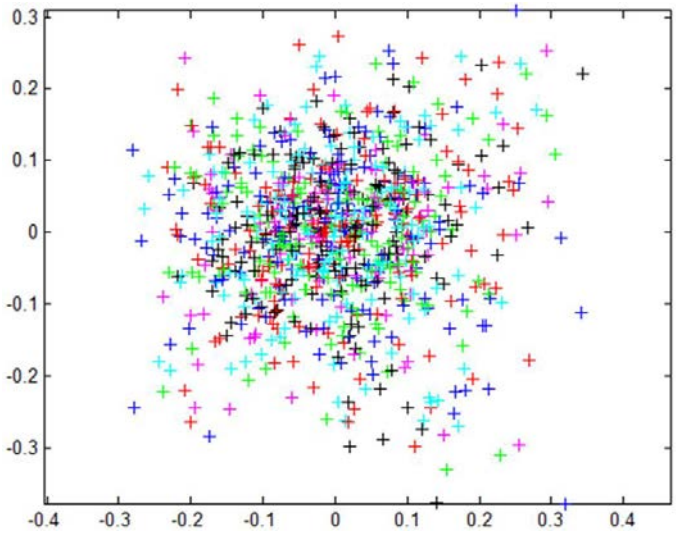


Figure 10 The reprojection error of the ToF camera

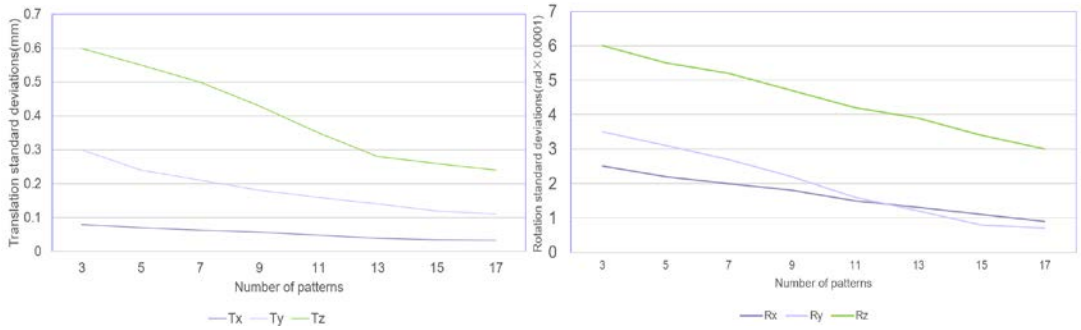


Figure 11. Calibration accuracy vs. the number of patterns

6.2 **Filtering and depth data test**

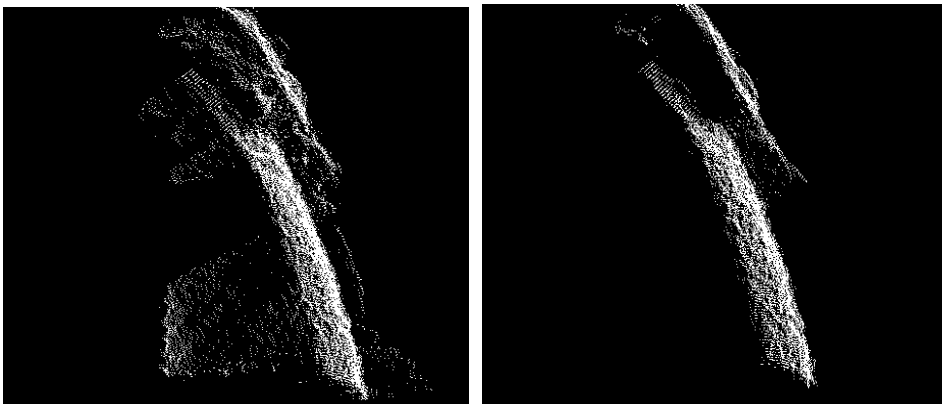


Figure 12 A comparison between before and after the anomalies removal

Figure 12 left shows the raw data generated by the ToF camera. The system errors have been removed. It is clear that the image contains plenty of abnormal points and noises. Figure 12 right shows the same image with the abnormal points removed. The removal result is satisfying.

Figure 13 left shows the 3D point cloud data before filtering. There are many random noises in this figure. Figure 13 right shows the same image processed by the improved bilateral filtering algorithm, which effectively removed the noises.

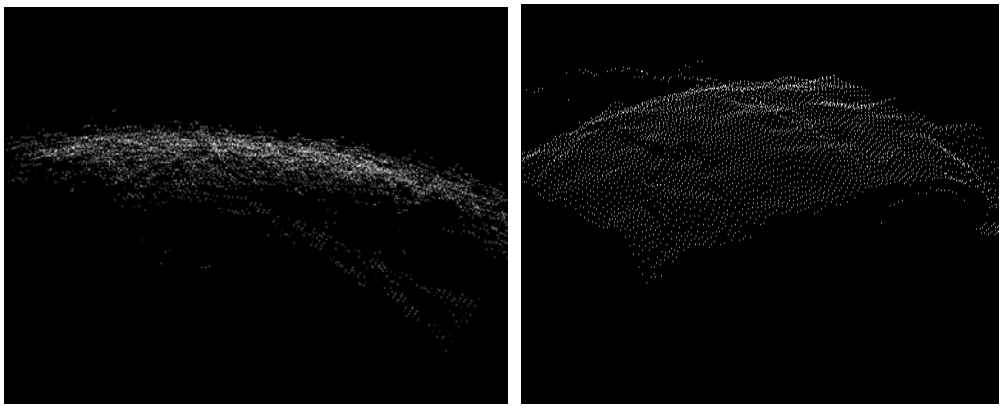


Figure 13 A comparison between before and after filtering

We apply the following method to verify the accuracy of the filtered data. Suppose each pixel of the optimized depth map corresponds to a point in the scene, so we can reconstruct the 3D point cloud data of the scene according to the depth map and the intrinsic parameters of the ToF camera. Giving the position of the color camera in the world frame, we obtain a 2D image of the scene by projecting the 3D point cloud to the image plane of the color camera. As shown in Figure 14, it is clear that the reconstructed object coincides with the image taken by the color camera, indicating that our filtering algorithm is valid.

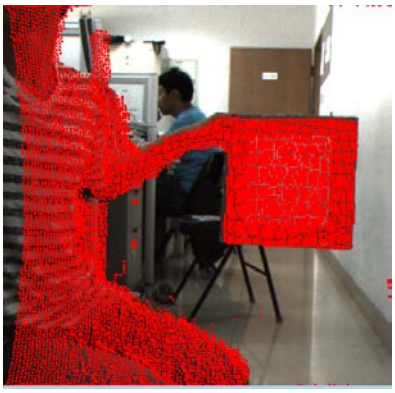


Figure 14 Point cloud data is projected to the view of color camera

6.3 Registration and test

During the experiment, we tracked two areas in the videos in real time, one is a hand and the other is a book. The tracking algorithm determined the range of the hand and the book in the color images space, and then retrieved the corresponding 3D point cloud data of the hand and the book from the optimized depth map that had been projected into the view of the color camera. The plane where the hand or the book was contained was fitted using the least squares method and displayed in the image space. As shown in Figure 15, the position and orientation of the hand and the book are different, and the result accurately shows the difference, indicating that the obtained depth map is accurate. In this paper, we implement a 3D registration method by tracking common objects instead of deliberate makers or markers with rich textures. Meanwhile we can effectively implement functionalities such as collision detection, occlusion between real and virtual objects, and layered positioning due to that we can obtain the depth information on all objects of in the scene.

Figure 15 The 3D registration results of different objects

Figure 16 shows the use of our method to complete the registration of the virtual teapot to the book and the hand. With the positional and directional change of the book or the hand, the position and orientation of the virtual teapot also changes. It is clear that the teapot is well integrated with the book and the hand.

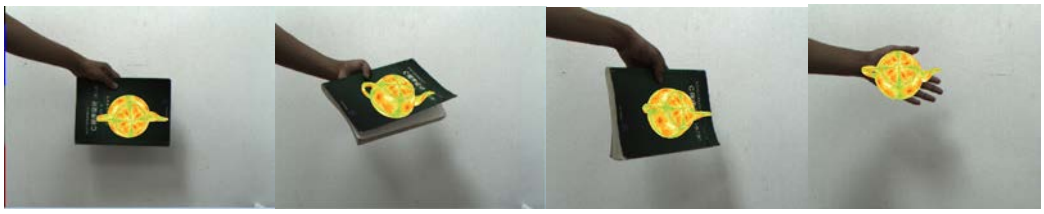


Figure 16 The results of integration with a single virtual object

Our system can realize multiple virtual objects' 3D registration of different levels of depth. Moreover, the binocular feature of this system facilitates real-time configuration of the positions of the virtual objects. As shown in Figure 17, we registered a virtual globe and a virtual teapot on a real desktop.

Figure 17 The results of integration with multiple virtual objects

7. Result

For meeting the accuracy and stability application of vision registration, a binocular stereo system that consists of a ToF camera and an ordinary industrial camera is built in our paper. We first calibrate the ToF camera and industrial camera to get their internal parameters and the relationship of position. We fit the systematic errors of the ToF camera using B-spline curve and use the curve to correct the system errors of the distance data obtained by the ToF camera. However, the distance data acquired by the ToF camera contains a lot of outliers and noises due to the defect of the hardware and the difficult light condition. With the help of the color information we design an algorithm of removing the outliers and an algorithm of smoothing the depth map. We get a refined depth map. Then we use a real-time tracking technology to determine the area where the virtual object needs to be registered. The position and orientation of the camera relative to the area in the real world from the depth map are calculated. In the end, we test some interactive applications of our method, and an AR system is designed and implemented using the binocular system. In the further, we will optimize and improve our method to meet the specific application of outdoor scene.

Acknowledgements

This paper is supported by National Natural Science Foundation of China (No.61402016, No.61502168, No.61502094), Social Science of Ministry of Education (14YJCZH200), Beijing Natural Science Foundation (No.4154067), Youth Talent project of Beijing (No.2016000026833ZK09), University Foundation of (No.XN018001) , Research Plan of Beijing (No.KM201610009008).

References

[1] Bok Y., Jeon H. G, and Kweon I. S. Geometric calibration of micro-lens-based light field cameras using line features. IEEE Trans. Pattern Analysis and Machine Intelligence, 39(2):287–300, 2017.

[2] Nousias S., Chadebecq F., Pichat J., et.al. Corner-based geometric calibration of multi-focus plenoptic cameras. In IEEE International Conference on Computer Vision(ICCV), poster, 2015, pp.1-9.

[3] Huang, S., Ying, X., Rong, J., Shang, Z., Zha, H. Camera calibration from periodic motion of a pedestrian. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp.3025-3033.

[4] Jeon H.G., Park J, Choe G., Tai Y.W., et. al. Accurate depth map estimation from a lenslet light field camera. In IEEE Int. Conf. Computer Vision and Pattern Recognition, 2015, pp. 1547–1555.

[5] Guido E., Matthijs H., Rob G. Automatic calibration of stationary surveillance cameras in the wild. In European Conference on Computer Vision(ECCV), 2016, pp.743-759.

[6] Bouaziz S., Tagliasacchi A., Li H. Modern techniques and applications for real-time non-rigid registration, SIFFRAPH ASIA Courses, 2016, No.11.

[7] Lin W., Cheng P.,Lin C.,et al. Intuitive 3D flight gaming with tangible objects, ACM SIGGRAPH Posters, 2016, No.78.

[8] Destelle F., Ahmadi A., Moran K., et al. A Multi-Modal 3D Capturing Platform for Learning and Preservation of Traditional Sports and Games, Proceedings of the 23rd ACM international conference on multimedia, 2015, 747-748.

[9] Bartili G., Bimbo A., Faconti M., et al. Emergency medicine training with gesture driven interactive 3D simulations, Proceedings of the ACM workshop on user experience in e-learning and augmented technologies in education, 2012, 25-30.

[10] L. Ma, C. Kerl, J. Stueckler, and D. Cremers. Cpa-slam: Consistent plane-model alignment for direct rgb-d slam. In ICRA, 2016.

[11] Jeong-Kyun Lee, Jaewon Yea, Min-Gyu Park, Kuk-Jin Yoon; Joint Layout Estimation and Global Multi-View Registration for Indoor Reconstruction. In IEEE International Conference on Computer Vision (ICCV), 2017, pp. 162-171.

[12] F. Arrigoni, B. Rossi, and A. Fusiello. Global registration of 3d point sets via lrs decomposition. In ECCV, 2016.

[13] Li C., Zhong F., Qin X., Accurate 3D Head Pose Estimation with noisy RGBD images, Proceedings of the 33rd computer graphics international, 2016, 37-40.

[14] Guo K., Xu F., Liu X. Real-time geometry, albedo and motion reconstruction using a single RGBD camera, Journal of ACM Transactions on Graphics(TOG), 2017, 36(4),No. 44a.

[15] Ishiguro H., Masashi Y., Saburo T., Omni-directional stereo, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992, 14(2), 257-262.

[16] Zhu J., Wang I., Yang R., Davis J. Fusion of time-of-flight depth and stereo for high accuracy depth maps. Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2008.

[17] B. Bartczak, I. Schiller, C. Beder, and R. Koch. Integration of a time-of-flight camera into a mixed reality system for handling dynamic scenes, moving view points and occlusions in real-time. Proceedings of the 3D PVT Workshop, Atlanta, GA, USA, June 2008.

[18] Li T., Xi X., Xie Y., et al. Reconstructing Hand Poses Using Visible Light, Proceedings of the ACM on Interaction, Mobile, Wearable and Ubiquitous Technologies, 2017, 3(1), 71-90.

[19] Zhang, Z.: Flexible Camera Calibration by Viewing a Plane from Unknown Orientations. In:

Proceedings of the International Conference on Computer Vision, Corfu, Greece ,1999, 666-673.
[20] Iindner M., Kolb A., Hartmann K. Data fusion of PMD-based distance-information and high resolution RGB-images. In IEEE Sym. on Signals Circuits& Systems (ISSCS), session on Alg. for 3D ToF-cameras, 2007, 121–124.
[21] Simon, G. Fitzgibbon, A. W. Zisserman, A. Augmented Reality Camera Tracking with Homo-graphics Augmented Reality, Proceedings of IEEE and ACM International Symposium, 2000, 22(6), 120-128.
[22] Comport, A.I., Marchand, E., Pressigout, M., et al. Real-time marker-less tracking for augmented reality: the virtual visual servoing framework. Visualization and Computer Graphics, IEEE Transactions on, 2006, 12(4), 615-628.
[23] Cheng Z., Ren J., Shen J, Miao H. Building a large scale test collection for effective benchmarking of mobile landmark search. Proceedings of International Conference on Multimedia Modeling, 2013, pp.36-46.
[24] Kolb A., Barth E., Koch R., Larsen R. Time-of-flight sensors in computer graphics. Computer Graphics Forum, 2010, 29(1), 141-159.
[25] Lange R. 3D Time-of-flight distance measurement with custom solid-state image sensors in CMOS/CCD-technology. PhD dissertaion, University of Siegen, 2000, 1-223.
[26] Darwish W., Tang S., Li W., et al. A new calibration method for commercial rgb-d sensors. Sensors, 2017.
[27] Wu C., Quigley A., David H. Out of sight: a toolkit for tracking occluded human joint positions. Personal and Ubiquitous Computing, 2017, 21(1), 125-135.
[28] Moreno D., Calakli F., Taubin G. Unsynchronized structured light. Journal ACM Transactions on Graphics (TOG), 2015, 34(6), No.178.
[29] Xtion PRO LIVE. Available online: https://www.asus.com/3D-Sensor/Xtion_PRO_LIVE/, 2016.
[30] Meet Kinect for Windows. Available online: https://dev.windows.com/en-us/kinect, 2016.
[31] Wang, K.; Zhang, G.; Bao, H. Robust 3D reconstruction with an RGB-D camera. IEEE Trans. Image Process. 2014, 23, 4893–4906.
[32] Chow, J.C.K.; Lichti, D.D. Photogrammetric bundle adjustment with self-calibration of the primesense 3D camera technology: Microsoft Kinect. IEEE Access, 2013, 22, 465–474.
[33] Wang, Y.-T.; Shen, C.-A.; Yang, J.-S. Calibrated Kinect sensors for robot simultaneous localization and mapping. In Proceedings of the 2014 19th International Conference on Methods and Models in Automation and Robotics (MMAR), 2014.
[34] Herrera, C.D.; Kannala, J.; Heikkila, J. Joint depth and color camera calibration with distortion correction. IEEE Trans. Pattern Anal. Mach. Intell. 2012, 34, 2058–2064.
[35] Tang, S.; Zhu, Q.; Chen, W.; Darwish, W.; Wu, B.; Hu, H.; Chen, M. Enhanced RGB-D mapping method for detailed 3D indoor and outdoor modeling. Sensors 2016, 16, 1589.
[36] Zhang, C.; Zhang, Z. Calibration between depth and color sensors for commodity depth cameras. In Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), 2011, 47-64.
[37] Harris C and Stephens M. A Combined Corner and Edge Detector. Proceedings of the Fourth Alvey Vision Conference, 1988, 147-151.
[38] Tissainayagam P., Suter D. Assessing the Performance of Comer Detector for Point Feature Tracking Applications, Journal of Image and Vision Computing, 2004, 22(8), 663-679.

[39] Iindner M., Kolb A., Hartmann K. Data fusion of PMD-based distance-information and high resolution RGB-images. In IEEE Sym. on Signals Circuits& Systems (ISSCS), session on Alg. for 3D ToF-cameras, 2007,121–124.
[40] Iindner M., Kolb A. Lateral and depth calibration of PMD-distance sensors. In Proc. Int. Symp. On Visual Computing, LNCS, Springer, 2006, 524–533.
[41] Huhle B., Schairer T., Jenke P., Strasser W. Robust non-local denoising of colored depth data. In IEEE Conf. on Computer Vision & Recognition, Workshop on ToF-Camera based Computer Vision, 2008, 1-8.
[42] Tai J, Tsang S, Lin C, Song K. Real-time image tracking for automatic traffic monitoring and enforcement application. Image and Vision Computing, 2004, 22(6):485-501.
[43]OpenCV Tools. http://graphicon.ru/oldgr/en/research/calibration/opencv.html.
[44] Scaramuzza, D.: Omnidirectional Vision: from Calibration to Root Motion Estimation. Ph.D. thesis, Swiss Federal Institute of Technology Zurich (ETHZ), 2008.
[45] IDS Imaging Development Systems GmbH, Obersulm, http://www.ids-imaging.com
[46] GoPro. https://gopro.com/.