We are IntechOpen, the world's leading publisher of Open Access books
Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Catadioptric Omni-directional Stereo Vision and Its Applications in Moving Objects Detection

Xiong Zhihui, Chen Wang and Zhang Maojun
College of Information System and Management, National University of Defense Technology
Changsha, P.R. China

1. Introduction

Catadioptric Omni-Directional Stereo Vision (ODSV) technology is the combination of omni-directional imaging and stereo vision, which has wide application potentials on robot vision and large-scale video surveillance [1-4]. Fig.1 gives the framework of ODSV technology, which includes four major parts: the design of omni-directional stereo vision imaging system, unwarping of omni-directional stereo images, rectification of omni-directional stereo images, stereo matching and depth estimation of omni-directional stereo vision.

Among these four parts, the imaging system can be used to capture omni-directional stereo image pair(s), which is the input of omni-directional stereo vision. An omni-directional stereo vision imaging system is typically composed of catadioptric mirrors, imaging sensors and fasteners.

The purpose of unwarping the omni-directional stereo images is to convert the circularity shaped omni-directional images into perspective projection images, which are suitable for human watching. Generally, we call the circularity shaped images captured by catadioptric omni-directional imaging system as omni-directional images, and we call the unwarped images that are suitable for human watching as panoramic images.

Rectification of omni-directional stereo images can be regarded as the pretreatment before stereo matching. In many cases, there are horizontal errors and vertical errors in the omni-directional images and panoramic images, these errors result in large searching space and mismatching when performing stereo matching. The rectification of omni-directional stereo images uses epipolar geometry to transform the images, which makes the matching points lie on a horizontal scan line, and reduce the searching space from two-dimension to one-dimension, so as to improve the stereo matching efficiency.

Stereo matching and depth estimation of omni-directional stereo vision are key problems in catadioptric omni-directional stereo vision, whose main function is to find correspondences between pixels among a pair of or more reference images, i.e. to estimate relative disparity for each pixel in reference images. Given pixel correspondence and calibrated camera, it is easy to figure out the depth information via triangulation for the determinate relationship between disparity and depth.

Taking its advantages of large FOV (Field of View) and depth information, catadioptric omni-directional stereo vision can be widely used in robot vision and video surveillance. For example, in robot football games, we can use this technology to make robots to "see" the

football whenever it is at any direction. Furthermore, since the depth information is not sensitive to the surrounding disturbance, this depth information can be used to detect the football position.
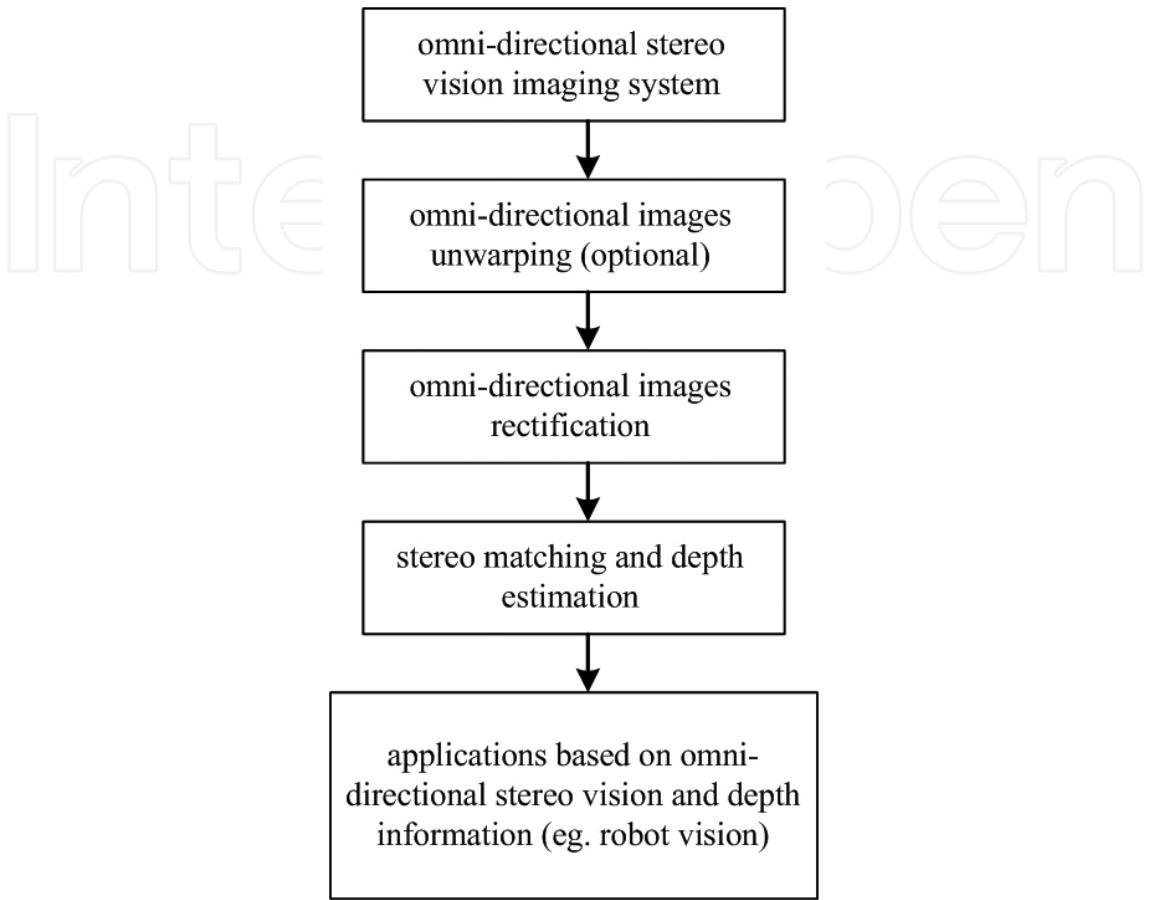

Fig. 1. Framework of catadioptric omni-directional stereo vision

2. Omni-directional stereo vision imaging system

2.1 Principle of catadioptric omni-directional imaging

The principle of the catadioptric omni-directional imaging is following: in the 3D space environment, rays from all the 360 degree FOV (Field of View) objects are collected and reflected by curve faced reflecting mirrors (such as paraboloid, hyperboloid, etc.); these reflected rays are captured by imaging systems and omni-directional images are obtained [5].

Taking the PROIS (Paraboloid Reflective Omni-directional Imaging System) [5] s an example, Fig.2 illustrates the principle and process of the catadiopric omni-directional imaging: 1) Incident rays from the scene are reflected by the paraboloid mirror. 2) Reflex rays run into the optics imaging system and forms an omni-directional image on the image sensor (such as CCD of a digital camera). 3) Finally, the omni-directional image is unwarped into panoramic image.

2.2 Omni-directional stereo imaging system

At present, there are three ways to construct the catadioptric omni-directional imaging systems [6], shown in Fig.3:

1. The pairs of omni-directional imaging systems are installed horizontally, and the axes of the pairs of imaging systems are vertical with the horizontal plane as shown in Fig. 3(a). However, there are occlusions between these two systems, which limit the FOV of the stereo vision system.

2. The pairs of omni-directional imaging systems are installed vertically on the same vertical axes, as shown in Fig. 3(b). Using this type of design, there exist no occlusions between these two systems, and it ensures that there exists parallax on 360 degree FOV. In this type of systems, the mirror can be reflecting mirrors (such as paraboloid, hyperboloid, etc.). When the system uses paraboloid as the reflecting mirrors, it requires using expensive telecentric lens.

3. Use two reflecting mirrors and a camera to construct an omni-directional stereo vision system, as shown in Fig. 3(c). This type of stereo imaging system requires that the two images of stereo vision are captured with a single camera, which limits the imaging resolution much. According to the principle of Fig. 3(c), Fig. 4 presents the design of a realistic omni-directional imaging system and its picture. As shown in Fig. 4(a), this system installs an *expensive telecentric* camera on vertical axes, and the upper mirror and nether mirror are also installed on the same axe. Fig.4(b) shows the practical picture of this system.
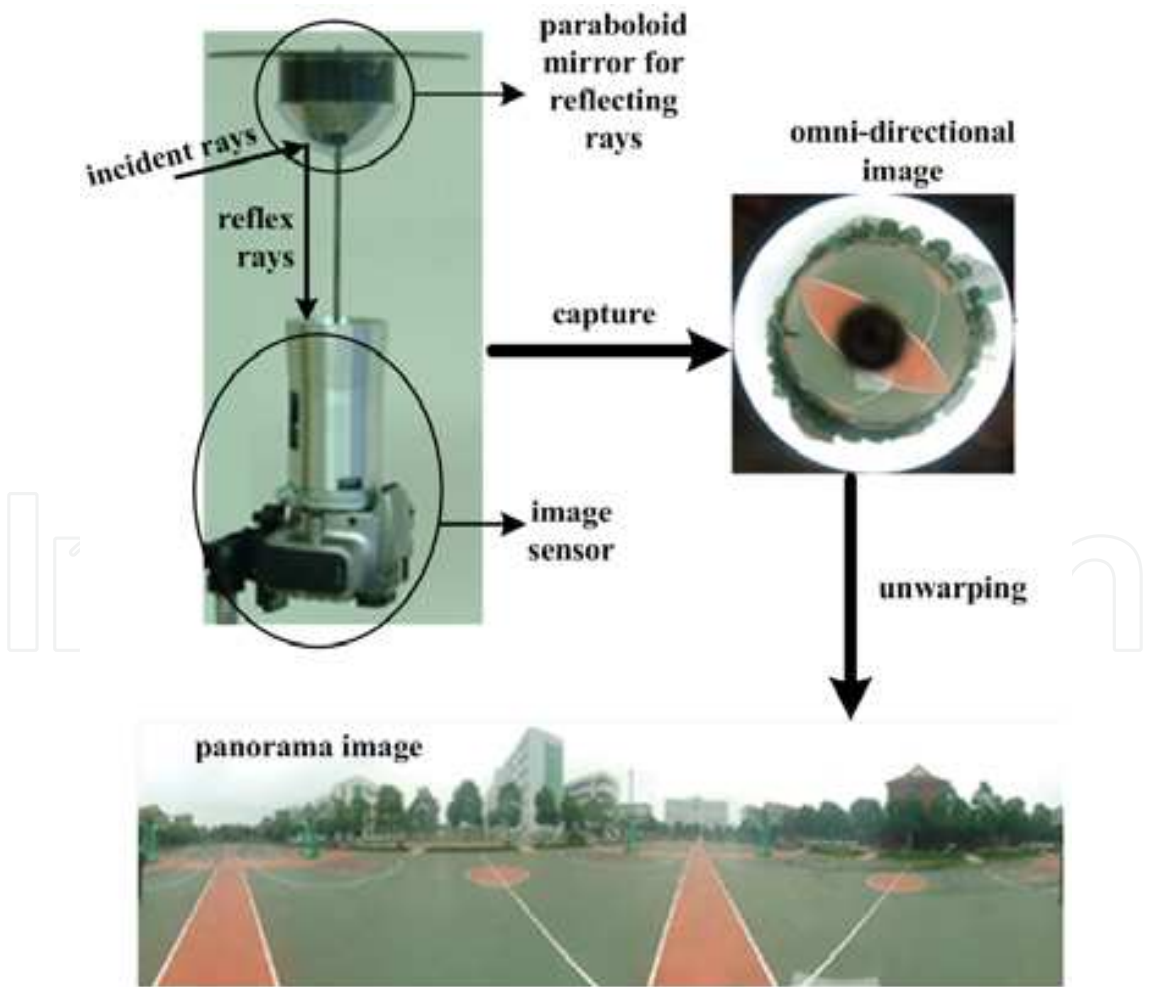

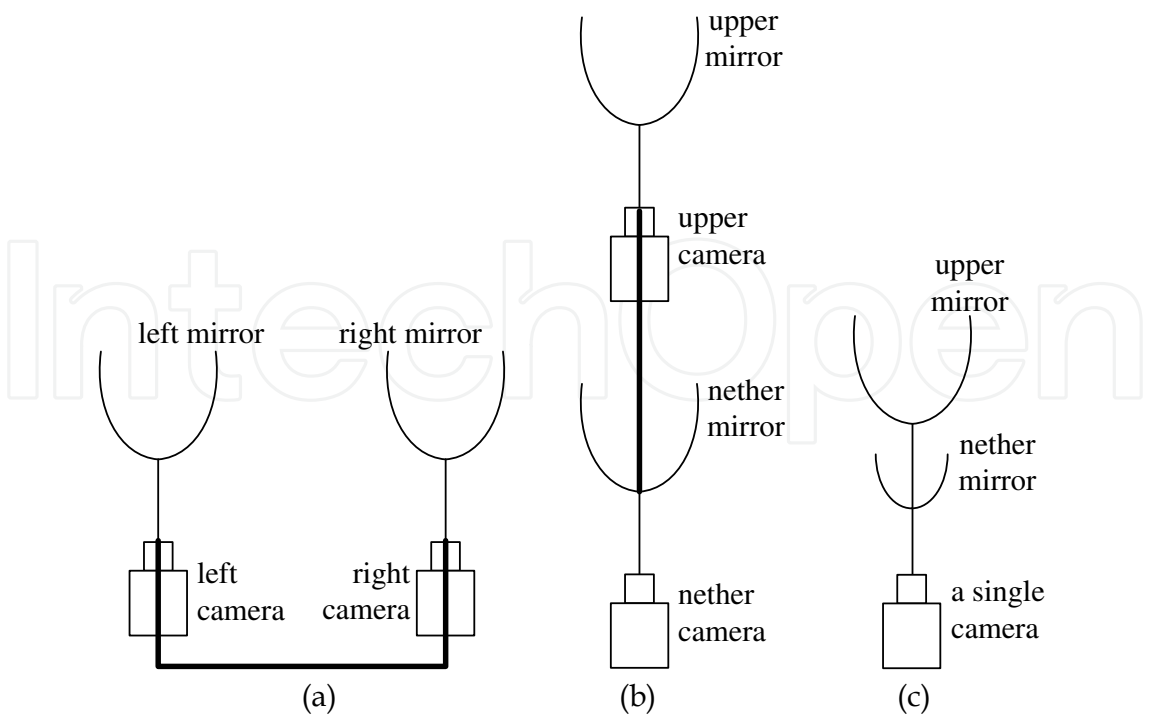Fig. 2. Principle of catadioptric omni-directional imaging and its unwarping

Fig. 3. Three ways to construct omni-directional stereo vision imaging system.
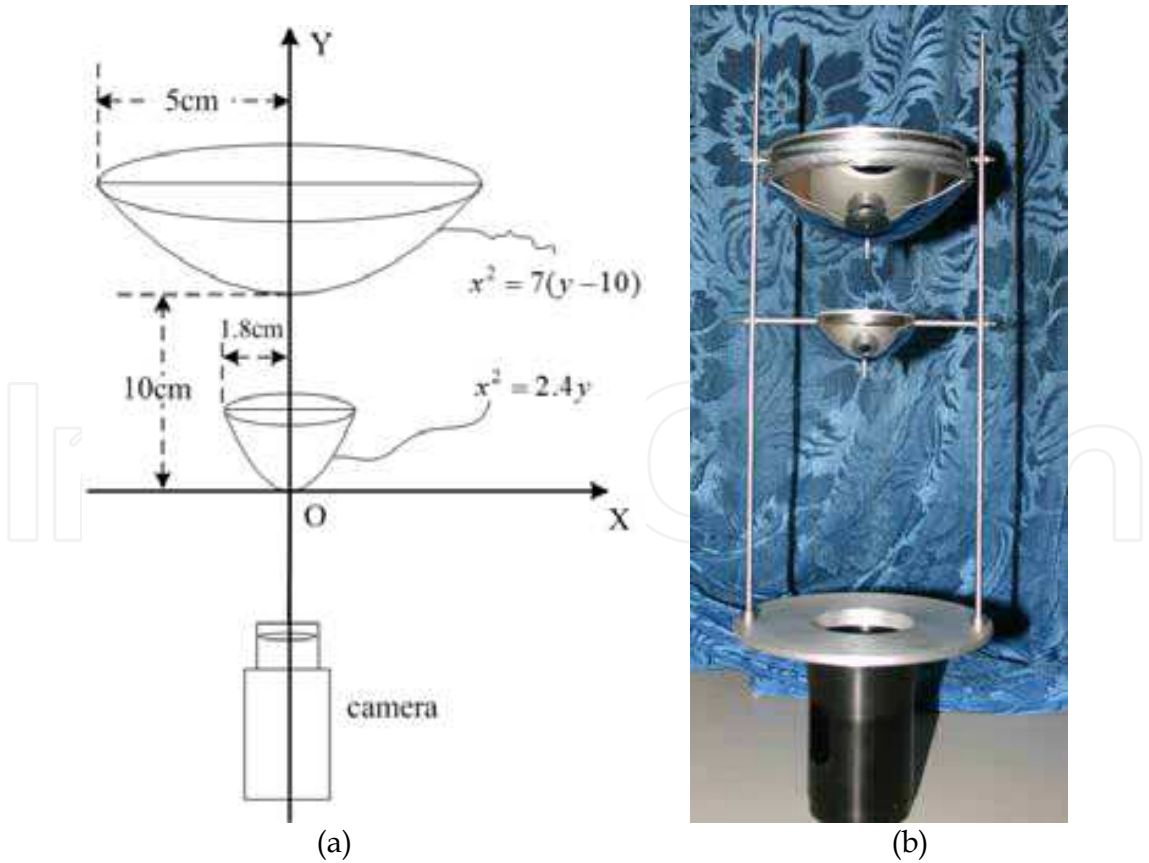(a)horizontal cameras; (b)vertical cameras; (c)vertical and single camera.

$$x^2 = 7(y-10)$$

$$x^2 = 2.4y$$

Fig. 4. Design and practical picture of single camera omni-directional stereo imaging system.
(a)design size; (b) practical picture.

3. Unwarping of omni-directional stereo vision images

There are two methods to perform stereo matching in omni-directional stereo vision. The first method is to perform stereo matching on the original images captured by the omni-directional imaging systems. When we use this method, we need to research and put forward new matching algorithms for omni-directional images. Another method is to unwarp the original captured omni-directional images into panoramic images, and then uses traditional stereo matching algorithms to finish the stereo matching on the panoramic images. When we use the second method, we need to unwarp the omni-directional images into panoramic images, as shown in Fig.5 (Note: the omni-directional image is captured with a virtual imaging system using 3D MAX).
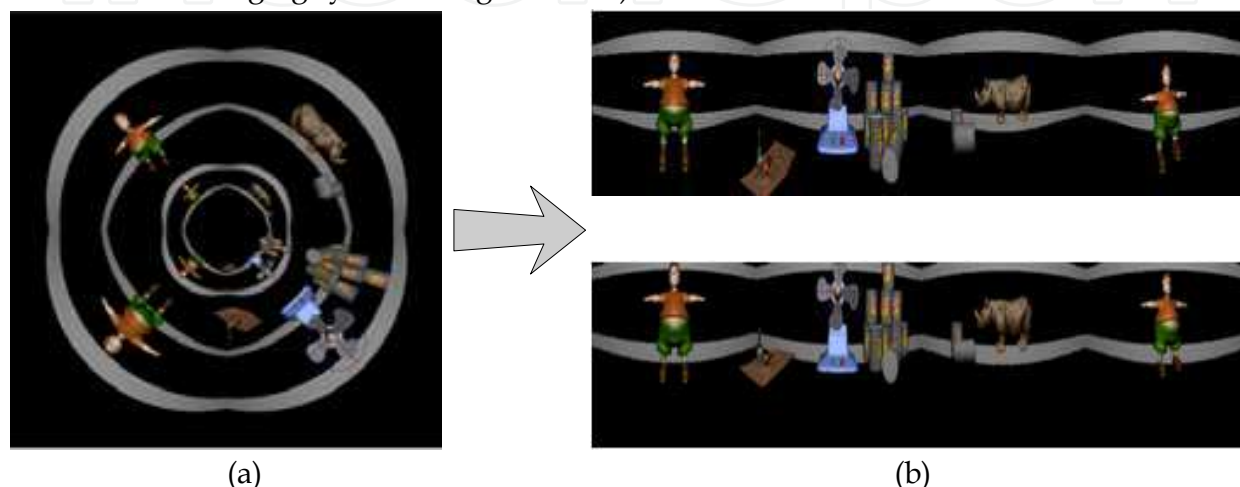

| (a) | (b) |

Fig. 5. Omni-directional stereo vision image is unwarped into panoramic stereo vision image pair.(a)Omni-directional stereo vision image;(b)Panoramic stereo vision image pair.

For simplicity, we take the unwarping of a single omni-directional image into panoramic image as an example to describe the unwarping of omni-directional image. Presently, some general approaches for omni-directional image unwarping include ray-trace coordinate mapping, concentric circle approximate unwarping and look-up table unwarping [7].

Among these approaches, the ray-trace coordinate mapping method tracks and analyzes the propagation trace according to principles of light propagation and reflection, and draws a pixel coordinate mapping between the original omni-directional images and the unwarped panoramic images. This method unwarps panoramic images with both high precision and less distortions. The disadvantage is that this method needs heavy computation for coordinate mapping, which slows down the unwarping speed.

On the other hand, the concentric circle approximate unwarping method treats the omni-directional image as a series of concentric circles, and these circles are pulled straight and extended or compressed to the same length. This method needs lower computation time. But this method unwarps omni-directional images with bad visual effect because of its lower precision and higher distortions.

3.1 Concentric circle approximate method for omni-directional image unwarping
Fig.6 depicts the principle of concentric circle approximate unwarping. Fig.6(a) denotes a circular omni-directional image with inner radius r and outer radius R. The region between r and R is valid region. Fig.6(b) shows the corresponding unwarped panoramic image. For a

pixel $P_0^{'}(X_0^{'}, Y_0^{'})$ in the unwarped panoramic image, the coordinate of the corresponding pixel $P_0 (X_0, Y_0)$ in the original omni-directional image (Fig.6(a)) can be determined by the following equation.

$$\begin{cases} X_0 = (r + Y_0^{'}) * \sin\theta \\ Y_0 = (r + Y_0^{'}) * \cos\theta \end{cases} \tag{1}$$

Where $\theta = X_0^{'} / (r + Y_0^{'})$.

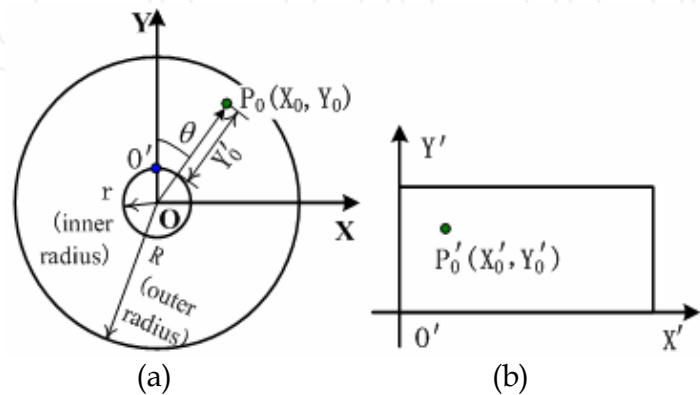(a)   (b)

Fig. 6. Principle of unwarping the omni-directional image into panoramic image. (a)omni-directional image;(b)panoramic image.

3.2 Look-up table method for omni-directional image unwarping

Fig.7 describes the principle of look-up table method for omni-directional image unwarping. First of all, this method calculates the pixel coordinate mapping relationship between the omni-directional image and panoramic image (e.g. using the ray-trace coordinate mapping method), and saves the mapping relationship into a loop-up table. After that, when performing the omni-directional image unwarping, for each pixel in the unwarped panoramic image, what we need to do is to get the mapping from the look-up table, and then get the corresponding pixel from the omni-directional image.
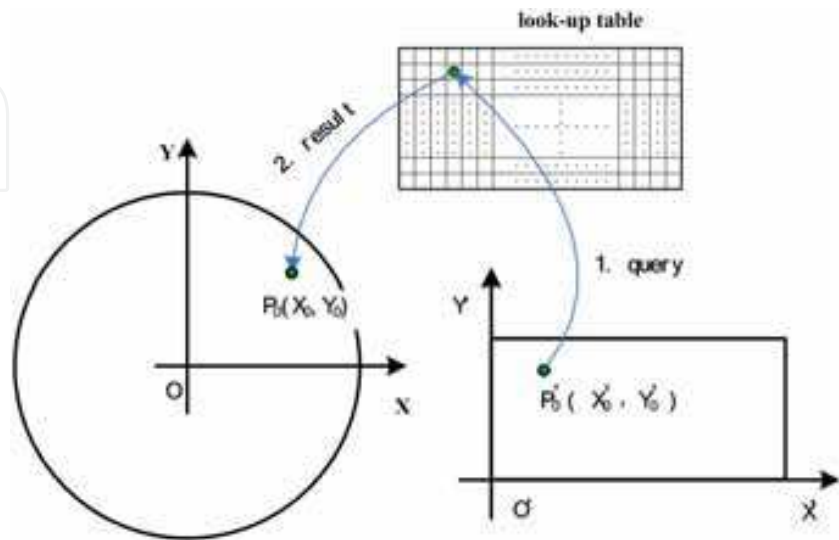

Fig. 7. Principle of look-up table method for omni-directional images unwarping

Advantages of the look-up table method including: this method unwarps omni-directional images with high precision (because this method generates the look-up table according to the results of ray-trace coordinate mapping); on the other hand, this method unwarps omni-directional images with high speed (since this method performs unwarping only by querying the look-up table and fetches the coordinate). Disadvantage of the look-up table method is: this method takes large storage space to store all the coordinate mappings (since we need to maintain an item for each pixel of the unwarped panoramic image).

3.3 Eight direction symmetry reuse algorithm for look-up table unwarping

In order to decrease the storage space needed for storing look-up table in the look-up table method for unwarping omni-directional images, we can use the eight direction symmetry reuse algorithm [7].

Eight direction symmetry reuse algorithm includes three steps: First of all, we sector the original omni-directional image into eight symmetrical regions and partition the target panorama image into eight rectangular regions (each corresponds to a sector). Secondly, based on any one of these sectors, compute the coordinate transform equation according to the principle of symmetrical transform. Finally, impose the ray-trace coordinate mapping function on only one sector, and perform symmetrical reusing process on the other seven sectors.

3.3.1 Eight direction symmetrical partition of omni-directional images

In order to reduce the storage space required to store the coordinate mappings, we reuse the coordinate mappings of the first sector using symmetrical principle. The strategy is to partition both the omni-directional image and cylinder panoramic images into eight symmetrical regions and find out the symmetrical relationship between them, and then reuse these relations.

Fig.8 indicates the principle of eight symmetrical partitions and eight direction symmetry reuse. In Fig.8(a), the coordinate system XOY is constructed at the centre of the omni-directional image O. The omni-directional image is divided into eight symmetrical regions named A_0, A_1,……, A_7. In Fig.8(b), we construct the coordinate system $X^{'}O^{'}Y^{'}$ at point $O^{'}$ and partition the target panorama into eight equal regions named $A_0^{'}$, $A_1^{'}$, ……, $A_7^{'}$.

After partitioning, the sectors A_0, A_1, ..., A_7 in Fig.8(a) are mapped to the rectangular regions $A_0^{'}$, $A_1^{'}$, ……, $A_7^{'}$ in Fig.8(b) respectively. We define the sector A_0 and the rectangular region $A_0^{'}$ as storing region, on which the coordinate mapping relationship need to be stored. The rest sectors A_1, A_2,……, A_7 and the reset rectangular regions $A_1^{'}$, $A_2^{'}$,……, $A_7^{'}$ are defined as reusing regions, on which the coordinate mapping relationship can be obtained by symmetrical transform reusing the computation result of storing region(A_0 and $A_0^{'}$).

3.3.2 Eight direction symmetry coordinate transform

For the point P_0 (X_0, Y_0) in storing region in Fig.8(a), its symmetrical points in the seven reusing regions A_i (i=1,2,3,...,7)are P_i (i=1,2,3,...,7). At the same time, the seven points P_i (i=1,2,3,...,7) in Fig.8(a) map to the seven points $P_i^{'}$ (i=1,2,3,...,7). Supposing the coordinates

(a) Eight direction symmetry points in omni-directional image

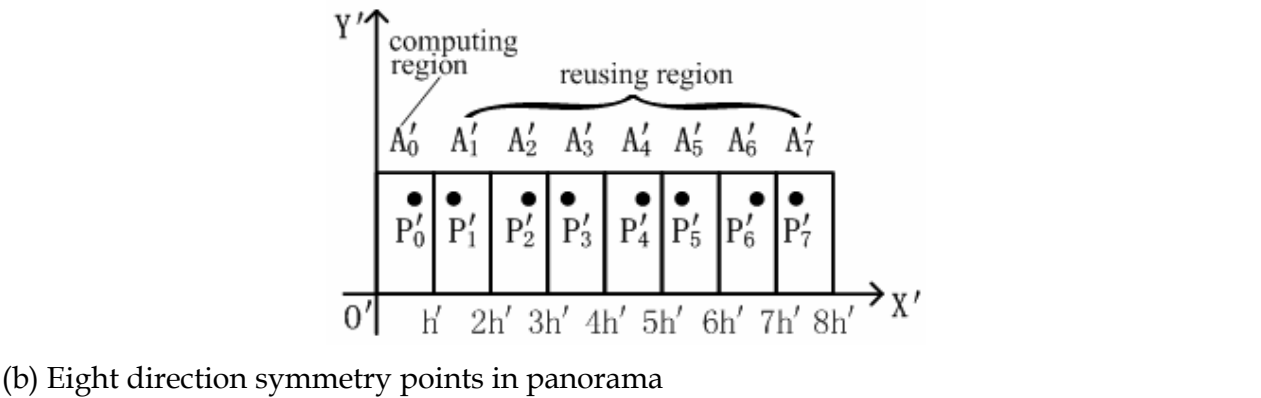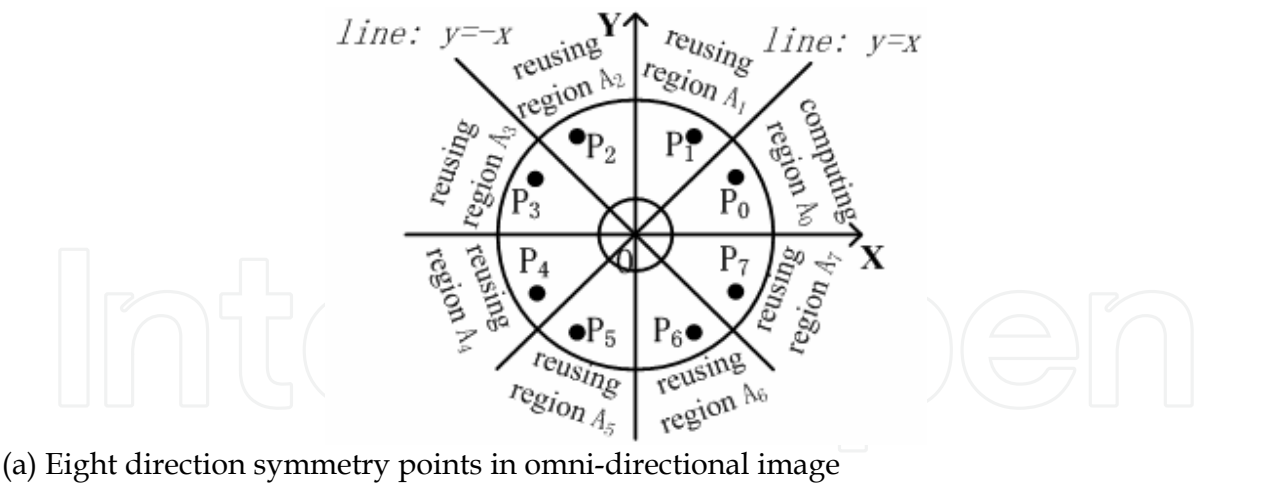(b) Eight direction symmetry points in panorama

Fig. 8. Eight symmetrical partition and symmetry reuse

of point P_0 is (X_0, Y_0), owing to the symmetry of P_1 and P_0 at line $y = x$, the coordinates of point P_1 is (Y_0, X_0). By the same token, P_2 and P_1 are symmetric with respect to y-axis and the coordinates of P_2 is $(-Y_0, X_0)$. The rest may be deduced by analogy, finally we get:

$$\begin{cases} P_1 = (\ y_0,\ \ x_0),\ \ P_2 = (-y_0,\ \ x_0) \\ P_3 = (-x_0,\ \ y_0), P_4 = (-x_0, -y_0) \\ P_5 = (-y_0, -x_0), P_6 = (\ y_0, -x_0) \\ P_7 = (\ x_0, -y_0) \end{cases} \tag{2}$$

In Fig.8(b), suppose the coordinates of point $P_0^{'}$ is $(x_0^{'}, y_0^{'})$, the length of each rectangular region is $h^{'}$, and the seven points $P_i^{'}$ are the correspondences of the points P_i (i=1,2,3,...,7) in Fig.8(a). From Eq.(2) we can see that points P_i (i=1,2,3,...,7) relocated on the same concentric circle, thus they are of the same vertical coordinate value $y_0^{'}$ (equal to the vertical coordinate of $P_0^{'}$). In addition, their horizontal coordinates $x_i^{'}$ satisfies the following expression:

$$x_i^{'} = \begin{cases} i * h^{'} + x_0^{'} & \text{if } i = 0 \quad 2 \quad 4 \quad 6 \\ (i+1) * h^{'} - x_0^{'} - 1 & \text{if } i = 1 \quad 3 \quad 5 \quad 7 \end{cases} \tag{3}$$

Therefore, if the coordinates $(x_0^{'}, y_0^{'})$ of point $P_0^{'}$ are known, the coordinates of the rest seven reusing points $P_i^{'}$ (i=1,2,3,...,7) are:

$$P_i^{'} = \begin{cases} (i*h^{'} + x_0^{'}, \ y_0^{'}) & \text{if } i = 2, \ 4, \ 6 \\ ((i+1)*h^{'} - x_0^{'} - 1, \ y_0^{'}) & \text{if } i = 1, \ 3, \ 5, \ 7 \end{cases} \qquad (4)$$

3.3.3 The eight direction symmetry reuse algorithm

The algorithm includes three steps: preprocessing, coordinate mapping and coordinate reusing.

// preprocessing

Step1. Define the centre of the omni-directional image, the inner and outer radius of the valid region;

Step2. Define the resolution (height and width) of the unwarped panoramic image;

Step3. Define parameters in the ray-trace coordinate mapping;

//perform coordinates mapping and coordinate reusing

Step4. For each pixel $P_0^{'}(x_0^{'}, y_0^{'})$ in the rectangular region $A_0^{'}$ of the unwarped panoramic image, implement step 5 to step 9;

// coordinate mapping

Step5. Fetch the coordinates of point from the look-up table as (X_0, Y_0), in region A_0 of the omni-directional image, where point P_0 is correspondent with point $P_0^{'}(x_0^{'}, y_0^{'})$;

Step6. Copy the pixel value of point $P_0 \ (X_0, Y_0)$ as that of $P_0^{'}$;

// coordinate reusing

Step7. Adopting Eq.(2), compute the coordinates (X_i, Y_i) of points P_i in the seven reusing region A_i of the omni-directional image respectively, and guarantee that P_i is the symmetrical point of P_0, where i=1,2,3,...,7;

Step8. Adopting Eq.(4), compute the coordinates $(x_0^{'}, y_0^{'})$ of points $P_i^{'}$ in seven reusing region $A_i^{'}$ of the unwarped panoramic image respectively, and guarantee that $P_i^{'}$ is the symmetrical point of $P_0^{'}$, where i=1,2,3,...,7;

Step9. Copy the pixel values of symmetrical points $P_i \ (x_i, y_i)$ in the seven reusing regions A_i of omni-directional image in step 7 as that of points $P_i^{'}(x_i^{'}, y_i^{'})$ in the seven reusing regions $A_i^{'}$ of unwarped panoramic image in step 8.

// complete the algorithm

Step10. When the implement traverses all the pixels in unwarped panoramic image, every pixel in other seven reusing regions of the unwarped panoramic image will get its correct pixel value.

4. Omni-directional stereo image rectification

After omni-directional images capturing and unwarping, we obtain cylindrical panoramic image pairs from the same scene. But it is not appropriate to carry out the stereo matching

and depth estimation directly on the obtained cylindrical images for randomicity of imaging positions between panorama image pairs, and a rectification process should be done first (imaging systems in Fig.3(b), 3(c) are special cases of omni-directional imaging system, i.e. in which the relationship between imaging position of image pairs is particular, and unwarped panoramic pairs need no rectification). There is only horizontal disparity within rectified image pairs, therefore, we can search the corresponding points just along image scan line direction, which is a faster and more accurate approach.

The principle of rectification for cylindrical image pairs and for ordinary perspective image pairs is similar, both based on epipolar geometry. But because of particularity of cylindrical imaging model, there is biggish difference between the two rectification processes. Some related work has been done. [8-10] studied the epipolar geometry of cylindrical image and relevant mathematical formula was given. [9-12] discussed image rectification and corresponding matching problem in plenoptic modeling, range estimation and 3D reconstruction based on cylindrical image, but did not mention how to rectify image pair and to search corresponding points. [13-14]proposed a particular panoramic camera fix for range estimation and investigated the rectification problem, however, they didn't discuss about rectification on arbitrary pose of cylindrical camera pairs. While [15] firstly segmented panoramic image into several parts along axes direction, then projected every part on to plane tangent to the cylinder, and at last rectified image with proposed method for planar image. When working with approach in [15], the homographic and perspective project transformation (HPPT in short) is necessary and therefore inevitable result in pixel information losing and image distortion [16-18], and eventually reduces the correctness of the corresponding matching. Although [17] proposed a method, which transformed a line in planar image into a vertical line of cylindrical image, to minimize by-effect of HPPT, this method, in view of particularity of cylindrical imaging, can not apply to cylindrical image directly.

According to recent study literatures, the normal approach for rectifying cylindrical image pairs is the way via epiline-sampling based on epipolar geometry. It samples reference images as much as possible to obtain better pixel-maintain-rate and effective-pixel-rate by analyzing the epipolar constrain of cylindrical image pairs, and has a fair rectification result for unwarped omni-directional images. Compared with approach of HPPT, this method can reduce image distortion and pixel information losing obviously and maintain the scene information better.

4.1 Epipolar geometry of cylindrical panoramic image pairs

Epipolar geometry is an important constraint for stereo images from camera pairs, it reveals the inner relationship of the pixel positions of special scene projected onto the two cameras.

As illustrated in Fig.9, let V_1, V_2 denote two viewpoints with origin v_1, v_2, and $Clind_1$, $Clind_2$ are two cylindrical panoramic images, p means a scene point with coordinates $p_1=(x_1,y_1,z_1)^T$ in V_1, $p_2=(x_2,y_2,z_2)^T$ in V_2. For generality, we assume coordinates V_1 is coincide with world coordinates. The plane spanned by v_1, v_2 and p is so-called epipolar plane, and its intersection lines ($Epiline_1$, $Epiline_2$) with $Clind_1$, $Clind_2$ are named epiline. Epipolar geometry of two views can be formalized as follows. Let p_{v1}, p_{v2} denote positions of a scene point in viewpoint coordinate of V_1 and V_2. The transformation between them is given by

$p_{v2}=Rp_{v1}+T$, where R is a (3x3) rotation matrix and T is a (3x1) translation vector. Then the normal of plane spanned by T and p_1 is $N_{TP1}=Txp_1$ where x denotes outer product. And the normal of plane spanned by T, p_2 is $N_{TP2}=RN_{TP1}=R(Txp_1)$. Since p_2 belongs to the intersection line between epipolar plane and $Clind_2$, Eq.(5) can be established by $N_{TP2} \cdot p_2=0$, i.e.
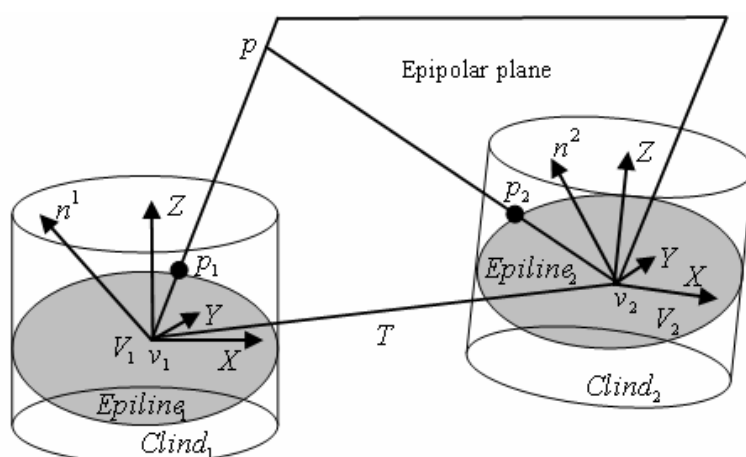

Fig. 9. Epipolar geometry of cylindrical image pairs

$$R(T \times p_1) \cdot p_2 = 0 \tag{5}$$

let $T = [t_x, t_y, t_z]^T$, and $[T]_x = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}$, then Eq.(5) can be expressed as

$$(R[T]_x p_1) \cdot p_2 = (Mp_1) \cdot p_2 = p_2^T M p_1 = 0 \tag{6}$$

Where $M = R[T]_x$ is a (3×3, rank 2) matrix. Eq.(5,6) presents the epipolar geometry of cylindrical panoramic image pairs.

4.2 Coordinate transformation

For what we actually get is the pixel coordinate of image, a transformation from pixel coordinates to camera coordinates is necessary. The transforming equation is:

$$p_1 = (x_1, y_1, z_1)^T = \begin{cases} x_1 = f \cos \alpha \\ y_1 = f \sin a \\ z_1 = (v_c - v_1) PixelSize \end{cases} \tag{7}$$

Where $\alpha = u_1 x 2\pi / ColNum$, f denotes camera focus, i.e. the radius of cylinder, (u_1, v_1) is pixel coordinates of p_1 on unrolling cylindrical image, v_c means the row index of center point on unwarped cylindrical image, $ColNum$ denotes width of unrolling cylindrical image, $PixelSize$ denotes the size of pixel. Obviously, the same transformation can by applied to p_2. Let $Mp_1=(k_{ij})_{3x1}=K$, according to Eq.(6,7), there is

$$K \cdot \begin{bmatrix} f\cos\alpha_2 \\ f\sin\alpha_2 \\ (v_c - v_2)PixelSize \end{bmatrix} = 0$$ where $\alpha_2 = u_2 \times 2\pi / ColNum$. Without subscript components of α_2

and v_2, we have $\quad v = v_c + \dfrac{k_{11}f\cos\alpha + k_{21}f\sin\alpha}{k_{31}PixelSize}$, $\alpha = u \times 2\pi / ColNum$.

4.3 Epiline equation solving

When omni-directional image unwarping, it is obvious that: f can be fixed, and v_c, PixelSize is known (for the size of camera film is known). Therefore, we can only consider solving epipolar geometry equation under the following two cases:

1. Cylindrical camera has been calibrated, i.e. the values of R , T , f , v_c , $PixelSize$ are known. Then Eq.(6) will be a group of linear equations about p_2 if coordinates of p_1 (u_1, v_1) is known. And vice versa.

2. Cylindrical camera has not been calibrated. Thereby, we should firstly estimate fundamental matrix M . Let $M = (m_{ij})_{3\times3}$, Eq.(6) can be expressed as

$$m_{11}x_1x_2 + m_{12}y_1x_2 + m_{13}z_1x_2 + m_{21}x_1y_2 + m_{22}y_1y_2 + m_{23}z_1y_2 + m_{31}x_1z_2 + m_{32}y_1z_2 + m_{33}z_1z_2 = 0 \qquad (8)$$

If coordinates of (x_1, y_1, z_1) and (x_2, y_2, z_2) are known, Eq.(8) will be a nine variables linear equation with constrain of $\det(M) = 0$. Referring to 8-points algorithm of estimating fundamental matrix, we can get M similarly and then the corresponding epiline equation of every point.

4.4 Rectification via epiline sampling

To reduce distortions and pixel-information-losing of rectified images, it should sample the reference image maximum according to epipolar geometry.

For generality, assume the radius and height of $Clind_1, Clind_2$ is f_1, f_2, h_1, h_2 , where $f_1 = f_2, h_1 = h_2$) f and h can be fixed when omni-directional unwarping), V_1, V_2 denote two viewpoints with origin v_1, v_2 , X axes sets along $\overrightarrow{v_1v_2}$,and the coordinates V_1 is coincide with the world coordinates (see Fig. 10). For simplicity, we study the epiline sampling algorithm on $Part_1$ and $Part_2$, parts of the cylindrical images.

Let $e_1 = (e_x^1, e_y^1, e_z^1)$, $e_2 = (e_x^2, e_y^2, e_z^2)$ be the two epipolars which have minimal distance among the four epipolars on $Clind_1$, $Clind_2$, since e_1, e_2 must be on all of the epipolar planes, i.e. all of the epipolar planes are composed of the plane cluster which intersect at $\overrightarrow{e_1e_2}$. Let the angle between epipolar plane and PH which is a plane spanned by $\overrightarrow{e_1e_2}$ and point

p_f ($p_f = (-me_y^1, me_x^1, 0)$, $m = \dfrac{f}{\sqrt{e_y^{12} + e_x^{12}}}$) on $Clind_1$ be β . In order to sample pixel

information on $Part_1$, $Part_2$ as much as possible, the value region ofs should be as large as possible, that is to say, epipolar plane should intersect $Part_1$, $Part_2$ maximally.
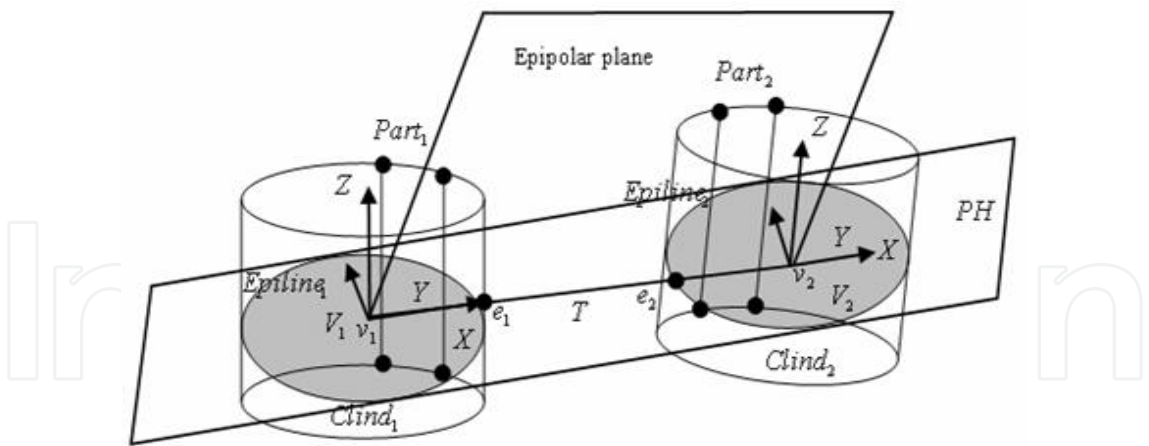
Fig. 10. Principle of epiline-sampling on cylindrical images

To calculate the largest value range of β, vertexes coordinates of $Part_1, Part_2$ should be taken into consideration. Let the coordinates of vertexes of $Part_1, Part_2$ be (x_i^j, y_i^j, z_i^j) where $i = 1, 2, 3, 4$ and $j = 1, 2$ (when $i = 1, 2$, (x_i^j, y_i^j, z_i^j) denotes vertexes above PH, otherwise, denotes vertexes below PH), the normal of plane spanned by (x_i^j, y_i^j, z_i^j), v_1, v_2 is n_i^j, and the normal of PH is n_{ph}, then the angle between these two planes is $\beta_i^j = \arccos \dfrac{n_i^j \cdot n_{ph}}{\left|n_i^j\right|\left|n_{ph}\right|}$. Thereby when the value region of cut angle β between PH and epipolar plane intersecting $Part_j$ is $[0, \max(\beta_{i=3,4}^j)]$ below PH, and $[0, \max(\beta_{i=3,4}^j)]$ above PH, the intersection line between epipolar plane and $Clind_1, Clind_2$ can sample the pixel information of $Part_1, Part_2$ as much as possible.

Following is the algorithm of calculating epilines on $Part_1, Part_2$. Let $\max(\beta_{i=1,2}^j) = \beta_m^l$, $\max(\beta_{i=3,4}^j) = \beta_n^k$, $m, l, k \in \{1, 2\}, n \in \{3, 4\}$, the Z coordinate of intersection point between line defined by $\begin{cases} x = x_m^l \\ y = y_m^l \end{cases}$ and plane spanned by $(x_m^l, y_m^l, z_m^l), e_1, e_2$ be z_{max}, and Z coordinate of intersection point between the same line and plane spanned by $(x_n^k, y_n^k, z_n^k), e_1, e_2$ be z_{min}, then for any point on

$$\begin{cases} x = x_m^l \\ y = y_m^l \\ z_{min} \leq z \leq z_{max} \end{cases} \qquad (9)$$

we can figure out the corresponding epiline equation on $Clind_2$ referring to $(Mp_k) \cdot (x, y, z)^T = 0$, and corresponding epiline on $Clind_1$ as the intersection of cylinder and epipolar plane where the cylinder is defined as

$$\begin{cases} x^2 + y^2 = f^2 \\ 0 \le z \le h \end{cases} \qquad (10)$$

and the epipolar plane is defined as

$$[(p_k - e_1)[(p_k - e_2)]_x] \cdot [p_k - (x, y, z)] = 0 \qquad (11)$$

With Eq.(10,11), epiline corresponding to p_k on $Clind_1$ can be solved. When set the value of z_k from z_{min} to z_{max}, we can sample $Part_1$, $Part_2$ by pairs of epilines as many as $(z_{max} - z_{min})$.

4.5 Rectification experiments

Corresponding rectifying experiments on part of the cylindrical image (Fig.11(a),Fig.12(a)) via algorithm of HPPT[15] and epiline-sampling are shown in Fig.11(b,c) and Fig.12(b,c). As experiments illustrated, rectifying with HPPT resulted in worse image distortions and resolution losing, while the approach based on epiline-sampling shows a better result, for it keeps the length of epiline and height and width of the image unchanged and reduced pixel information losing, as shown in Fig.11(c) and Fig.12(c).

(a)   (b)   (c)
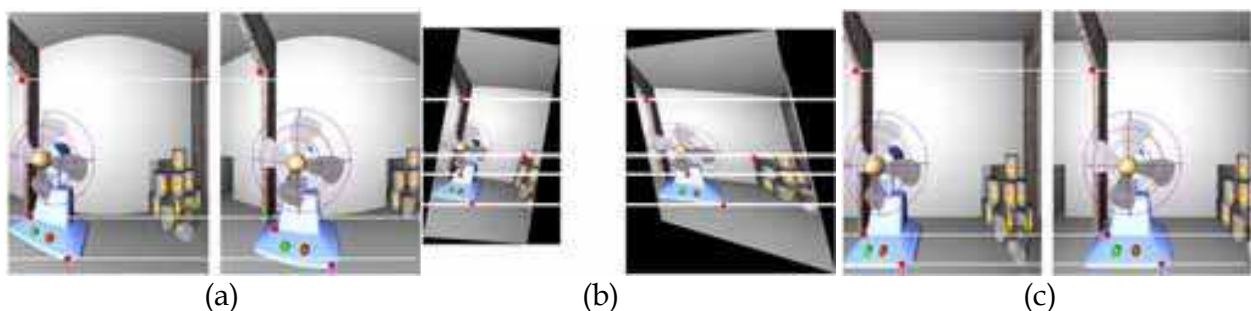
Fig.11 (a): Uncalibrated image pair of part of synthesized cylindrical images (corresponding points are not in the same scan line); (b): Rectified image pairs corresponding to images in (a) using the method [15] (corresponding points are in the same scan line); (c): Rectified image pairs corresponding to images in (a) using the proposed method (corresponding points are in the same scan line)

(a)   (b)   (c)
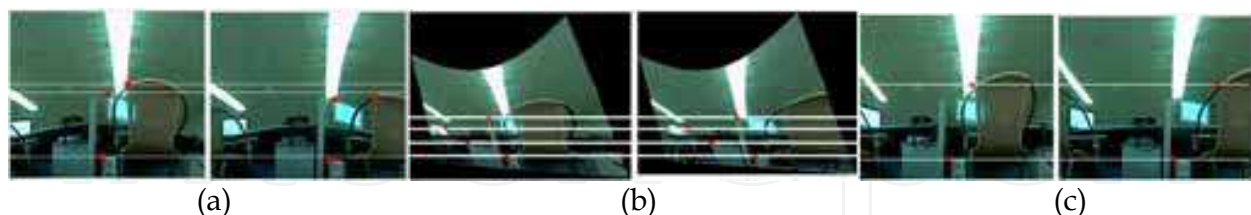
Fig. 12. (a): Uncalibrated image pair of part of real cylindrical images (corresponding points are not in the same scan line); (b): Rectified image pairs corresponding to images in (a) using the method [15] (corresponding points are in the same scan line); (c): Rectified image pairs corresponding to images in (a) using the proposed method (corresponding points are in the same scan line)

5. Stereo matching and depth estimation

After rectification of omni-directional image pairs, stereo matching should be done on rectified images to recovery depth information. Given camera calibrated, it is easy to

figure out depth information from disparity via triangulation because of correspondence between disparity and depth. In this way, the problem of depth recovery is equal to finding correspondence within image pairs or series of images, i.e. stereo matching. However, stereo matching is a hard work for the reasons of noises, occlusion and perspective distortion.

At present, researches in this field mainly differ in problem modeling and solving, they can be categorized into algorithms based on feature information, region information, and local or global approaches based on pixel information. Each method has their own advantage and disadvantage, for example, algorithms based on feature information is robust against noise, but can only obtain sparse disparity image; algorithms based on region matching although can get dense disparity map, but its reliability is poor and result is incorrect in the region of untextured, occluded and discontinuity; algorithm based on local information is faster compared to other ones but more sensitive to image local features and apt to produce wrong result.

However, approaches with global graph cuts (GC) optimization are widely studied and showing strong performance in last few years. [19-21] utilized GC optimization in stereo matching decade ago; Yuri Boykov and Vladimir Kolmogorov developed new faster GC algorithm[23-27] on the basis of[19,21-22], which coped smoothness problem of resultant disparity map in the depth-jump area. Meanwhile, [28, 29, 31] adopted occlusion restriction in GC optimization, but occlusion itself is still too complex to deal with. [29-32] firstly initialized a disparity map and segmented it, then fitting each segment with a planar equation and labeling them via GC optimization, , therefore, a better result can be obtained.

In this section, methodology and processes of stereo matching via GC optimization will be introduced in the first instance, and then a stereo matching algorithm based on region restriction and GC optimization will be illustrated in detail. This algorithm assumes disparity jumps only at the region of color discontinuity, and constructs energy function subjected to restriction between region boundaries. In this way, not only the global solution of energy function can be work out, but also that the solution has the feature of discontinuity-preserving, meanwhile embodies occlusion restriction, ordering restriction and uniqueness restriction. Additional, for energy function is only constructed based on boundary pixels, the number of graph vertex in GC optimization should certainly reduce significantly and results in great efficient performance.

5.1 Energy minimization presentation of stereo matching

Stereo matching between image pair can be compared to a problem of carbonization optimization. For a scene of limit depth, the corresponding discrete integral disparity is also a limit set, and our goal is to find (according to some rules) an optimal combinatorial configuration of disparity for each pixel in stereo images. Energy minimization is a common solution for this issue, which is usually presented as:

$$\min \ E(f) = \sum_{p \in I} D_p(f_p) + \sum_{\{p,q\} \in N} V_{p,q}(f_p, f_q) \tag{12}$$

Where p, q denote pixels with neighborhood system N in image I, f is the combinatorial configuration of disparity for every image pixel. $D_p(f_p)$ (called data term) expresses color consistency of pixel p when its disparity is f_p. $V_{p,q}(f_p, f_q)$ (called smoothness term) means smoothness between pixels p with disparity f_q and q with disparity f_q.

5.2 GC optimization

To solve Eq.(12), there are many approaches such as simulating anneal, dynamic programming and neural network, but these methods are either fit for high dimension dataset or hard to converge, and usually are inefficient in terms of computation. Recent researches prove that algorithms based on min cuts are very appropriate for problem of combinatorial optimization. They can be categorized mainly into two sets:

1. GC algorithms obtaining global optimal solution

[20,21,33] regard combinatorial optimization as a labeling problem, and solving the problem via constructing a special graph whose min cuts/max flow just corresponds to optimal solution of energy function. To ensure a given energy function can be presented with graph, it demands smoothness term of energy function must be convex, therefore, result in bigger punish and over-smooth at boundary of disparity jump. Additional, for the convexity of smoothness term, some issues such as occlusion problem, ordering restriction can not be dealt with very well.

2. GC algorithms obtaining global second optimal solution

To settle the over-smooth problem, [23-25] proposed to adopt unconvex smoothness term to make energy function discontinuity-preserving. So called Potts model $V_{p,q}(f_p, f_q) = \lambda \times T(f_p \neq f_q)$ is a common simple smoothness term, however, even with Potts model, [23,26,28] proved it is NP-hard to optimize such energy function and there is no efficient global optimization algorithm. Therefore, [23,26,27]developed approximating ways to cope this problem by dividing optimization of multivariable energy function into iterative optimization of two-variable energy function, it meanwhile showed that, after certain iteration, the solution we get via this approach is just near the global optimal solution[21]. Because of unconvexness of smoothness term, such algorithm also can deal with occlusion, ordering problems and uniqueness. However, there are still some disadvantages. Approximating approach can only obtain second optimal solution in iterative way, it is hard to measure the computation complexity, and when take occlusion or ordering into consideration, it will bring in more additional computation.

5.3 Graph construction of GC optimization

Graph construction of GC optimization in this section is very similar to the way proposed in [21], as Fig.13 illustrated, where vertexes of graph network are possible matching pixel pairs, and edges of graph network denote neighborhood interaction and restrictions ensuring correspondence between a min cut of graph and a disparity configuration of stereo matching.
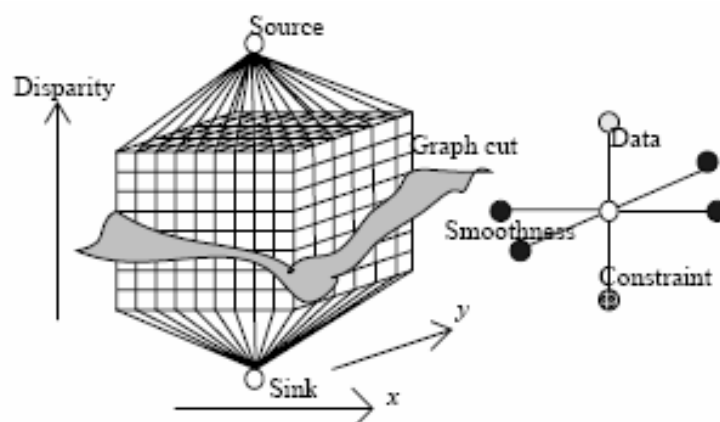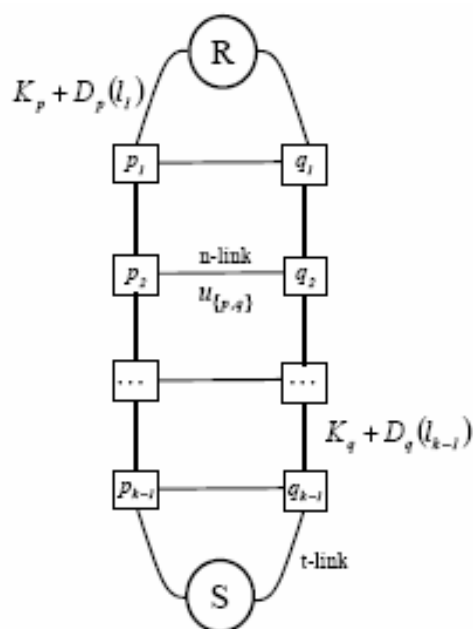
Fig. 13. A graph structure for stereo matching

Fig. 14. A subgraph of Fig.13 corresponding to pixels p and q

Fig.14 is a subgraph of Fig.13 corresponding to pixels p and q, and vertexes R and S are terminals *Source* and *Sink* in Fig.13. The detail graph construction process can be defined as follows:

1. For each pixel p, create a set of vertexes $p_1,\ldots\ldots, p_{k-1}$ (k is the number of labels). Connect them by edges which is called *t-links* $\left\{ t_1^p,...,t_k^p \right\}$ where

$$t_1^p = \left\{ R, p_1 \right\}, t_j^p = \left\{ p_{j-1}, p_j \right\}, t_k^p = \left\{ p_{k-1}, S \right\};$$

2. For each pair of neighboring pixels p, q and for each $j \in \left\{ 1,...,k-1 \right\}$, create an edge called *n-links* $\left\{ p_j, q_j \right\}$ with weight $u_{\{p,q\}}$;

3. Each *t-links* t_j^p is assigned a weight $K_p + D_p(l_j)$ where K_p is any constant such that

$$K_p > (k-1) \sum_{q \in N_p} u_{\{p,q\}} .$$

Any cut of graph define in this way corresponds to a disparity configuration of stereo matching, and the cut cost is:

$$\left|C\right| = \sum_{p \in I} K_p + \sum_{p \in I} D_p(f_p^C) + \sum_{\{p,q\} \in N} u_{\{p,q\}} \left|f_p^C - f_q^C\right| \qquad (13)$$

In Eq.(13), $\sum_{p \in I} K_p$ is a constant, and $\sum_{p \in I} D_p(f_p^C)$ is corresponding data term in Eq.(12). If we define $V_{p,q}(f_p, f_q) = u_{\{p,q\}} \left|f_p^C - f_q^C\right|$, then there is only a constant term dispersion between Eq.(12) and Eq.(13). Therefore, the two equations can get optimal solution simultaneously, i.e. corresponding solution of min cut of constructed graph is also the global optimal solution of energy function. More information can refer to paper [21].

5.4 Stereo matching based on region boundary restriction and GC optimization

For the unconvexness of $u_{\{p,q\}} \left|f_p^C - f_q^C\right|$, energy function with the form of Eq.(13) can obtain global optimal solution via GC, therefore stereo matching algorithm based on region boundary restriction and GC optimization is naturally to represent energy function in the form similar to Eq.(13), but additional, it also need to deal with over-smooth problem meanwhile embody ordering and unique restrictions.

5.4.1 Assumptions
This algorithm assumes that:
1. Most of disparity jump occur at the region of color discontinuity.
2. If the image is over-segmentation based on color information, then it is sensible to consider that pixels in same segment have equal disparity [29,31].
3. Most occlusions in image pair are part-occlusion.
With qualitative analysis, it is clear that the assumptions given above are rational in most of cases.

5.4.2 Algorithm principle
Because pixels in same region have equal disparity after image segmentation, therefore, disparity of whole segment is up to disparity of its boundary pixels, and the key problem of stereo matching can be regarded as find corresponding disparity for every pixel on each segment.

Hence, an approach is region matching, for that if region correspondence between images is known, it is easy to figure out disparities of boundary pixels. [37] proposed a region matching algorithm, which worked as following: (1)segment image based on color information; (2) match regions by region features; (3) compute disparity map of whole image according to region matching result. However, this algorithm is not consistent with the fact. For factors such as imaging process and segmentation algorithm itself, segmentation results of image pairs of same scene may be different in many ways, which will be certainly result in wrong region matching. Fig.15 illustrates the difference in shape and size of segments of image pairs.
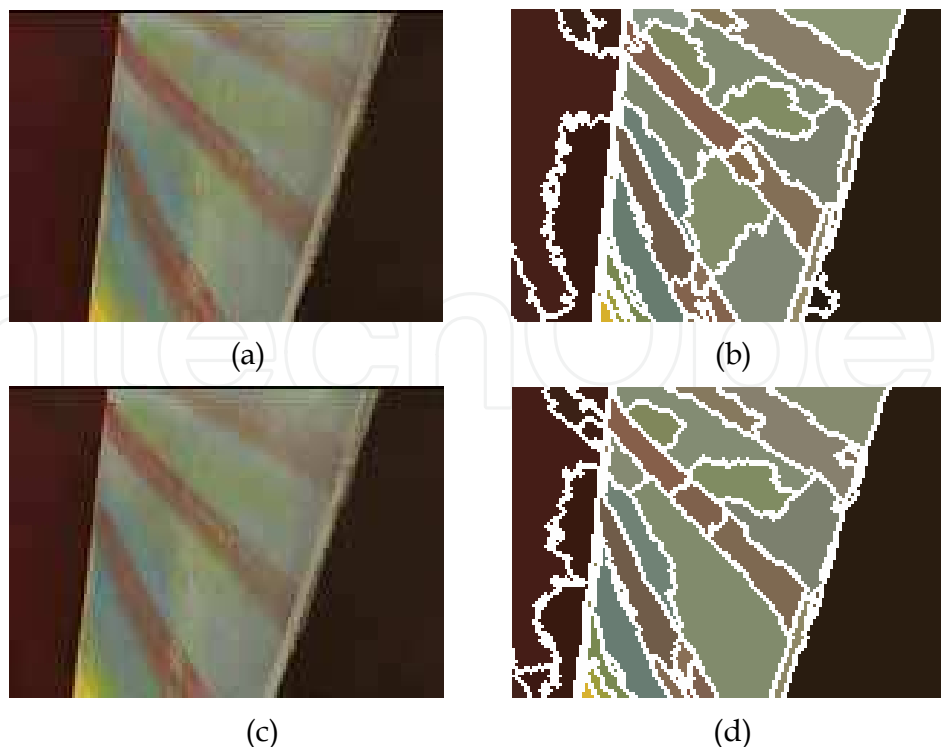
(a)

(b)

(c)

(d)

Fig. 15. (a): middle-top part of left view of venus testing image pair; (b): segmentation result of (a) via mean-shift algorithm; (c): middle-top part of right view of venus testing image pair; (d): segmentation result of (c) via mean-shift algorithm

To overcome problem mentioned above, in this section, a new approach for region matching is proposed. It only segments one of reference images (for generality, we can assume it is left image) based on color information, and then constructs energy function according to color-consistency and neighborhood relationship. It will show that, when optimization via graph cuts with this approach, region boundary of left image will close to corresponding region boundary of right image automatically. Constructed energy function is defined as:

$$min \ E(f) = \sum_{p \in BI} D_p(f_p) + \sum_{\{p,q\} \in N_1} V1_{p,q}(f_p, f_q) + \sum_{\{p,q\} \in N_2} V2_{p,q}(f_p, f_q) \qquad (14)$$

Where BI is the image consist of pixels on left and right region boundary (see [37] for definition of left and right region boundary) from one reference image, $D_p(f_p)$ denotes color difference for pixel p with disparity f_p, $V1_{p,q}(f_p, f_q) = u1_{p,q} \times |f_p - f_q|$, $V2_{p,q}(f_p, f_q) = u2_{p,q} \times |f_p - f_q|$ indicate the punish for different disparity of p, q with neighborhood N_1, N_2, where N_1 is neighborhood of pixels along scan line direction on adjacent region(as p, q illustrated in Fig.16(a)), N_2 is neighborhood of pixels on left and right boundary of the same region(as sillustrated in Fig.16(a)), $u1 > u2$. And Fig.16(b) shows how this algorithm works.

(a)   (b)

Fig. 16. (a): position relation illustration of pixels on region boundary(part magnified image of Fig.15(b)); (b): principle of region boundary pixels matching illustration(part magnified image of Fig.15(c))

p, q are pixels on boundary of two adjacent region, it is clear that, corresponding points of p, q in Fig.16(a) should lie in the area near central point of the circle in Fig.16(b). If assume correspondence of p, q in optimal configuration f^c is not the case, i.e. corresponding points of p, q in Fig.16(a) are p, q in Fig.16(b) respectively, then according to Eq.(14) , $V1_{p,q}(f_p, f_q) > 0$ and is proportional to distance between p, q, hence it is contradictory to the assumption that f^c is the optimal configuration. Because if choose p, q in Fig.16(b) more close to central point of the circle, $V1_{p,q}(f_p, f_q)$ will certainly reduce and $V2_{p,q}(f_p, f_q)$ and $D_p(f_p)$ should keep the same cost (for pixels in same region are color consistency and $u1 > u2$), therefore, the total cost of energy function $E(f)$ will reduce.

With globally optimizing Eq.(14) via GC, we can get fairly correct result of region matching, but for occlusion between regions, the dense disparity map can not be obtained yet.

5.4.3 Dealing with occlusion

Different depth of scene objects is the essential cause of occlusion, and only pixels in unoccluded part of one image have corresponding point in another image, that is to say, if we can match unoccluded part of reference images correctly, we can get correct disparity map meanwhile. After optimizing Eq.(14) via GC, region matching is fairly accurate, thereby the subsequent key problem is dealing with occlusion. We discuss that in three cases (because there is only horizontal displacement, so we need only analyze occlusion occurring at left and right region boundary):

1. There is no occlusion existing at left or right region boundary, then disparity of pixels on left and right region boundary are equal to each other, and also to disparity of the region.
2. Only left/right region boundary is occluded, in this case, disparity of whole region can be inferred from that of unoccluded region boundary.
3. Both left and right region boundary are occluded. Under this circumstance, we can not certainly get correct region disparity because neither disparity of left or right region boundary is authentic.

Analyze occlusion and disparity estimating in the cases of situation 1 and 2. As Fig.17 illustrated, Fig.17(a) and Fig.17(b) are stereo image pairs. Assume corresponding point pairs after region matching are $\{a1, A1\},...,\{c2, C2\}$, then disparity of pixels on region boundary in Fig.17(a) can be expressed as $(x_{A1} - x_{a1}),....,(x_{C1} - x_{c1})$. It is obvious that there is part-

occlusion existing in region A, C (labeled with red rectangle), therefore, only disparity of left region boundary of A is authentic, i.e. disparities of pixels between $a1, a2$ can be considered to be $(x_{A1} - x_{a1})$. Similarly, disparities of pixels between $c1, c2$ can be considered to be $(x_{C2} - x_{c2})$. For $(x_{A2} - x_{a2}) > (x_{A1} - x_{a1}), (x_{C1} - x_{c1}) > (x_{C2} - x_{c2}), (x_{B1} - x_{b1}) = (x_{B2} - x_{b2})$, we can present disparity estimating equation as:

$$D_p = \min((x_{P1} - x_{p1}), (x_{P2} - x_{p2})) \tag{15}$$

where P, p are corresponding pixels on left and right region boundary. As an example ,Fig.17(c) gives the corresponding dense disparity map of Fig.17(a) (disparity of background pixel is set to zero).

(a)   (b)   (c)
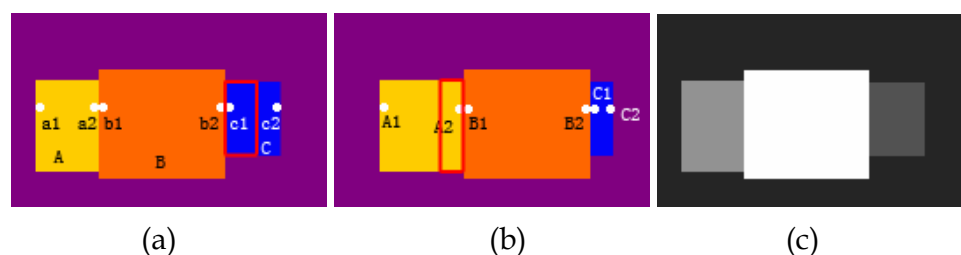
Fig. 17. Occlusion and disparity calculation. (a): left view of image pair; (b): right view of image pair; (c): corresponding disparity result

As discussed above, with this approach, it can obtain authentic region disparity and meanwhile occlusions occurring within reference image pairs. Additional, although there is punish given for adjacent region boundary, it allows disparity jump between them (as disparity jump existing at the left and right boundary of region B in fiugre17).

5.4.4 Uniqueness
In this algorithm, though it is not defined uniqueness restriction explicitly, uniqueness restriction is still fulfilled in result disparity map for color consistency and neighborhood interactions have been enforced in energy function.

As illustrated in Fig.17, assume boundary pixel $a2$ in Fig.17(a) is (x, y), its corresponding point in Fig.17(b) is $A2 = (x + d, y)$, and $b1 = (x + 1, y)$ is the adjacent boundary pixel of $a2$, hence coordinates of $B1$ as corresponding point to $b1$ should be $(x + d + 1, y)$ (when region B unoccluded) or $(x + d + k, y), k > 1$ (when region B occluded), otherwise, $B1$ will lie in region A and result in more cost of $D_p(f_p)$ for different color within two regions.

To sum up, the process of this algorithm can be described as following:
1. Segment arbitrary one image of stereo image pairs and obtain corresponding region boundaries.
2. Construct graph according to Eq.(14) and segmentation result.
3. Find the min cut of constructed graph and matching regions.
4. Estimate disparity map and analyze occlusion according to Eq.(15).

Among those steps, the second one is the key operation. Although it is similar to approach proposed in [21], it differs from that in graph element and neighborhood system, and the essence of energy function despite of alike equation form.

5.5 Stereo matching experiments

Fig.18 illustrates some examples of stereo matching and disparity estimating with image pairs from Middlebury dataset and real scenes. The results of stereo matching are measured with criteria proposed in [34], i.e. record number of pixel whose disparity discrepancy is beyond one from ground truth(called error). For every pair of reference image, calculate three statistics: (1) error on all pixels in image D_{all} ; (2) error on pixels in undiscontinuity region; (3) error on pixels in untexture region. Experiments result are shown in Fig.18, Fig.19, Fig.20.

(a)

(b)

(c)

(d)
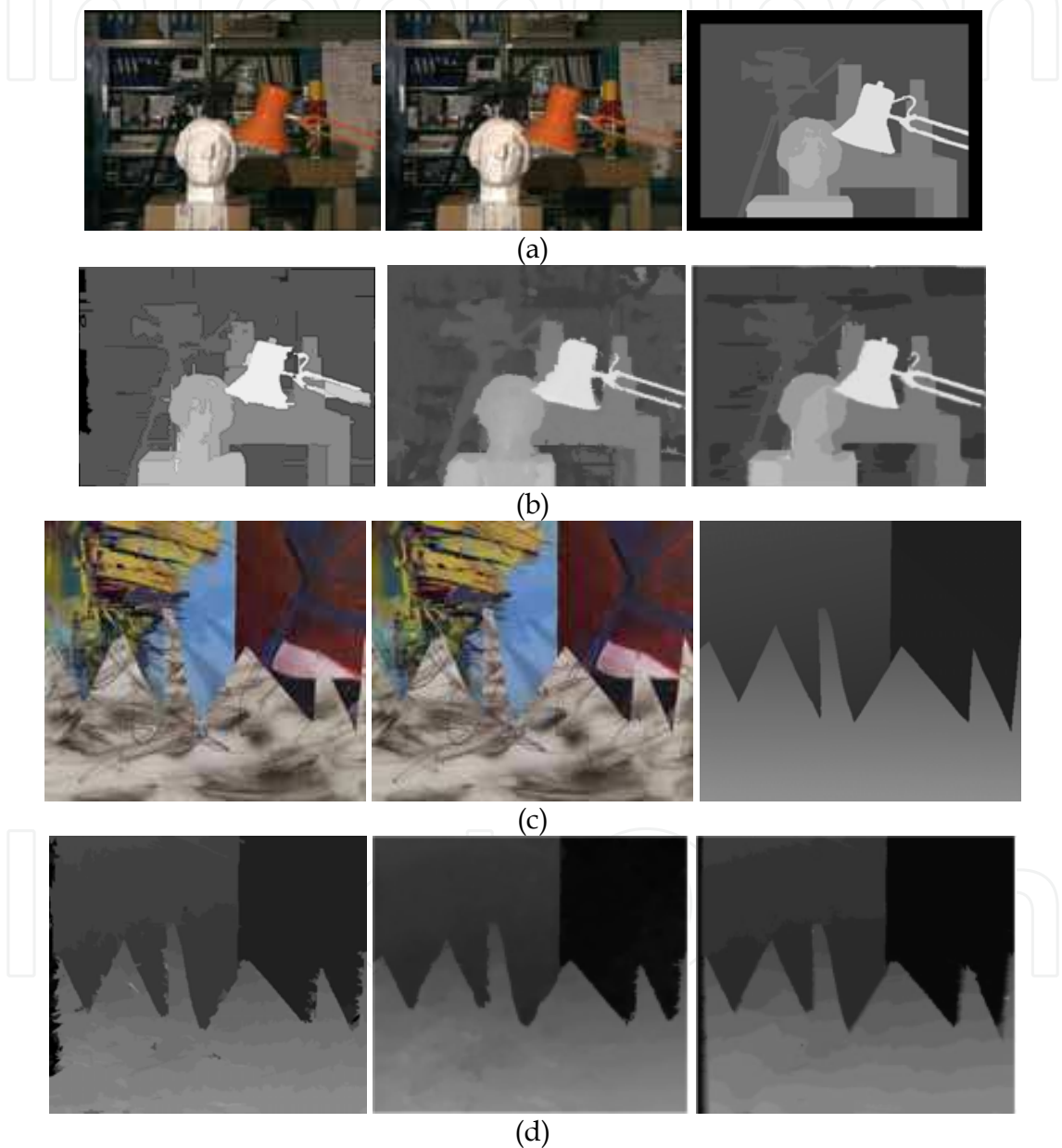
Fig. 18. Result on tsukuba and sawtooth testing image pair. (a): tsukuba testing image pair and corresponding ground truth; (b): corresponding stereo matching results on tsukuba testing image pair via algorithms of proposed, [35-36] from left to right; (c): sawtooth testing image pair and corresponding ground truth; (d): corresponding stereo matching results on sawtooth testing image pair via algorithms of proposed, [35-36] from left to right

(a)

(b)
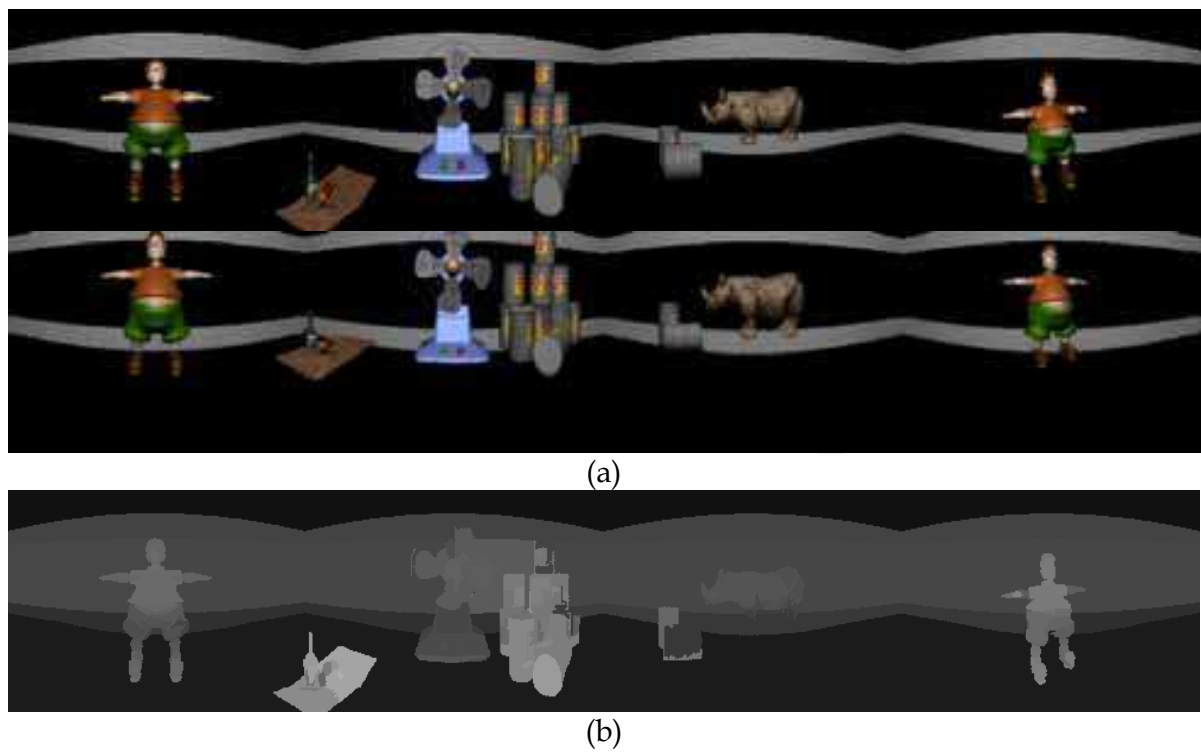
Fig. 19. Experiment with synthesized scene. (a) synthesized cylindrical panoramic image pair after rectification; (b) disparity map of synthesized cylindrical panoramic image pair.

(a)

(b)
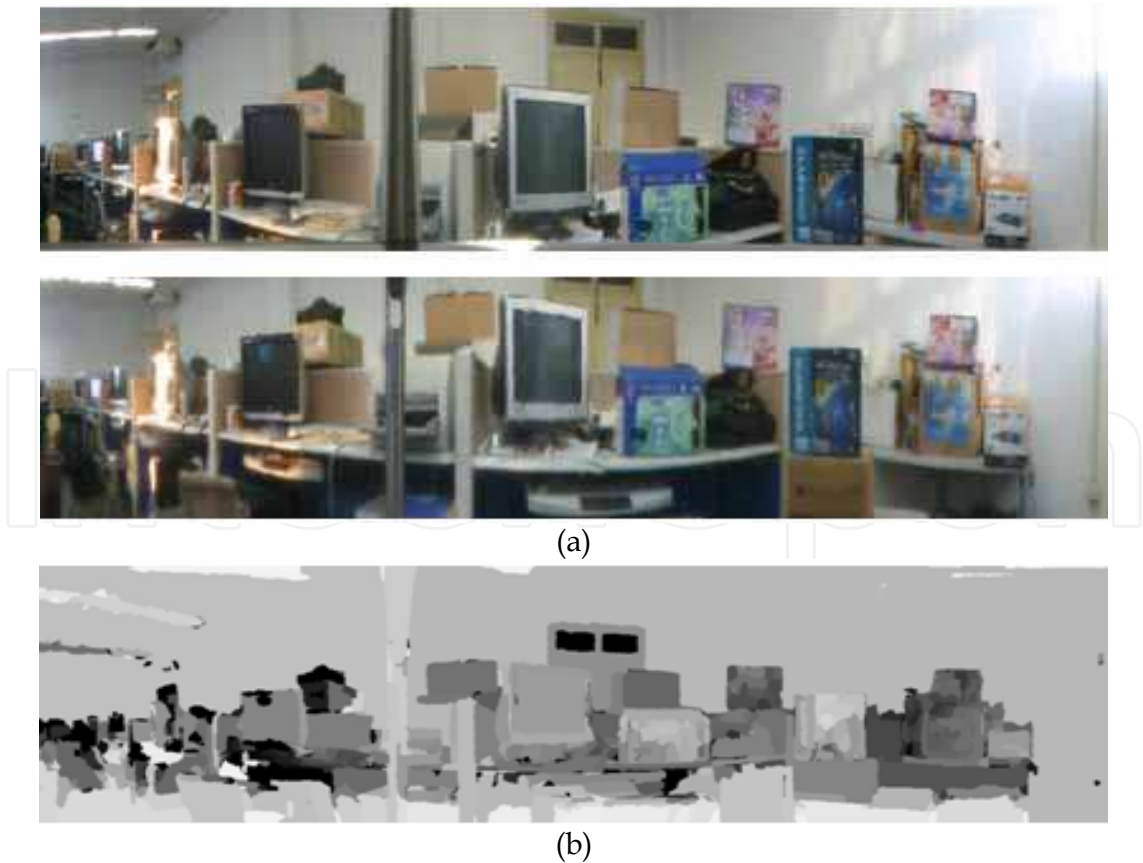
Fig. 20. Experiment with real scene. (a) real cylindrical panoramic image pair after rectification; (b) disparity map of real cylindrical panoramic image pair.

6. Applications on moving object detection and tracking

Moving object detection is a foundation problem in computer vision, and it is also a foundation problem in catadioptric omni-directional stereo vision. So, we will present an application case of moving object detection that is based on the omni-directional stereo vision.

As we know, traditional moving object detection methods that are based on a single camera have some difficulties, including: 1) Need to consider the background change, such as the illumination changes, background disturb (e.g. wind blowing and tree swing), shadow, movement of the cameras, et al. 2) When tracking the moving objects, the object is likely to go beyond the camera resulting in lose of the objects. At this time, we need to rotate the camera and search the object again.

The above problems can be resolved using the omni-directional stereo vision method. In this method, we utilize the depth information from the stereo vision to detect objects, since the depth information calculation is independent from the background image, it is not affected by illumination change and background disturb. Catadioptric omni-directional imaging system captures the 360 degree FOV just in one shot, so it does not need to rotate the camera when tracking moving objects.
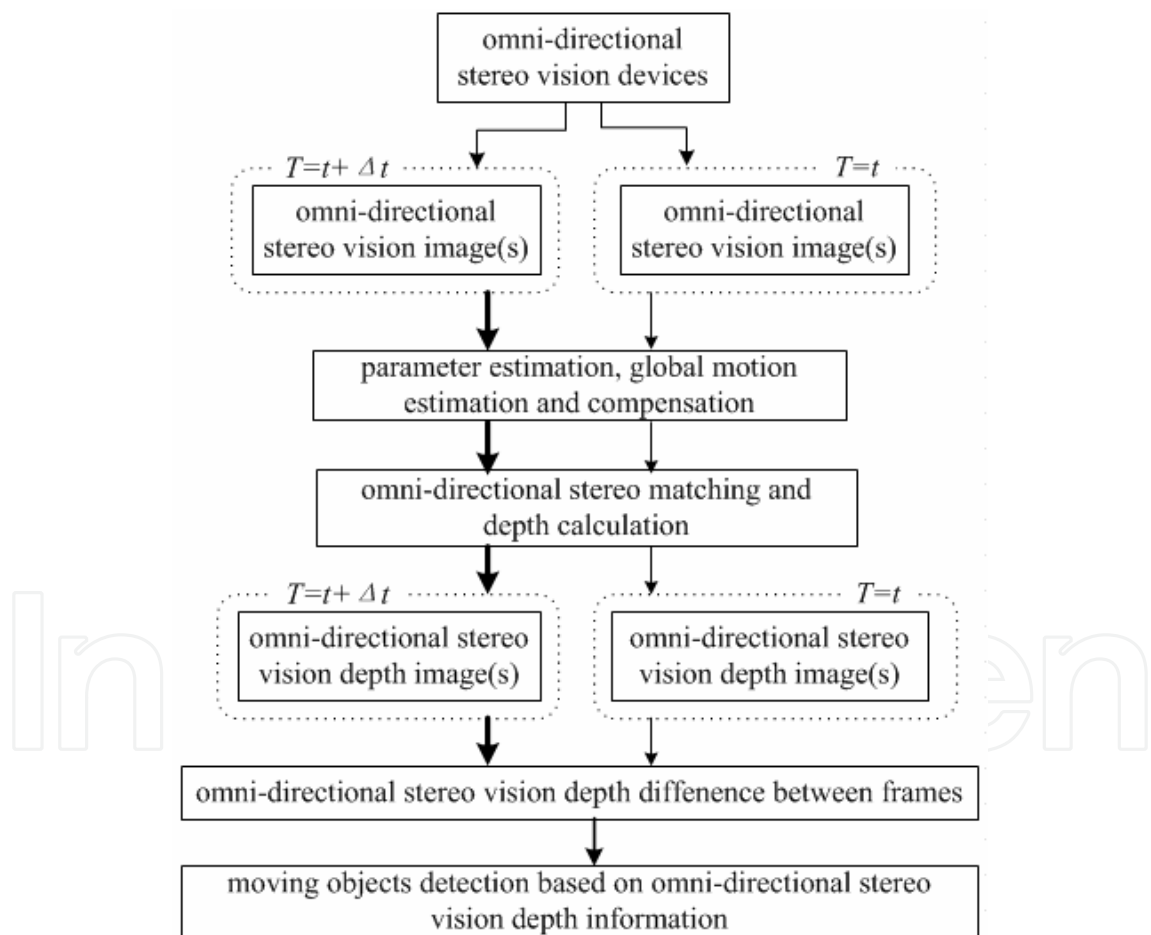

Fig. 21. Overall flow of moving object detection based on omni-directional stereo vision

Fig.21 shows the overall flow of moving object detection based on omni-directional stereo vision. In this figure, the omni-directional stereo vision device takes charge to capture omni-directional stereo vision image (or image pair). After camera parameters estimation, global movement estimation and global movement compensation, we get the omni-directional

stereo image (or image pair) that has compensated the global movement of the camera. At this time, there exists no camera movement, and exists only object movement (i.e. local movement). Based on these images, we can perform stereo matching and depth calculation, so as to get the omni-directional stereo depth image at the same time (e.g. $T = t + \Delta t$). Then, we can do differential calculation between frames of the omni-directional stereo depth images at different times, and we can detect and track the moving objects.

7. Acknowledgements

8. References

A.A. Argyros, C. Bekris, S.C. Orphanoudakis and L.E. Kavraki. Robot homing by exploiting panoramic vision, Journal of Autonomous Robots, 2005, 19(1), 7-25.

H.Y. Shum and L.W. He. Rendering with concentric mosaics, Proceedings of ACM SIGGRAPH, 1999, pp.299-306.

T. Gandhi and M.M. Trivedi. Parametric ego-motion estimation for vehicle surrounds analysis using an omni-directional camera, Machine Vision and Applications, 2005, 16(2), 85-95.

C. Sun and S. Peleg. Fast panoramic stereo matching using cylindrical maximum surfaces, IEEE Transactions on Systems, Man, and Cybernetics, Part B, 2004, 34(2), 760-765.

M.J. Zhang, W. Xu and W. Wang. Real VR space construction and its devices for curve face reflective imaging, P.R. China Patent application No. 200510031552.X, 2005, 5.

R. Benosman, S.B. Kang. Panoramic vision: sensors, theory, and applications. Springer-Verlag New York, Publishing House, 2001.

Zhihui Xiong, Maojun Zhang, Yunli Wang, et al. Fast panorama unrolling of catadioptric omni-directional images for cooperative robot vision systems. Proceedings of IEEE International Conference on Computer Supported Cooperative Work in Design (CSCWD), Melbourne, Australia, Apr. 26-28, 2007(2):1100-1104.

L. Smadja, R. Benosman, J. Devars. Determining epipolar constraint on cylindrical images and using it for 3d reconstruction, Proc. ICAR, Aug 2001.

R. Bunschoten et al.. Range estimation from a pair of omnidirectional images, Proc. IEEE Int. Conf. on Robotics and Automation, Seoul, Korea, 2001:1174~1179.

R. Bunschoten, B. Krose. 3-D scene reconstruction from cylindrical panoramic images, Robotics and Autonomous Systems (special issue), 2002, 41(2/3):111~118.

M. Kimura and H. Saito. 3D reconstruction based on epipolar geometry, IEEE Transactions on Information and Systems, 2001.

L. McMillan and G. Bishop. Plenoptic modeling: An image-based rendering system. ACM SIGGRAPH. 1995: 39~46.

H. Koyasu, J. Miura and Y. Shirai. Real omni-directional stereo for obstacle detection and tracking in dynamic environments. IEEE proceedings, International Conference on Intelligent Robots and System, 2001:31~36.

J. Gluckman, S. K. Nayar and K. Thorek. Real-time omni-directional and panoramic stereo. DARPA Image Understanding Workshop, California, 1998.

A. R. Lawrence and D. C. Nathan. Method and system for panoramic image morphing. United States Patent, 6795090, 2004.

D. Oram. Rectification for any epipolar geometry. 12th British Machine Vision Conference (BMVC), 2001.

S. M. Roy and J. Cox. Cylindrical rectification to minimize epipolar distortion. Computer Vision and Pattern Recognition Proceedings, 1997.

H. H. P. Wu, Y. H. Yu, and W. C. Chen. Projective rectification based on relative modification and size extension for stereo image pairs. IEEE Proceedings, Vision Image and Signal Processing, 2005.

Y. Boykov, O. Veksler and R. Zabih. Markov random fields with efficient approximations. In IEEE Conference on PAMI, 1998, 648-655.

H. Ishikawa and D. Geiger. Segmentation by grouping junctions. In IEEE Conference on PAMI, 1998, 125-131.

O. Veksler. Efficient graph-based energy minimization methods in computer vision. Ph.D. Thesis, Cornell University, 1999

V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts, In IEEE Conference on PAMI 2004, 26(2):147-159

Y. Boykov, O. Veksler and R. Zabih. Fast approximate energy minimization via graph cuts. In IEEE transaction on PAMI, 2001, 23(11):1222-1239.

V. Kolmogorov and R. Zabih. Multi-camera scene reconstruction via graph cuts. In Proc. ECCV, 2002, 82-96.

J. Kim, V. Kolmogorov and R. Zabih. Visual correspondence using energy minimization and mutual information. In Proc. ICCV, 2003.

Y. Boykov, V. Kolmogorov. An experimental comparison of min-cut max-flow algorithms for energy minimization in vision, In IEEE transactions on PAMI, 2004, 26(9):1124-1137.

V. Kolmogorov. Graph based algorithms for scene reconstruction from two or more views. Ph.D. Thesis, Cornell University, 2004.

V. Kolmogorov, R. Zabih. Computing visual correspondence with occlusions using graph cuts. In ICCV, 2001, 2:508-515.

L. Hong, G. Chen. Segment-based stereo matching using graph cuts. In IEEE Computer Society Conference on CVPR, 2004, 74-81.

M. Bleyer, M. Gelauz. A layered stereo matching algorithm using image segmentation and global visibility constraints, ISPRS Journal of Photogrammetry & Remote Sensing, 2005,128-150.

M. Bleyer, M. Gelauz. Graph-cut-based stereo matching using image segmentation with symmetrical treatment of occlusions, Signal Processing: Image Communication, 2007, 127-143.

Y. Deng, Q. Yang, X. Y. Lin and X. O. Tang. Stereo correspondence with occlusion handling in a symmetric patch-based graph-cuts model stereo. In IEEE transaction on PAMI, 2007, 1068-1079.

H. Ishikawa. Exact optimization for Markov random fields with convex priors. In IEEE Transactions on PAMI, 2003, 1333-1336.

D. Scharstein, R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, International Journal of Computer Vision, 2002, 47:7-42.

P. Mordohai, G. Medioni. Stereo using monocular cues within the tensor voting framework. In IEEE transaction on PAMI, 2006, 28(6):968-982.

K. J. Yoon, I. S. Kweon. Adaptive support-weight approach for correspondence search. In IEEE transaction on PAMI,2006, 28(4):650-656.

M. Ansari, L. Masmoudi, A. Bensrhair. A new regions matching for color stereo images. Pattern Recognition Letters, 2007, 1679-1687.

Computer Vision

Edited by Xiong Zhihui

This book presents research trends on computer vision, especially on application of robotics, and on advanced approachs for computer vision (such as omnidirectional vision). Among them, research on RFID technology integrating stereo vision to localize an indoor mobile robot is included in this book. Besides, this book includes many research on omnidirectional vision, and the combination of omnidirectional vision with robotics. This book features representative work on the computer vision, and it puts more focus on robotics vision and omnidirectioal vision. The intended audience is anyone who wishes to become familiar with the latest research work on computer vision, especially its applications on robots. The contents of this book allow the reader to know more technical aspects and applications of computer vision. Researchers and instructors will benefit from this book.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Xiong Zhihui, Chen Wang and and Zhang Maojun (2008). Catadioptric Omni-directional Stereo Vision and Its Applications in Moving Objects Detection, Computer Vision, Xiong Zhihui (Ed.), ISBN: 978-953-7619-21-3, InTech, Available from: http://www.intechopen.com/books/computer_vision/catadioptric_omni-directional_stereo_vision_and_its_applications_in_moving_objects_detection

INTECH
open science | open minds