Toward automatic reconstruction of 3D environment with an active binocular head

Chung-Yi Lin,[a,c] Sheng-Wen Shih,[b] Yi-Ping Hung,[a] and Gregory Y. Tang[c]

[a]Institute of Information Science, Academia Sinica, Taipei 115, Taiwan.

[b]Department of Computer Science and Information Engineering, National Chi Nan University, Nantou 545, Taiwan.

[c]Department of Computer Science and Information Engineering, National Taiwan University, Taipei 106, Taiwan.

swshih@ncnu.edu.tw; hung@iis.sinica.edu.tw

July 1, 2001

Abstract

In this paper, we propose a new automatic approach for reconstructing 3-D environments using an active binocular head. To efficiently store and access the depth estimates, we propose to use an inverse polar octree which can transform both unbounded depth estimates and unbounded estimation errors into a bounded 3-D space with appropriate resolution. The depth estimates are computed by using the asymptotic Bayesian estimation method. Estimated depth values are then smoothed by using discontinuity-preserving Markov random fields. The path of the local motion required by the asymptotic Bayesian method is determined online automatically to reduce the ambiguity of stereo matching. Rules for checking the consistency between the new observation and the previous observations have been developed to properly update the inverse polar octree. Experimental results showed that the proposed approach is very promising for automatic generation of 3-D models which can be used for rendering a 3-D scene in a virtual reality system.

Keywords: Active Vision, Stereo Vision, 3-D Reconstruction, Asymptotic Bayesian Estimation, 3-D Data Integration, View Interpolation

1 Introduction

In the last few years, virtual reality (VR) has found many applications in different areas, such as education, business and entertainment. Because of the rapid growth of VR applications, automatic generation of 3-D models from images has attracted much attention recently. In order to reconstruct 3-D models from images many people chose to use the stereo vision techniques. However, it is well known that stereo correspondence is a very difficult problem. The results of 3-D reconstruction obtained by using automatic stereo matching algorithms (such as those described in [1]) are still not reliable enough for practical use. Successful 3-D reconstruction systems used at the present time still have to utilize structured light (e.g., laser light stripe) or to impose strong constraints (e.g., the continuous surface constraint) to simplify the stereo matching problem. Otherwise, human interaction had to be introduced to the reconstruction system for solving the general stereo correspondence problem, e.g., Debevec et al. [2] and Hung et al. [3]. About one decade ago, Alominos and Badyopadhyay showed that many computer vision problems which are ill-posed, nonlinear and unstable for a passive observer become well-posed, linear and stable for an active observer[4]. Their work is based on the assumption that the active vision system is well calibrated such that accurate camera parameters are always available with respect to any configuration of the active vision system. However, most of the existing active vision systems are not accurately calibrated because of the following two main reasons. The first one is that many vision task can be accomplished without having to reconstruct the 3-D model of a scene [5], and the second one is that existing calibration techniques developed for passive vision systems are not suitable for active vision systems. Nevertheless, since a calibrated active vision system has many advantages over an uncalibrated one, people are developing calibration techniques for active vision system according to their requirements. Reid and Beardsley has developed a method for aligning a pair of stereo cameras such that their optical axes are parallel [6]. Based on a simplified kinematics model, McLauchlan and Murray at Oxford used a variable state-dimension filter to recursively estimate the head/eye orientation relation and the vertical and horizontal effective focal lengths without having to use a special calibration object [7]. The KTH group developed a three-stage calibration methods, i.e., zoom lens calibration [8] [9], kinematic calibration [10] and head/eye calibration [11], for their active binocular head. We have also spent many years on developing a four-stage calibration method for our binocular head and have achieved very accurate calibration results[12]. Based on our well calibrated binocular head (referred to as the IIS head), we have the ability to try solving the 3-D reconstruction problem using the active vision paradigm. Reconstructing the 3-D environment map using an active vision system involves the following three main problems:

1. The where-to-look-next problem: The main advantage of an active vision system over a passive one is that some of the external and internal camera parameters of the active vision system can be adaptively controlled so that an assigned vision task can be accomplished in a more efficient and accurate way. In particular, when a very specific vision task, e.g., man-made object recognition [13], face recognition [14], mobile robot navigation [15] and visual pursuing [16], is assigned to an active vision system, the criteria for selecting the best next observation configuration for the system is well defined. On the other hand, if there is no specific goal set for the vision system, then the

where-to-look-next problem will become too subjective that no generally acknowledged criterion can be found. In this work, the given vision task is to reconstruct a textured 3-D environment map that is visually consistent with the observed images. When there is no prior knowledge about the environment, each view point is of equal importance because we need as many as possible images to detect and to remove visually-inconsistent 3-D data. Therefore, we can simply choose to use a uniform sampling strategy to acquire images at equally spaced view points. Notice that determining the next best observation configuration for reconstructing a global scene is different from determining the best configuration for reconstructing a small 3-D surface patch. Because the latter one has a more specific goal then the former.

2. Navigation problem: Once the next best view point has been determined, the active vision system should have the ability to safely navigate itself to the view point. However, since robot navigation has been extensively studied for decades (refer to [17]) and it is not the research focus of this work, we will assume that our active vision system is moving in an obstacle-free environment.

3. Integration of the estimated 3-D data problem: As the active vision system moves to a new view point and computes new 3-D estimations, we have to update the 3-D map so that it is consistent with the new observations. The main difficulty of data integration is that when the new observations are inconsistent with the old one, we have to determine which one is incorrect even though both the new and old data are estimated using the same method.

For the VR applications, the virtual 3-D scene can be rendered by computing the shading of an object given the position of light sources and surface properties of the object. In this way, the 3-D model of the object has to be very accurate because the computed shading appearance is very sensitive to 3-D noise. On the other hand, when using image-based techniques to render a 3-D scene with texture extracted from real images, the quality of the rendered image is more tolerable to inaccurate 3-D data. Therefore, with image-based rendering techniques, one does not have to reconstruct a very accurate 3-D model in order to obtain a photo-realistic VR scene [18] [19] [20]. Knowing that image-based rendering techniques do not require accurate 3-D reconstruction and that 3-D reconstruction is, in general, an ill-posed problem, we do not intend to reconstruct highly accurate 3-D model of the scene. Instead, our goal is to reconstruct an approximate 3-D model having some associated texture information such that this approximation model, together with the texture information, can be used to synthesize images which look similar to the real ones when observing from arbitrary viewpoints *within a pre-specified viewing area*. If the synthesized image looks different from the real image when observing from a new point of view, then our goal is to update the current scene model so that the model will be consistent with all the previous views the vision system has observed. Hopefully, the scene model will become more accurate as the vision system samples more viewpoints within the specified viewing area. The idea, which we have proposed independently in [19], of reconstructing a textured 3-D model that is visually consistent with the observed images is similar to the space carving technique proposed by Kutulakos and Seitz [20]. They call the reconstructed approximated 3-D model the *photo-consistent shape*. In the absence of *a priori* geometric information, reconstructing a photo-consistent shape of an object is more feasible than reconstructing its exact 3-D model. The main difference between our approach and the space carving

technique is that our method is based on stereo vision techniques for 3-D estimation, but the space carving technique does not directly compute the 3-D data; instead, they simply remove all non-photo-consistent surface voxels. The advantage of the space carving approach is that the computation algorithm is very simple and relatively robust. But it will need to use many images sampled densely in 3-D space in order to generate a visually compelling photo-consistent shape. On the other hand, if only handful of calibrated images are available, then we will have to use stereo vision techniques to reconstruct a 3-D model. However, since the computed stereo correspondences usually contains some mismatched pairs, it is then necessary introducing an error-correction mechanism into the system to update the 3-D data. In our method, the error-correction mechanism is first to detect visually-inconsistent 3-D data and then to re-estimate them. In our point of view, our method is kind of a combination of the surface estimation technique and the space carving technique. In particular, when trying to reconstruct a wide scene, the proposed method can provide a visually-consistent 3-D model with fewer images than that the space carving technique requires.

In addition to the space carving technique, many approaches based on active vision systems for automatic generation of 3-D models from images have been proposed in the past decades. Beß et al. used a calibrated active color camera for 3-D reconstruction [21]. 3-D data were estimated from monocular color image sequences by using a stereo technique which combines feature-based and correlation-based stereo matching method. Disparities obtained from feature-based stereo were used to guide correlation-based stereo matching. Maru et al. used a stereo rig which is mounted on a calibrated translation stage such that the orientation of the translation stage with respect to the camera coordinate systems is know; hence, rough depth estimates can be obtained from the detected optical flow and then can be used to guide stereo matching [22]. Grosso and Tistarelli used a binocular head for robot visual guidance [23]. Since the task of visual guidance does not require accurate 3-D model of the environment; therefore, calibration task can be simplified as there are only a few parameters need to be roughly estimated. In their work, a simple method for estimating some necessary camera parameters was proposed and methods for estimating the rough depth value of the environment as well as the time-to-impact were developed. Abbott and Ahuja have studied the 3-D reconstruction problem using an active stereo system [24]. To reconstruct the surface from stereo images, they argued that the tightly coupled use of focus, camera vergence and stereo disparity results in a more powerful and complete system for surface estimation than when those cues are used individually. In their approach, aperture setting, focus setting, vergence angle, depth estimates and the integration of depth estimates from different cues and different viewpoints can be determined by minimizing a criterion function with a smoothness constraint. In their experiments, object surfaces were covered with textured newspapers to make stereo matching and depth from focusing visual module provide reliable results. Ours in distinction from theirs is that we have a well-calibrated active binocular head, so that we can apply the active sensing paradigm to simplify the 3-D reconstruction problem, to achieve more precise stereo matching result and to integrate 3-D estimates from multiple viewpoints; hence, we are able to deal with the reconstruction problem of a more complicated environment. Also, instead of applying the active vision system to reconstruct a narrow scene, our goal is to reconstruct a wide scene. Marchand and Chaumette have developed an active vision system consisting of a monocular

camera mounting on the end effector of a six degrees of freedom Cartesian robot [25]. Their active vision system was accurately calibrated so that 3-D data can be computed using structure from controlled motion method. Unknown static scenes containing objects of simple shapes can be automatically explored and reconstructed using image acquired and processed at nearly video rate. The reconstruction results of simple scenes (several light-color objects plus a black background) were very accurate, but their method was developed for polyhedral objects and cylinders only. We do not impose such strong constraints on the scenes in our method; therefore, we do not intend to achieve comparable accuracy as their system.

In this paper, we propose a new approach to reconstructing a model of the 3-D environment automatically by using a well-calibrated active binocular head [12]. The reconstructed 3-D points and their gray level values are stored in a volumetric data structure, i.e., the inverse polar octree, which will be described in section 2. An active control scheme has been used to minimize the ambiguity in stereo matching. The 3-D structure of the scene is estimated by using the asymptotic Bayesian estimation method [26], which is similar to the multiple-baseline stereo method [27] except that the asymptotic Bayesian estimation processes only one image at a time and it also provides the uncertainty information of the depth estimates. Details of the reconstruction process is described in section 3. Some experimental results on reconstructing the 3-D model of complex scenes are presented in section 4. Conclusions are given in section 5.

2 Inverse Polar Octree

In this work, we chose to use the voxel-based data structure for storing the reconstructed 3-D data because the overhead of integrating noisy 3-D estimations with voxel-based data structure is smaller than that of using mesh-based data structure. When the 3-D reconstruction process is completed, the voxel-based 3-D data can be transformed into mesh-based representation for fast rendering. To use the voxel-based 3-D representation, we have to solve the problem of packing the 3-D information contained in the infinite 3-D space into the finite memory space in a computer. To deal with this problem, we found that the 3-D measurement error of a stereo vision system is proportional to the distance between the object and the stereo cameras[28]. This fact suggests that uniform quantization of the 3-D data obtained by the stereo vision system is inefficient. A better quantization scheme is to have the resolution of the volumetric representation inversely proportional to the object distance. However, non-uniform quantization will result in complicated octree representations. Our solution to this problem is to take an *inverse polar transformation* of the estimated 3-D data before quantizing them into voxels. The inverse polar transformation is described in the following:

1. Transform the 3-D Cartesian coordinates, $(x,\ y,\ z)$, of a point P_{3D} to spherical coordinates, $(\rho,\ \theta,\ \phi)$, where ρ is the distance from the origin to point P_{3D}, and θ and ϕ are the angles specifying the direction of a unit vector pointing from origin to point P_{3D}.
2. For the 3-D spherical coordinates, $(\rho,\ \theta,\ \phi)$, compute its inverse polar coordinates, $(\frac{1}{\rho},\ \theta,\ \phi)$.

There are two major advantages of taking the inverse polar transformation. The first one is that, after the transformation, all the surrounding 3-D objects farther than a minimum distance to the observer,

say R_{min}, will be enclosed within a sphere with radius $\frac{1}{R_{min}}$. In this way, the infinite 3-D world outside a sphere is now mapped into a finite sphere, as shown in Fig. 1. The second advantage is that, after taking the inverse polar transformation, we can apply the uniform quantization scheme because the estimation error is now bounded, which is explained below in more detail.

Let ρ be the distance of an object point away from the observer. Since the 3-D estimation error is proportional to the object distance, the 3-D estimation error of the object point is approximately $k\,\rho$, where k is a constant determined by the configuration of the stereo cameras. Therefore, the estimate of the object distance is approximately $(1+k)\rho$. Notice that the estimation error, $k\,\rho$, is unbounded because the estimation error will approach infinite as ρ approaches infinite. However, after the inverse polar transformation, the object distance is now mapped to $\frac{1}{\rho(1+k)}$. Since the 3-D estimation error is, in general, much smaller than ρ, i.e., $k \ll 1$, we have

$$\frac{1}{\rho(1+k)} \approx \frac{1}{\rho} - \frac{k}{\rho}. \tag{1}$$

Here, the second term of the right hand side of (1) is the transformed estimation error, which is now bounded by $\frac{k}{R_{min}}$, if $\rho > R_{min}$. If we choose the quantization unit to be $\frac{k}{R_{min}}$, then the estimation error will be less than the quantization error for all the object points outside the sphere of radius R_{min}. It is advantageous to have a quantization error larger than the estimation error when using volumetric representation in 3-D reconstruction, because if the estimation error is larger than the quantization unit, then there will be many undesired false voxels that are caused by the estimation noise and are located around the real object position. As a result, when the quantization unit is smaller than $\frac{k}{R_{min}}$, we will not only have to use much larger amount of memory to store the 3-D data but also will obtain sparser scattering of 3-D measurement data. Sparsely scattered 3-D data will make the data integration more difficult, thus it is unwanted.

After taking the inverse polar transformation, 3-D data are stored in an octree according to the three coordinate components, $\frac{1}{\rho}$, θ, and ϕ. Let $\Delta\theta$ and $\Delta\phi$ be the angular resolution of the octree. The octree is created in the spherical coordinate system to maintain the uniform angular resolution, as shown in Fig. 2. Two 3-D points with 3-D coordinates $(\rho,\ \theta,\ \phi)$ and $(\rho,\ \theta+\Delta\theta,\ \phi+\Delta\phi)$ will be stored at $(\frac{1}{\rho},\ \theta,\ \phi)$ and $(\frac{1}{\rho},\ \theta+\Delta\theta,\ \phi+\Delta\phi)$, respectively. Obviously, the angular resolution of the octree remains unchanged after the inverse polar transformation and the uniform quantization into an octree.

2.1 Some Practical Issues of Using the Inverse Polar Octree

Notice that the inverse polar transformation is introduced in this work to map an infinite set

$$\left\{p \in R^3, \|p\| \geq R_{min}\right\}$$

into a finite space

$$\left\{q \in R^3, \|q\| \leq \frac{1}{R_{min}}\right\}.$$

Thus we can quantize the finite space and store the reconstructed 3-D information in a limited memory space of a computer. However, introducing the inverse polar transformation also brings in some problems

because it makes the volumetric representation become viewer centered. For a viewer-centered representation, when the observer moves to a new position, the entire 3-D data would have to be transformed to a new center accordingly. Due to the quantization scheme that we have adopted (the inverse polar octree), 3-D data at different distances to the center contain different levels of quantization error. When the observation center jumps to a new point at a long distance faraway from the original center, the non-uniform quantization error may cause the transformation more time-consuming because a voxel in the original octree may be mapped to multiple voxels in the other one and vice versa. However, it is rare the case for a moving observer changing its view point in this way. In general, the observation center usually moves slowly, thus the levels of quantization error associated with the 3-D data also change slowly and can be approximately regarded as constants. In this case, only simple point-to-point transformations are required when performing the re-centering computation. But since the quantization error does not change evidently, an alternative approach is to use an IPO with a fixed center so that we can skip the re-centering computation. This approach is especially suitable when the pre-specified viewing area is small and the IPO is centered at the viewing area.

Notice that our goal is to reconstruct a textured 3-D model that is visually consistent with the images acquired in a pre-specified viewing area. Therefore, it is not guaranteed that images synthesized using the reconstructed model are still serviceable if the specified points of view are outside the original viewing area. When the synthesized image is not acceptable, we can simply select another center to reconstruct a new IPO. Consequently, we will obtain multiple IPOs for a widely spread 3-D scene which can keep he 3-D data reasonably accurate and reduce the amount of memory required to save an infinitely large 3-D space to a manageable size. The idea of representing 3-D environments using multiple IPOs is similar to the QuickTime VR technique of the Apple Inc.. As the QuickTime VR technique, the IPOs can be used to provide panoramic views of a 3-D scene. But they also can be used to synthesize images when the virtual camera moves around the centers of the IPOs, which can not be achieved by using the QucikTime VR technique.

3 Automatic 3-D Reconstruction

3.1 Visually-Inconsistent Regions

The schematic diagram of our active 3-D reconstruction process is shown in Fig. 3. We assume that the positions and orientations of the stereo cameras at any configuration of the binocular head are available for exploring and reconstructing the 3-D environment. In practice, this requirement can be achieved by using a well-calibrated active binocular head equipped with accurate position and orientation sensors, such as InterSense IS-900 CT, Fastrak, or Flock of Birds. That is, the parameters of the stereo cameras on the binocular head are known at any time instant, based on the kinematic model of the binocular head and the readings of the position and orientation sensors. Hence, we can adopt the asymptotic Bayesian estimation method which assumes the camera parameters are know for each camera position.

To reconstruct the 3-D environment more efficiently, we do not apply the asymptotic Bayesian estimation to an image region unless it is necessary, or more precisely, unless it is a *visually-inconsistent region*

(which is defined below). If a set of camera parameters are specified, we can synthesize images according to the current world model stored in the IPO by using ray tracing techniques (refer to Appendix A). Next, the synthesized image is subtracted from the observed one to obtain a difference image. The difference image is thresholded into binary image format and then filtered by using the morphological opening to remove noise. The active pixels in the filtered binary image can be grouped into regions indicating where the depth information are either incorrect or not available. The detected regions will be referred to as the VIRs (Visually-Inconsistent Regions) because the observed image is visually inconsistent with the synthesized image according to the IPO. At the very beginning, the IPO contains no valid data. Hence, the whole image is a VIR, and has to be processed to estimate 3-D depth as described in the next subsection. Once some 3-D depth estimates are stored in the IPO, only the VIRs have to be processed.

3.2 Depth Estimation

To estimate the 3-D depth, we first partition the VIRs into small blocks, and then assume that each block in the left image is projected from a 3-D planar patch having a constant depth. The depth estimation method we used in this work is mainly the asymptotic Bayesian estimation method proposed by Hung et al. [26], which is adapted for active vision purpose and is described briefly in the following. At first, a *reference image* is acquired by using the left camera at an initial pose. The reference image will also be used in integrating new 3-D observations into the IPO (refer to section 3.4); therefore, it is kept in the memory during the whole 3-D reconstruction process. Suppose that the depth, d, of block P in the VIRs of the reference image is to be estimated. Let the initial estimate of the reciprocal variance of d be $\Phi_1 = 0$, since we do not have any information about d yet. In order to obtain the correct stereo correspondence of P in the right image, we first move the left camera locally and incrementally to compute a rough estimate of the depth of P (the way we determine the path of the local motion will be described in the next subsection). Now, suppose that we have obtained a sequence of images all acquired by using the left camera, denoted by I_1, I_2, ..., where I_1 is the reference image of this image sequence and the other images are acquired during the local camera motion. Since the binocular head is well calibrated, we have the relative geometric relation (i.e., the relative camera position and orientation) of the image pair (I_1, I_n), where $n \geq 2$. Based on the geometric relation and a given depth estimate $\hat{d}$, we can compute, for each pixel in a block $P \in I_1$, the corresponding image point in the nth image. For convenience, let s be a 2-D image point in image block $P \in I_1$ and let $u_n(s, \hat{d})$ denote its corresponding image point in the new image i.e., (the nth image), as showed in Fig. 4. The depth of the image patch, d, can be refined by minimizing the following objective function:

$$J_n(d) = \frac{1}{2}(d - \hat{d}_{n-1})^t \Phi_{n-1}(d - \hat{d}_{n-1}) + \frac{1}{2}\sum_{s \in P}\left[I_n(u_n(s,d)) - I_1(s)\right]^2, \tag{2}$$

where $I_1(s)$ and $I_n(u_n(s,d))$ are the intensity value of pixel s in image 1 and the intensity value of pixel $u_n(s,d)$ in image n, respectively, and Φ_{n-1} denotes the reciprocal variance of the estimated depth $\hat{d}_{n-1}$ given images 1, 2, ..., $n-1$. The reciprocal variance can be updated by using the following equation:

$$\Phi_n = \Phi_{n-1} + \frac{\partial^2}{\partial d^2}\left\{\frac{1}{2}\sum_{s \in P}\left[I_n(u_n(s,d)) - I_1(s)\right]^2\right\}\Bigg|_{d=\hat{d_n}}. \tag{3}$$

The asymptotic Bayesian process for estimating the depth of an image patch P is summarized in the following.

1. Set $\Phi_1 = 0$, $n = 1$ and $d_1 = \infty$.
2. Acquire the reference image I_1 using the left camera.
3. Set $n = n + 1$.
4. Acquire the nth image using the left camera.
5. Compute $\hat{d_n}$ by minimizing the error function in (2) using a gradient descent method.
6. Update Φ_n with (3).
7. Discard image I_n.
8. Repeat steps 3–7 until the estimation error is satisfactorily small.

More details about the asymptotic Bayesian method can be found in [26].

The local motion and asymptotic Bayesian estimation method can be repeated until we obtain accurate enough depth estimates; however, recall that the goal of performing local motion is to get rough estimate of the depth value for determining stereo correspondences more accurately. Since the depth estimation error using local motion is proportional to the length of the effective movement, the local motion can be terminated when the length of the length of movement is greater than some value. According to our analysis result (refer to AppendixB), 50 millimeters of incremental local motion can reduce the depth uncertainty to a level such that the search region for stereo correspondence is less than 10 pixels in our setup, i.e.,

$$\left|u_n(s, \hat{d_n}) - u_n(s, d_{true})\right| \leq 5, \tag{4}$$

where d_{true} is the true value of the depth of block P. Therefore, once the effective movement length of the incremental local motion is greater than 50 millimeters, our system will use the image taken by the right camera as the new input image of the asymptotic Bayesian estimation process (i.e., a big jump) and perform an exhaustive search for the minima of (2) along the epipolar line in the ten-pixel search region centered at $u_{right}(s, \hat{d_n})$. Then, a gradient descent search is performed to further refine the depth estimate. After the depth estimates of all the patches in the visually-inconsistent regions are computed with the above process, Markov random fields can be used to smooth the depth map while preserving the depth discontinuity [29]. The smoothed 3-D data are integrated into the IPO using the method described in subsection 3.4. The active binocular head then moves to another new station and repeat the depth estimation procedure again for the VIRs until the size of all VIRs are small enough.

3.3 Path Planning for Local Motion

Having a well-calibrated active binocular head, we are able to control the cameras to move along a path which can reduce the ambiguity level of stereo matching. Our path planning method is based on the following observation: when performing stereo matching, the stereo correspondence can be determined more easily and reliably if the edge orientation is perpendicular to the epipolar line, as shown in Fig. 5. On the other hand, if the edge orientation is parallel to the epipolar line, then finding stereo correspondence

is an ill-posed problem. To eliminate the ambiguity in stereo matching, the local motion is selected to form epipolar lines which are perpendicular to most edges having highly uncertain depth estimates. The following procedure describes the way we determine the local motion:

1. Perform Sobel edge detection on the new input image and record the orientations θ_j of each edge pixel j.
2. For each edge pixel j, get its reciprocal variance value, Φ_j, determined in the asymptotic Bayesian process. Notice that a larger value of Φ_j indicates that the depth estimate of pixel j is more reliable because $\frac{1}{\Phi_j}$ is the variance of the depth estimate of pixel j.
3. Compute the average edge orientation weighted by its variance value as follows:

$$\Theta = \frac{\sum_{j:\Phi_j>0}(\frac{\theta_j}{\Phi_j})}{\sum_{j:\Phi_j>0}(\frac{1}{\Phi_j})}. \tag{5}$$

   Notice that in (5), edge orientations corresponding to depth estimates of higher uncertainty will be weighted more heavily.

4. Compute the horizontal and vertical motion components, H_{move} and V_{move}, of the camera:

$$H_{move} = \Delta_H \cos\left(\Theta + \frac{\pi}{2}\right), \tag{6}$$

   and

$$V_{move} = \Delta_V \sin\left(\Theta + \frac{\pi}{2}\right), \tag{7}$$

   where Δ_H and Δ_V are two predetermined constants specifying the step size of each movement.

3.4 Consistency Check for a new 3-D Observation

Suppose that we have ever moved the active binocular head to $K+1$ different stations, that we have collected $K+1$ reference images and the corresponding camera parameters at the $K+1$ stations, that the 3-D data observed at the first K stations have been integrated into the IPO, and that the 3-D data observed at the $(K+1)$st station is to be integrated into the IPO. Only those new 3-D data which are consistent with the old data were integrated into the IPO. Rules for checking the consistency is described below.

Suppose a new depth estimate, whose 3-D coordinates are $p_{3D}(K+1)$, is considered to be integrated into the IPO. Let $p_{2D}^{K+1}(K+1)$ be the image location of $p_{3D}(K+1)$ on the $K+1$st reference image, and $p_{2D}^{K+1}(k)$ be the projected 2-D image location of $p_{3D}(K+1)$ on the kth reference image. By back-projecting $p_{2D}^{K+1}(k)$ into a 3-D ray and compute the first intersection point of the 3-D ray and the occupied voxel in the IPO, we have $p_{3D}(k)$. We say that the new observation $p_{3D}(K+1)$ is *geometrically compatible* with the kth observation if $p_{3D}(k)$ is not occluded by $p_{3D}(K+1)$ when observed from the projection center of the kth reference image (see Fig. 6 for an example of geometrically incompatible 3-D data). If $p_{3D}(K+1)$ is geometrically incompatible with the kth reference image, then we further check if its color (or graylevel) is compatible with that of the kth observation, i.e.,

$$\left|I_{K+1}(p_{2D}^{K+1}(K+1)) - I_k(p_{2D}^{K+1}(k))\right| < \tau_C,$$

where τ_C is a given threshold value. If $p_{3D}(K+1)$ is either geometrically compatible or color-compatible with the kth reference image, then we say that $p_{3D}(K+1)$ is compatible with the kth reference image, since, in either case, adding the 3-D point, $p_{3D}(K+1)$, into the IPO will not cause visual inconsistency between reference image $K+1$ and k.

If more than two third of the reference images are compatible with the new observation, then $p_{3D}(K+1)$ is said to be *largely consistent* with the previous observations and is used to update the IPO. If the new observation is not largely consistent with the previous observations, it is discarded. After a new observation $p_{3D}(K+1)$ is determined to be largely consistent with the previous observations, we then remove those old 3-D voxel data which occlude $p_{3D}(K+1)$ on the projection center of the $K+1$st reference image.

Notice that, in the above-mentioned consistency check process, we did not use the uncertainty information, i.e., Φ_j^{-1}, a byproduct of the asymptotic Bayesian estimation method (refer to section 3.2), of the 3-D estimates. This is because that the uncertainty measurement is valid only when the 3-D estimation error is small. When the 3-D estimation error is small, it is unlikely to fail in the consistency check process. Therefore, we did not use the uncertainty information in the consistency check. As to the data integration, even if the estimated 3-D data is largely consistent with the previous observations, we still can not ensure that the 3-D data is likely to be very accurate, and hence, we also ignore the uncertainty information of the 3-D estimates in data integration process.

4 Experiments

In the experiments, a well-calibrated binocular head [12] as shown in Fig. 7 was used to acquire stereo image sequences. The binocular head is mounted on an X-Y table which is used for emulating a mobile robot platform. Two experiments were conducted to test the proposed active vision algorithm. In the first experiment, the target scene consisted of five objects. Four of them have textured surfaces, namely, a planar background, a cylinder, a rectangle pillar, and a box, whereas the other one is a texture-less half-sphere. The relative positions of the five objects and five viewpoints for 3D measurement chosen in advance are illustrated in Fig. 8. The reference images acquired at those five viewpoints by the left camera were shown in Fig. 9. At each viewpoint, a sequence of local movement were performed automatically to estimate the 3-D information corresponding to the VIRs by using the asymptotic Bayesian method. At the very beginning, the inverse polar octree contained no valid data. Hence, the whole image was visually-inconsistent, and had to be processed to estimate 3D depth. Fig. 10(a) shows a bird's eye view of the world model reconstructed at viewpoint A. In Fig. 10(a), we can find the contours of the rectangle pillar and part of the cylinder. When the active binocular head was driven to viewpoint B, an image was synthesized at viewpoint B according to the world model reconstructed at viewpoint A. The synthesized image is shown in Fig. 10(b), whereas the observed image at viewpoint B is shown in Fig. 10(c). The VIRs (marked with black pixels) computed based on Figs. 10(b)–(c) is shown in Fig. 10(d). Only the 3-D data of the VIRs needed to be estimated/re-estimated using the asymptotic Bayesian method. The estimation results were then used to update the world model. Fig. 10(e) shows

the synthesized image at viewpoint B rendered by using the updated world model. Fig. 11 shows a bird's eye view of the reconstructed world model after all the images observed at the five viewpoints A–E have been processed. Fig. 12 shows some intermediate and final results in the reconstruction process. Those images were synthesized at two virtual viewpoints F and G, where virtual viewpoint F is located between viewpoints A and B, and virtual viewpoint G is located at the right side of viewpoint C. Figs. 12(a)–(b) shows the images synthesized at virtual viewpoints F and G according to the world model reconstructed using only the observations made at viewpoint A, respectively. Figs. 12 (c)–(d) shows the images synthesized according to the world model reconstructed after processing all the images acquired at the five viewpoints, respectively. Notice that the reconstructed 3-D model of the scene became more complete as more and more images were processed.

In the second experiment, we show how a complex scene in our laboratory can be reconstructed with the active binocular head. First, twenty viewpoints for 3-D reconstruction were chosen in advance. The reference images acquired by the left camera at the twenty viewpoints are shown in Fig. 13. The relative position of the objects and the twenty viewpoints are shown in Fig. 14, where Object 1 is a bookshelf in the background, Object 2 is a textured cardboard and Object 3 is a box. At each of the twenty viewpoints, a sequence of local movements are performed to estimate the depth value of the VIRs by using the asymptotic Bayesian method. Fig. 15 shows typical images of synthesized and observed images, as well as the corresponding computed VIRs. Fig. 16 shows an image sequence (for illustrating the effect of our local motion planner) acquired along two paths of local motion, a purely horizontal path (shown in Fig. 16(a)–(b)) and a sequence of motion determined by the method described in section 3.3 (shown in Fig. 16(c)–(d)). Notice that, in Fig. 16(d), the reciprocal variance value increased from left to right as more and more images were acquired and processed. Also, the computed local motion drove the camera to move both vertically and horizontally to reduce the ambiguity of stereo matching. Since the vertical camera motion could not be generated by the X-Y translation table, we moved the tilt joint to generate an equivalent vertical camera motion, which was possible (although quite limited) because the lens center of the camera was located a distance off the rotation axis of the tilt joint.

The 3-D data obtained by using the images taken at viewpoints $A1$–$A4$ is shown in Fig. 17(a). Notice that when observing from viewpoints $A1$–$A4$, part of the bookshelf, i.e., region R^* marked in this figure, is occluded by Object 3. Fig. 17(c) shows an image synthesized at a virtual viewpoint which is located between viewpoints A and C, will contain several "holes" (black image regions) because the 3-D information of region R^* are still not valid. However, after all the images observed at the twenty viewpoints were used to update the IPO, the 3-D structure become more complete, as shown in Fig. 17(b), and the synthesized image based on the updated 3-D structure looks much better (most of the "holes" has been patched), as shown in Fig. 17(d). Notice that during the asymptotic Bayesian process, constant depth values were assumed for each square image block (refer to section 3.2). Next, the estimated depth values were interpolated and smoothed to obtain smooth 3-D surface, and the side effect of the interpolation process is that many undesired voxels between Objects 1, 2 and 3 are generated. However, these undesired voxels can be eliminated in the consistency check. As shown in Fig. 17(b), many undesired voxels originally found in Fig. 17(a) have been removed. Fig. 18 shows images

synthesized at a viewpoint which is located above the real viewpoints and overlooking the scene. This figure shows the reconstructed 3-D information becomes more complete and accurate as more images observed at different viewpoints were processed.

For testing the reconstruction results, we manually selected two test viewpoints—one was located between viewpoints $A4$ and $C2$ and the other was located between viewpoints $A3$ and $B4$—for acquiring test images. Figs. 19(a) and (c) show the acquired test images and Figs. 19(b) and (d) show the synthesized images by using the reconstructed 3-D environment stored in the IPO. Notice that Figs. 19(a) and (b) and Figs. 19(c) and (d) look very similar, which means that the reconstruction results is visually-consistent with the real image and thus can be used in some VR applications. The reconstructed data can also be converted to 3-D meshes for there are hardware graphic accelerators which can render texture-mapped 3-D meshes at video rate. Fig. 20 shows the 3-D meshes converted from the data stored in the IPO rendered at an overlooking viewpoint. Fig. 21 shows a sequence of images synthesized by using the reconstructed textured-mapped 3-D meshes.

5 Conclusion

We have presented a new approach to reconstruct the 3-D environment automatically with an active binocular head. Active vision has been advocated by many researchers such as Bajcsy, Aloimonos, and Ahuja about a decade ago. However, most active stereo vision system has been applied to object tracking and no much progress on 3-D reconstruction using active stereo has ever been made after Abbott and Ahuja's work mainly because calibrating an active binocular head is much more difficult than calibrating a fixed cameras. We have spent many years on calibrating our binocular head and have achieved very accurate calibration results[12]. Based on our well calibrated binocular head, we have developed an active stereo vision algorithm which can estimate the 3-D depth automatically, plan and maneuver a sequence of local movements to reduce the ambiguity in stereo matching, and integrate 3-D data obtained in different points of observation. Real experiments have been performed to verify the algorithm proposed in this paper. The experimental results shows that the proposed algorithm is promising.

Acknowledgments

This work was supported in part by National Science Council of Taiwan, ROC, under Grants NSC 86-2745-E-001-002.

A The Ray Tracing Method for Synthesizing an Image from the IPO

Let us choose the reference frame of the IPO as the WCS (World Coordinate System). Given a set of intrinsic and extrinsic camera parameters, we can back-project an image point, say P_{2D}, into a 3-D ray. The back-projected 3-D ray is then transformed into the WCS and the resulting line equation is given

by

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = P_0 + \lambda P_1, \tag{8}$$

where P_0 is the projection center of the camera measured in the WCS, P_1 is the direction of the 3-D ray, and λ is a positive number. Converting the Cartesian coordinates (x, y, z) into spherical coordinates, we have (ρ, θ, ϕ), where

$$\rho = \sqrt{\|P_0\|^2 + 2\lambda P_0^t P_1 + \lambda^2 \|P_1\|^2}. \tag{9}$$

From the above equation, we can derive a representation of the positive number λ as follows:

$$\lambda = \frac{-2P_0^t P_1 + \sqrt{(P_0^t P_1)^2 - 4\|P_1\|^2 \left(\|P_0\|^2 - \rho^2\right)}}{2\|P_1\|^2} \tag{10}$$

Substituting (10) into (8), the 3-D line is now parameterized by ρ, namely $(x(\rho), y(\rho), z(\rho))$. In order to search for a surface voxel by using the back-projected 3-D line, the 3-D line is mapped to a 3-D curve with parameter $\frac{1}{\rho}$ in the IPO. Given the the curve parameter $\frac{1}{\rho}$, we can compute the 3-D Cartesian coordinates of a point on the line $(x(\rho), y(\rho), z(\rho))$, from which we can determine θ and ϕ. If $\left(\frac{1}{\rho}, \theta, \rho\right)$ is the first voxel intersecting the back-projected 3-D line, then the color data contained in this voxel is used to fill the pixel, i.e., P_{2D}, on the output image. For each pixel of the output image, the ray tracing procedure is proceeded to synthesize the output image.

B Analysis of Effective Length of Local Motion Needed to Reduce the Depth Uncertainty

Unlike a passive camera whose orientation is fixed, an active camera is able to acquire images in any direction. Thus, if the rotation axis is roughly passing through the optical center of the active camera, then the planar image sensor can be regarded as a portion of a spherical image sensor as shown in Fig. 22(a). To analyze the effective length of local camera motion needed to reduce the depth uncertainty, we assume that the active camera has a spherical image sensor and the angular resolution of the camera is defined by $\frac{\delta u}{f}$, where δu is the pixel spacing of the active camera and f is its effective focal length. Suppose that the initial position of the active camera is located at L_1 and that effective length of the baseline is x when the active camera is moved to L_n. Let O be the point at the middle of the baseline and P be an object point located at a distance of R away from O as shown in Fig. 22(b). Since the acquired image is digitized with an angular resolution of $\frac{\delta u}{f}$, the reconstructed location of P, denoted by $\hat{R}$, could be anywhere within the shadow region in Fig. 22(b) if the feature extraction error is about one pixel. To reveal the relation between the uncertainty, δR, of the depth estimate, $\hat{R}$, and the length of the baseline, x, we first derive the relation between the depth value, R, and the angle between the baseline and the back-projected 3-D ray, $\theta = \angle PL_1O$, as follows:

$$R = \frac{x}{2}\tan(\theta). \tag{11}$$

The uncertainty of the depth estimate, δR, can then be obtained by differential approximation of the above equation, which yields

$$\delta R = \frac{\partial R}{\partial \theta}\delta\theta = \frac{x}{2}\sec^2(\theta)\delta\theta, \tag{12}$$

where $\delta\theta = \frac{\delta u}{f}$. Notice that, in Fig. 22(b), $\sec^2(\theta) = \frac{4R^2+x^2}{x^2}$; therefore, we have

$$\delta R = \frac{4R^2 + x^2}{2x}\delta\theta \tag{13}$$

which leads to

$$x = \frac{f\delta R}{\delta u} + \sqrt{\frac{f^2\delta R^2}{\delta u^2} - 4R^2}. \tag{14}$$

The above equation can be used to determine the effective length of the baseline (or the amount of movement of a monocular active camera), x, when we want to control the uncertainty of the depth estimates to be less than δR. In this work, we have chosen to use the depth estimate obtained from left monocular image sequences to predict the location of the stereo correspondences on the right image sequences (refer to Fig. 23); hence, δR is determined by the desired size of the searching window for stereo correspondences. Suppose that the width of the searching window for stereo correspondences along the epipolar line is W and that the baseline of the binocular head is B. Then, from (13), we have

$$\delta R = \frac{4R^2 + B^2}{2B}\delta\theta_W \tag{15}$$

where $\delta\theta_W = W\frac{\delta u}{f}$ is the angle spanned by the searching window. Equations (14) and (15) are used together to determine the required amount of movement of the active cameras for 3-D reconstruction.

Notice that in deriving (14), we did not consider the effect of subpixel feature extraction nor the use of adaptive filtering techniques such as the Kalman filter and the asymptotic Bayesian estimator, which all can further reduce the uncertainties of the depth estimates to some extent. Therefore, if recursive filters or accurate feature extraction techniques are applied in 3-D reconstruction, then the uncertainties in the depth estimates will be smaller than the designated uncertainties, δR.

References

[1] U. R. Dhond and J. Aggarwal, "Structure from stereo—a review," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, no. 6, pp. 1489–1510, 1989.

[2] P. E. Debevec, C. J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach," in *Proceedings of ACM SIGGRAPH'96*, pp. 11–20, 1996.

[3] Y.-P. Hung, K.-C. Hung, C.-S. Chen, and C.-S. Fuh, "Multi-pass hierarchical stereo matching for generation of digital terrain models from aerial images," *Machine Vision and Applications*, vol. 11, no. 1, pp. 280–291, 1998.

[4] J. Aloimonos and A. Badyopadhyay, "Active vision," in *Proceesings of the First International Conference on Computer Vision*, pp. 35–54, 1987.

[5] J. Aloimonos, "Purposive and qualitative active vision," in *Proceedings of the International conference on Pattern Recognition*, vol. 1, pp. 346–360, 1990.

[6] I. D. Reid and P. A. Beardsley, "Self-alignment of a binocular robot," *Image and Vision Computing*, vol. 14, pp. 635–640, 1996.

[7] P. F. McLauchlan and D. Murray, "Active camera calibration for a head-eye platform using the variable state-dimension filter," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 15–22, 1996.

[8] M. Li, "Camera calibration of the kth head/eye system," Tech. Rep. CVAP147, Computational Vision and Active Perception Laboratory, Department of Numerical Analysis and Computing Science, Royal Institute of Technology (KTH), S-100 44, Stockman, Sweden, 1994.

[9] M. Li and J. M. Lavest, "Some aspects of zoom-lens camera calibration," Tech. Rep. CVAP172, Computational Vision and Active Perception Laboratory, Department of Numerical Analysis and Computing Science, Royal Institute of Technology (KTH), S-100 44, Stockman, Sweden, 1995.

[10] M. Li, D. Betsis, and J. M. Lavest, "Kinematic calibration of the kth head-eye system," Tech. Rep. CVAP171, Computational Vision and Active Perception Laboratory, Department of Numerical Analysis and Computing Science, Royal Institute of Technology (KTH), S-100 44, Stockman, Sweden, 1994.

[11] M. Li and D. Betsis, "Head-eye calibration," in *Proceedings of the International Conference on Computer Vision*, pp. 40–45, 1995.

[12] S.-W. Shih, Y.-P. Hung, and W.-S. Lin, "Calibration of an active binocular head," *IEEE Transactions on Systems, Man and Cybernetics–Part A: Systems and Humans*, vol. 28, pp. 426–442, July 1998.

[13] K. Brunnstrom, J.-O. Eklundh, and T. Uhlin, "Active fixation for scene exploration," *International Journal of Computer Vision*, vol. 17, no. 2, pp. 137–162, 1996.

[14] H. Wu, T. Fukumoto, Q. Chen, and M. Yachida, "Active face observation system," in *Proceedings of the International Conference on Pattern Recognition*, vol. 3, pp. 441–445, 1996.

[15] A. Davison and D. Murray, "Mobile robot localisation using active vision," in *the Proceedings of the European Conference on Computer Vision*, pp. 809–825, 1998.

[16] I. D. Reid and D. W. Murray, "Active tracking of foveated feature clusters using affine structure," *International Journal of Computer Vision*, vol. 18, no. 1, 1996.

[17] M. Kam, X. Zhu, and P. Kalata, "Sensor fusion for mobile robot navigation," *Proceedings of the IEEE*, vol. 85, pp. 108–119, 1997.

[18] S. M. Seitz and C. R. Dyer, "Photorealistic scene reconstruction by voxel coloring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1067–1073, 1997.

[19] C.-Y. Lin, S.-W. Shih, and Y.-P. Hung, "Toward automatic reconstruction of 3d environment with an active binocular head," in *Proceedings of the International Conference on Pattern Recognition*, vol. 2, pp. 1708–1710, 1998.

[20] K. N. Kutulakos and S. M. Seitz, "A theory of shape by space carving," in *Proceedings of the International Conference on Computer Vision*, pp. 307–314, 1999.

[21] R. Beß, D. Paulus, and H. Niemann, "3d recovery using calibrated active camera," in *Proceedings of the IEEE International Conference on Image Processing*, pp. 855–858, 1996.

[22] N. Maru, A. Nishikawa, F. Miyazaki, and S. Arimoto, "Active binocular stereo," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 724–725, 1993.

[23] E. Grosso and M. Tistarelli, "Active/dynamic stereo vision," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 868–879, 1995.

[24] N. Ahuja and L. Abbott, "Active stereo: Integrating disparity, vergence, focus, aperture, and calibration for structure estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 1007–1029, 1993.

[25] E. Marchand and F. Chaumette, "An autonomous active vision system for complete and accurate 3d scene reconstruction," *International Journal of Computer Vision*, vol. 32, no. 3, pp. 171–194, 1999.

[26] Y.-P. Hung, D. B. Cooper, and B. Cernuschi-Frias, "Asymptotic bayesian surface estimation using an image sequence," *International Journal of Computer Vision*, vol. 6, no. 2, pp. 105–132, 1991.

[27] M. Okutomi and T. Kanade, "A multiple-baseline stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 4, pp. 353–363, 1993.

[28] S. D. Blostein and T. S. Huang, "Error analysis in stereo determination of 3-d point positions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 6, pp. 752–765, 1987.

[29] D. B. Cooper, J. Subrahmonia, Y.-P. Hung, and B. Cernuschi-Frias, *The Use of Markov Random Fields in Estimating and Recognizing Objects in 3D Space Markov Random Fields: Theory and Applications, edited by Rama Chellapa and Anil Jain*. Academic Press, 1993.

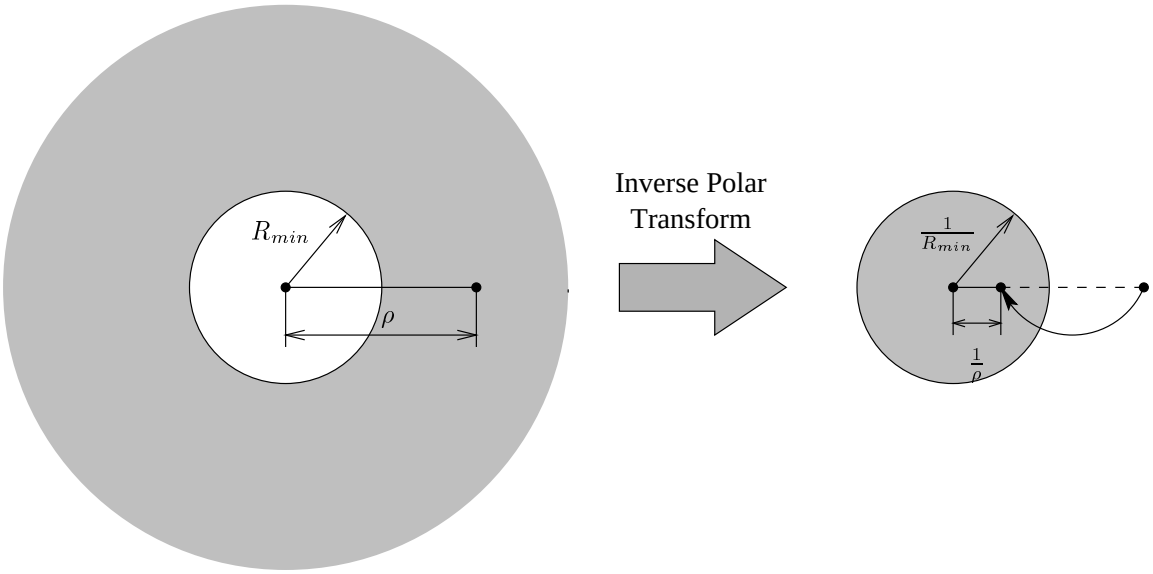


Figure 1: Inverse polar transform.

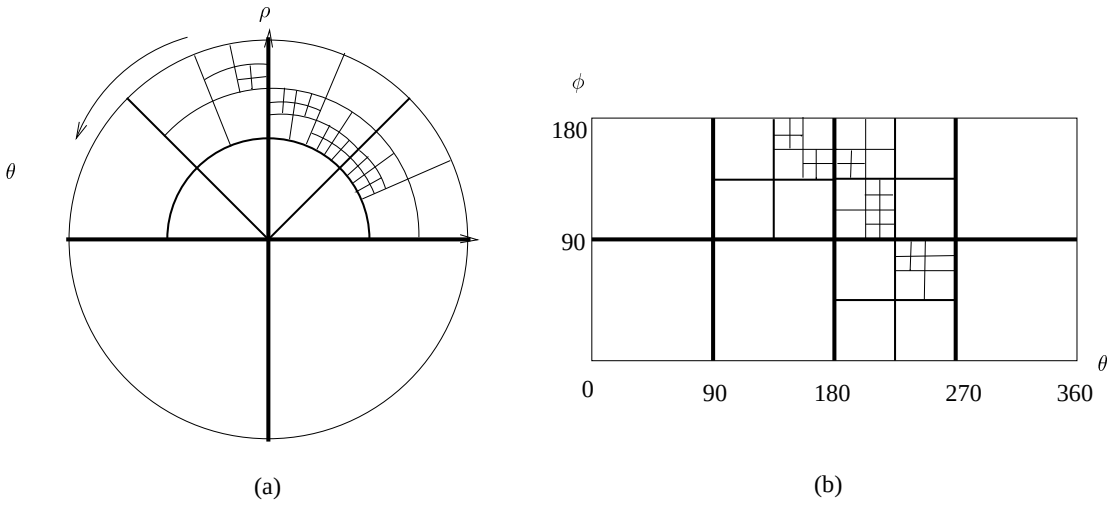


Figure 2: An illustration of the inverse polar octree.

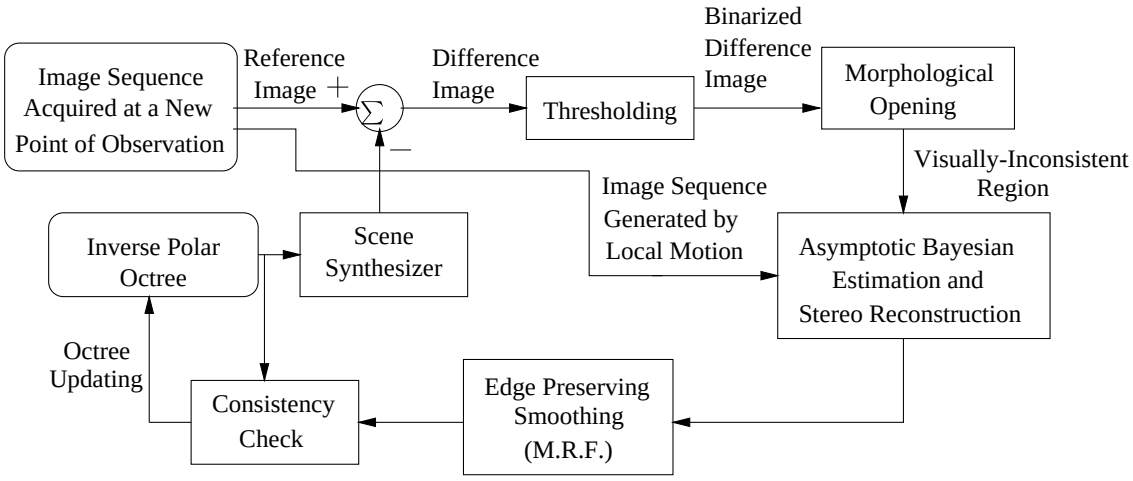


Figure 3: The schematic diagram of the automatic 3-D reconstruction process.

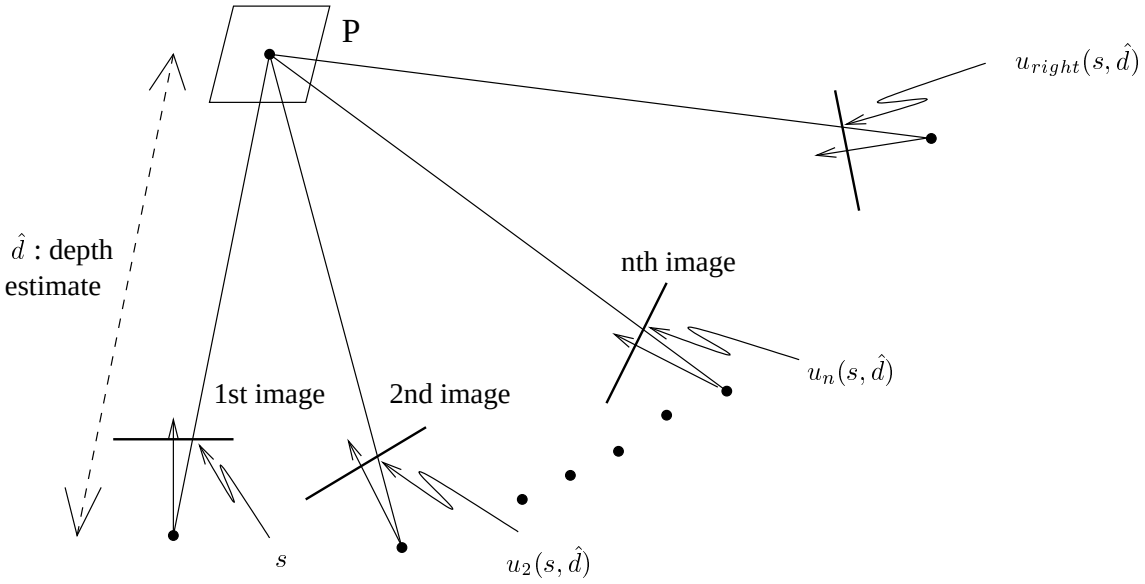


Figure 4: The stereo correspondence of pixel s computed by using the estimated depth $\hat{d}$ and the relative camera geometric parameters.

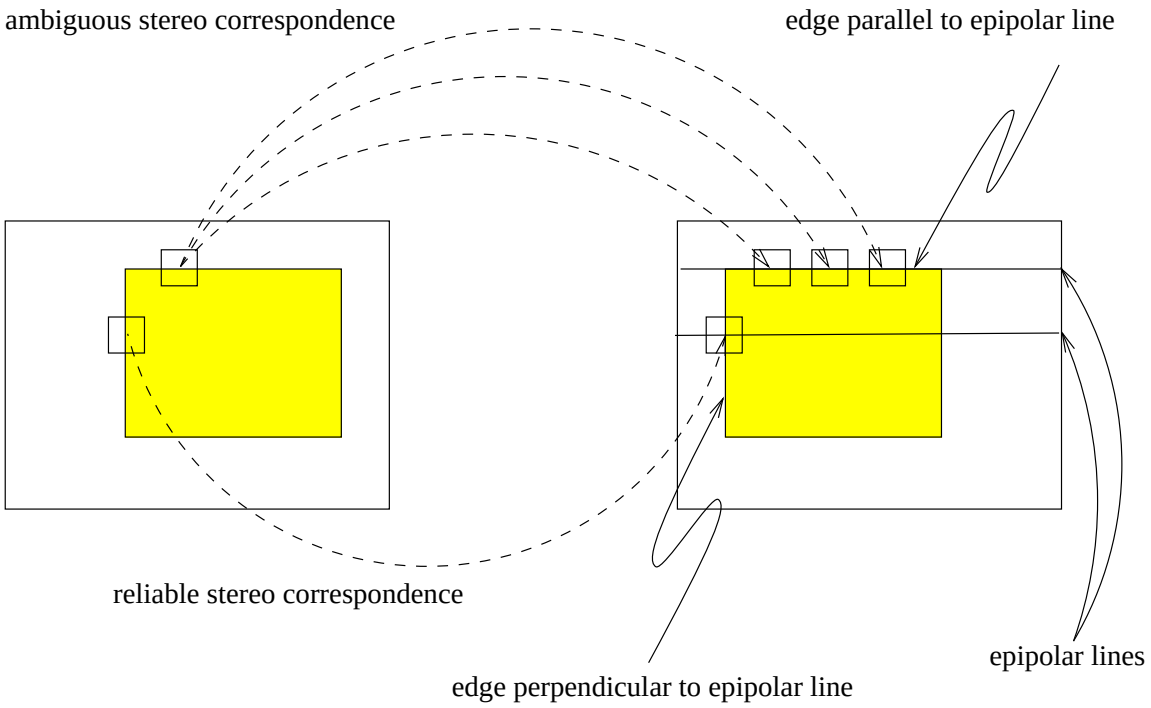


Figure 5: Edge parallel to the epipolar line will cause ambiguity in stereo matching.

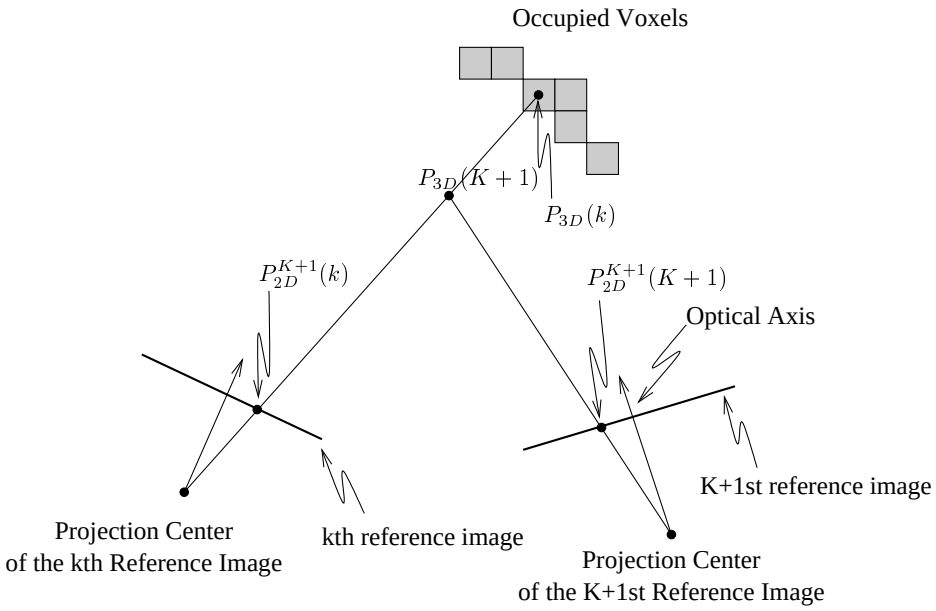


Figure 6: Geometrically incompatible 3-D data. Notice that $P_{3D}(k)$ is occluded by $P_{3D}(K+1)$ when observed from the projection center of the kth reference image.

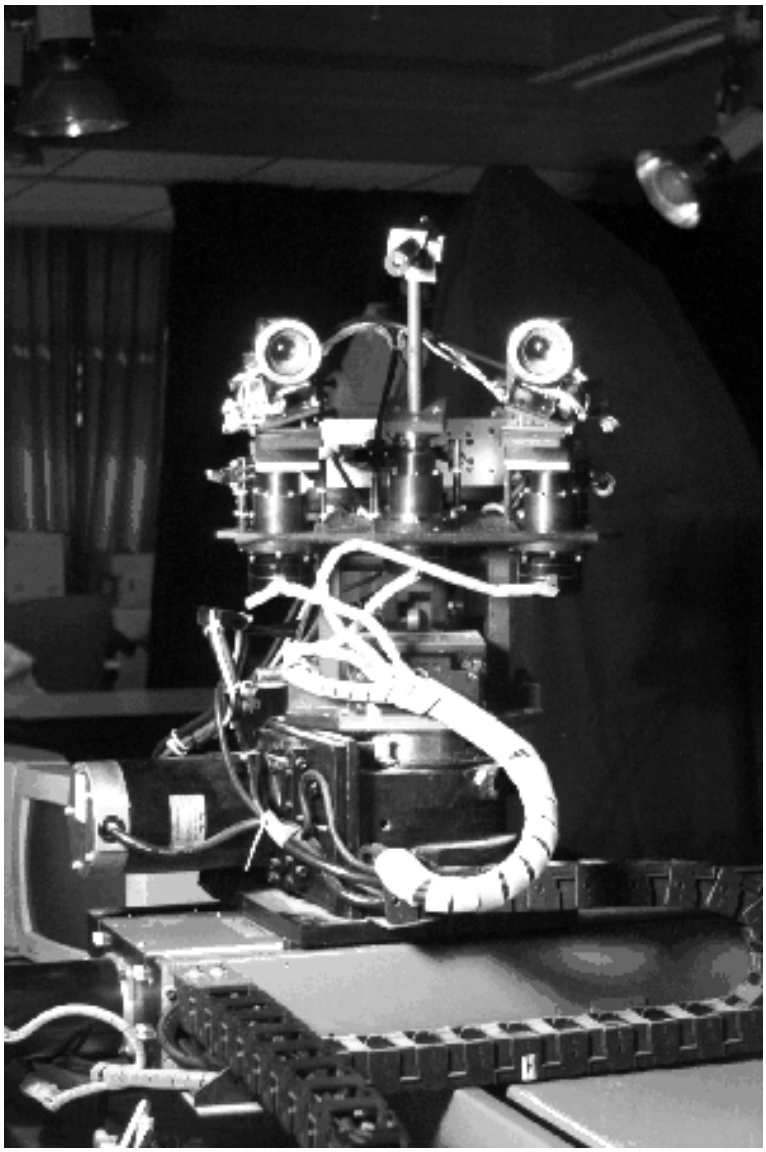


Figure 7: The active binocular head (the IIS head) used in the experiments.

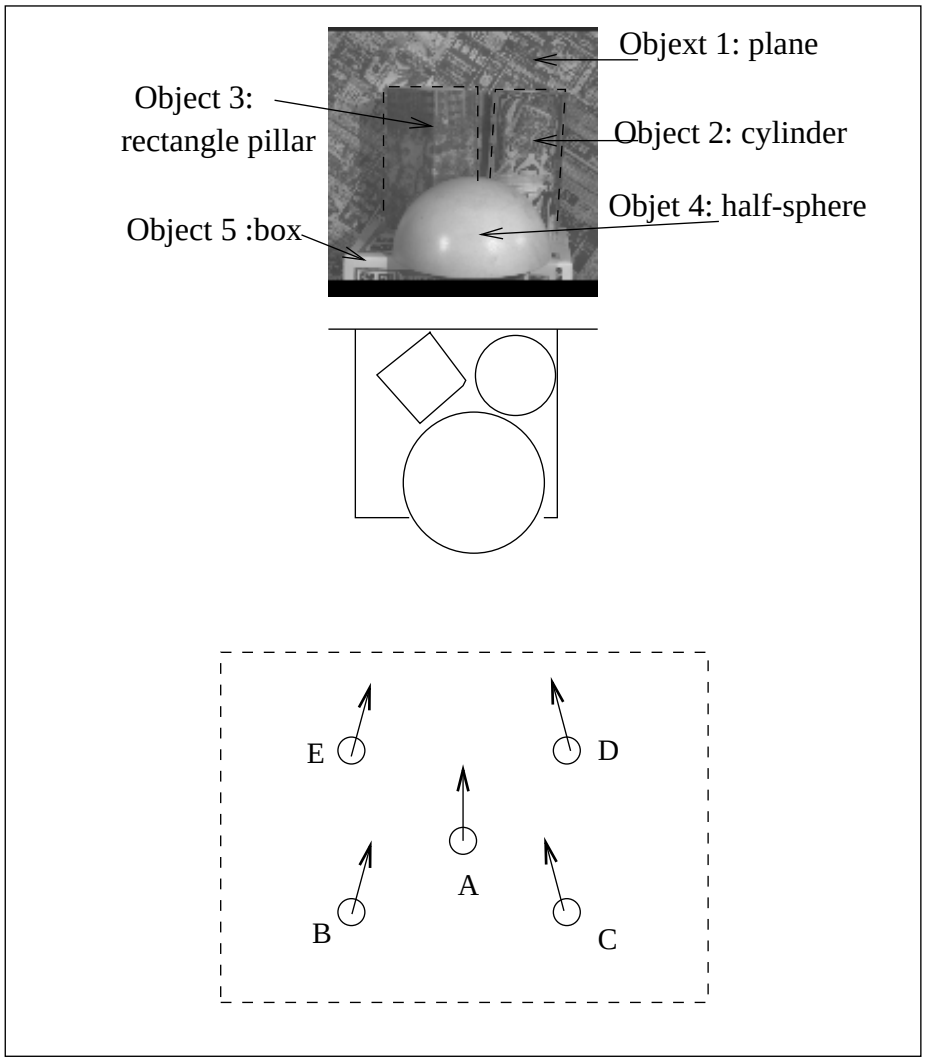


Figure 8: The distribution of the 5 points of view used to acquired the images as shown in Figure 9.

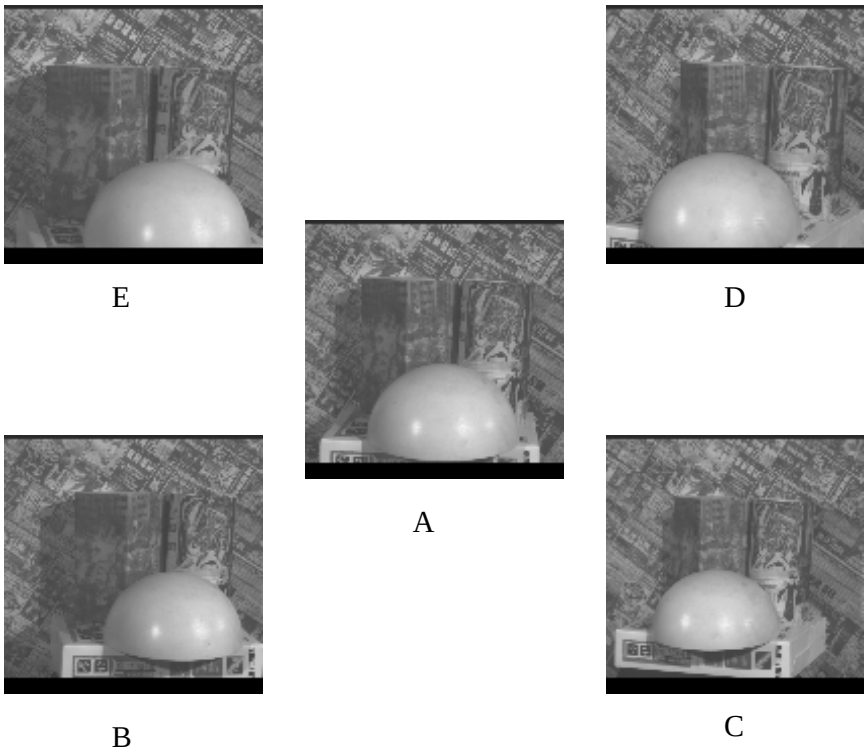


Figure 9: 5 reference images acquired by the left camera at the 5 points of view as shown in Figure 8.

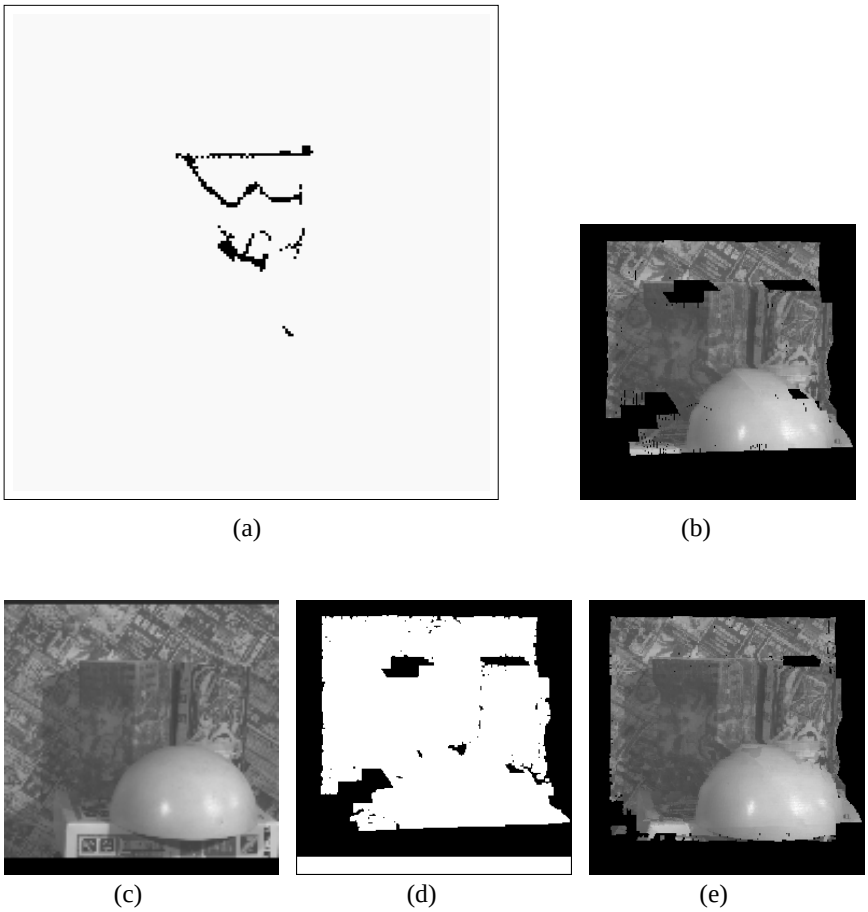
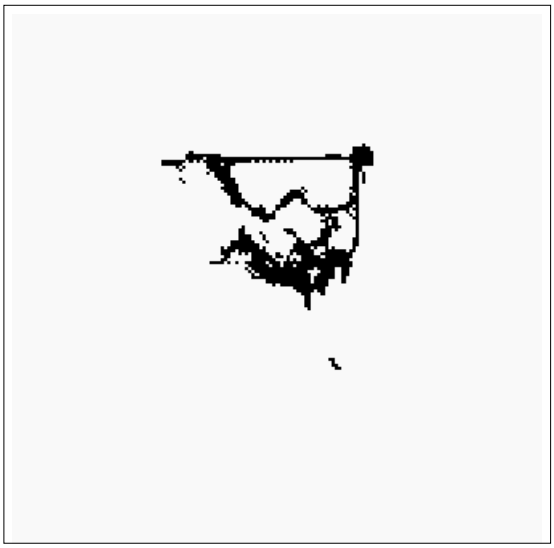
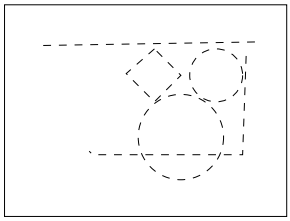


Figure 10: (a) The 3D data contained in the octree reconstructed at viewpoint A were projected to a plane parallel to the ground. (b) An image synthesized at viewpoint B using the octree data shown in (a). (c) The observed image at viewpoint B. (d) The difference image shows the virtually-inconsistent regions. (e) An image synthesized at viewpoint B using the octree data updated by the depth estimates in the visually-inconsistent regions of (d)



(a)



(b)

Figure 11: (a) The 3D data contained in the octree reconstructed at the 5 viewpoints were projected to a plane parallel to the ground. (c) The corresponding objects of (a)

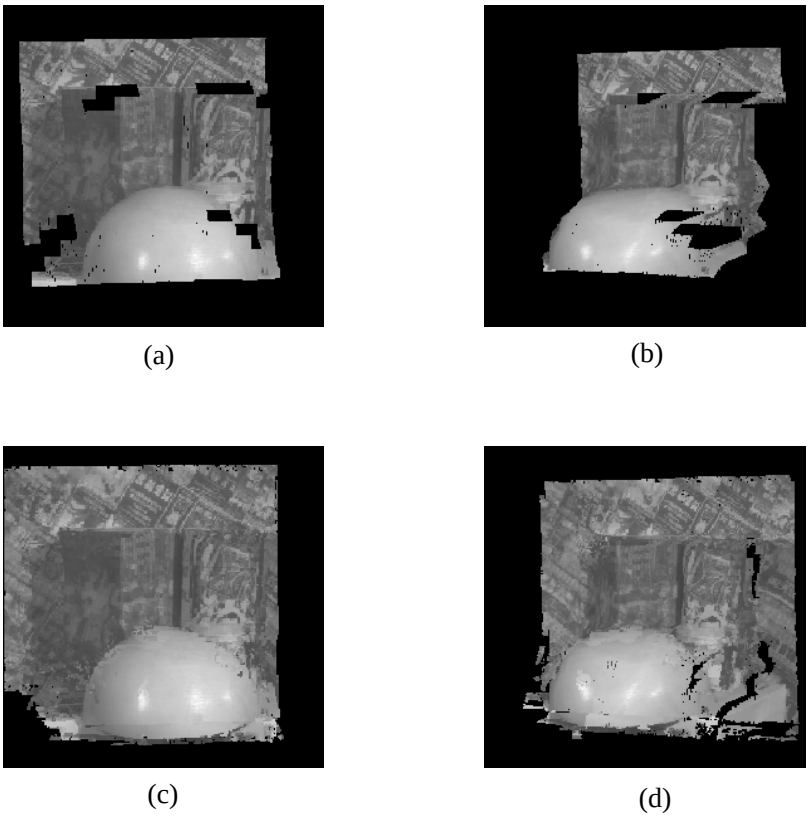


Figure 12: (a), (b) The images synthesized at a virtual viewpoint F which is located between viewpoints A, B and virtual viewpoint G which is located at the right side of viewpoint C by using the octree data reconstructed after taking observations at viewpoint A. (c), (d) The images synthesized at a virtual viewpoint F and G by using the octree data reconstructed after taking observations at viewpoint A–E.

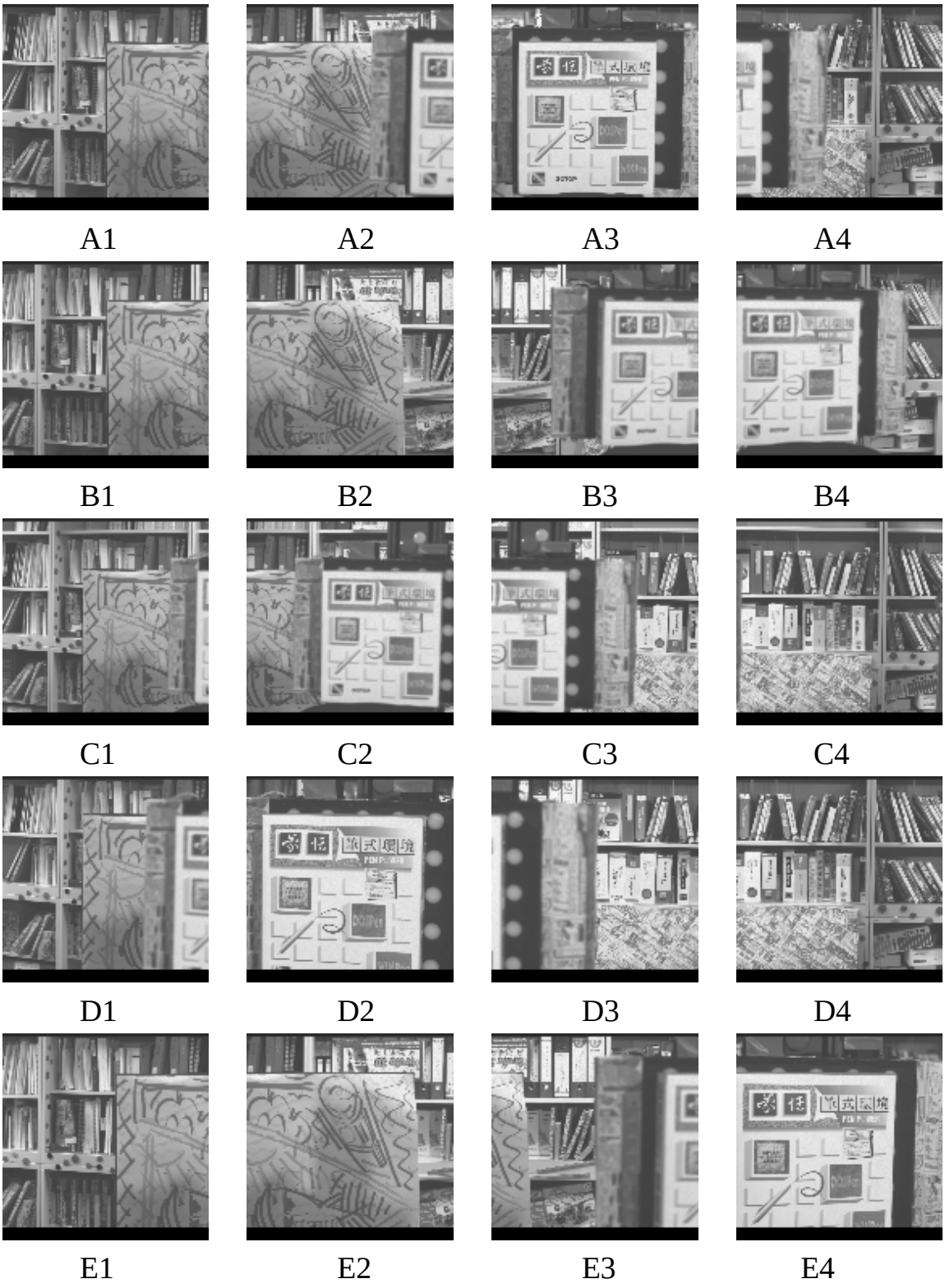


Figure 13: Twenty reference images acquired by the left camera at the twenty viewpoints, $A1$, $A2$, ..., and $E4$, illustrated in Figure 14.

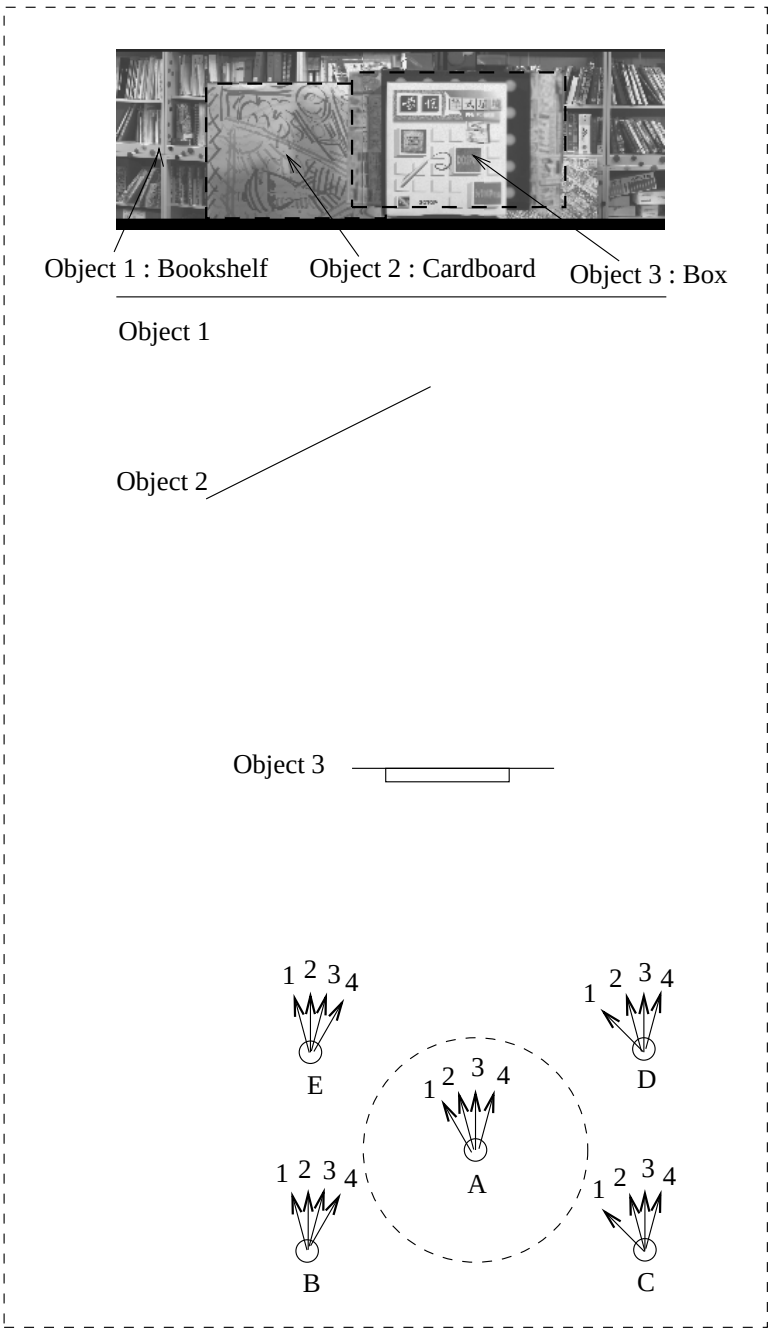
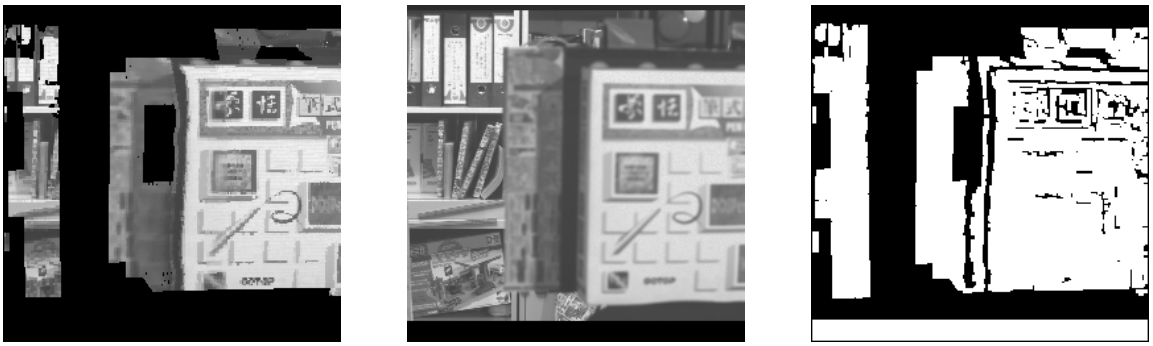


Figure 14: The relative position of the objects seen in the scene and the twenty viewpoints, $A1$, $A2$, ..., and $E4$, used for taking observations.



(a) (b) (c)

Figure 15: (a) Synthesized image (b) Observed image (c) Visually-inconsistent regions (those regions marked with black color).

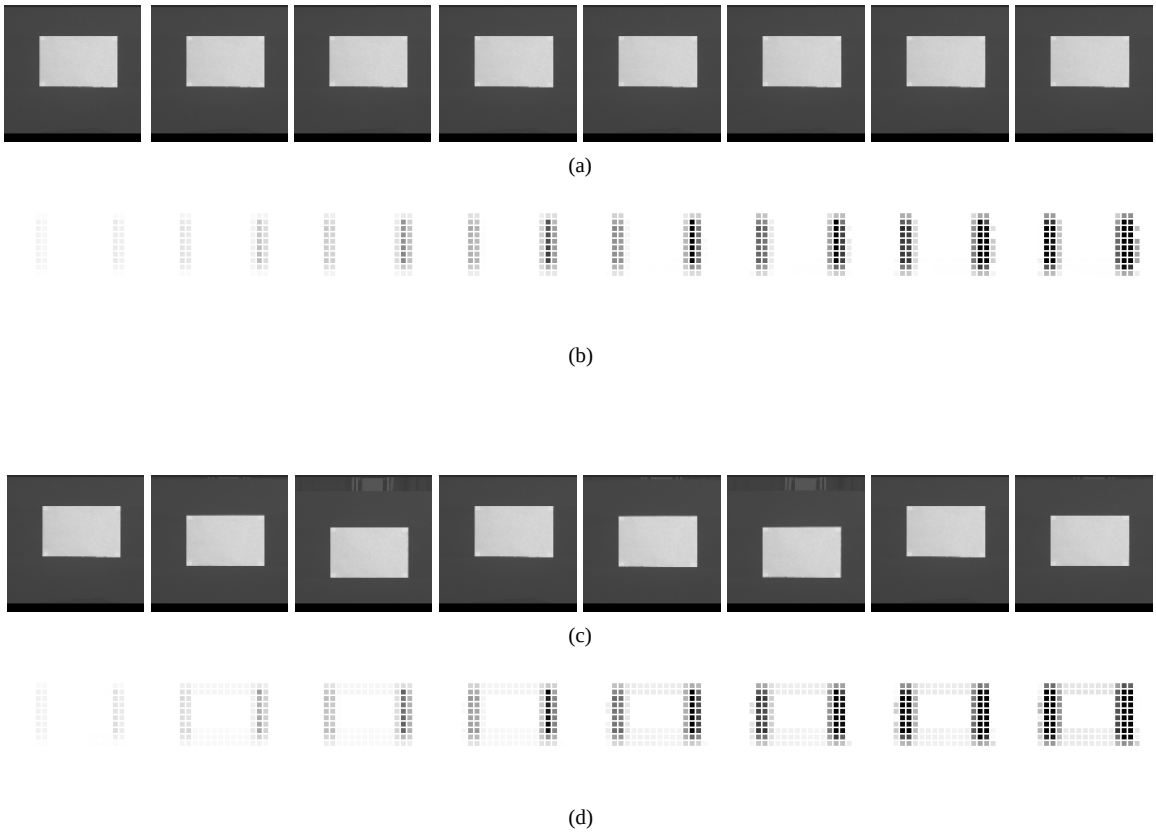


Figure 16: (a) A sequence of images acquired when the active camera was moved horizontally. (b) The reciprocal variance value Φ for each image in (a). (c) Images acquired in a sequence of local camera motion whose path is determined on line. (d) The reciprocal variance values Φ of the horizontal edges increased when the local motion planner functions.

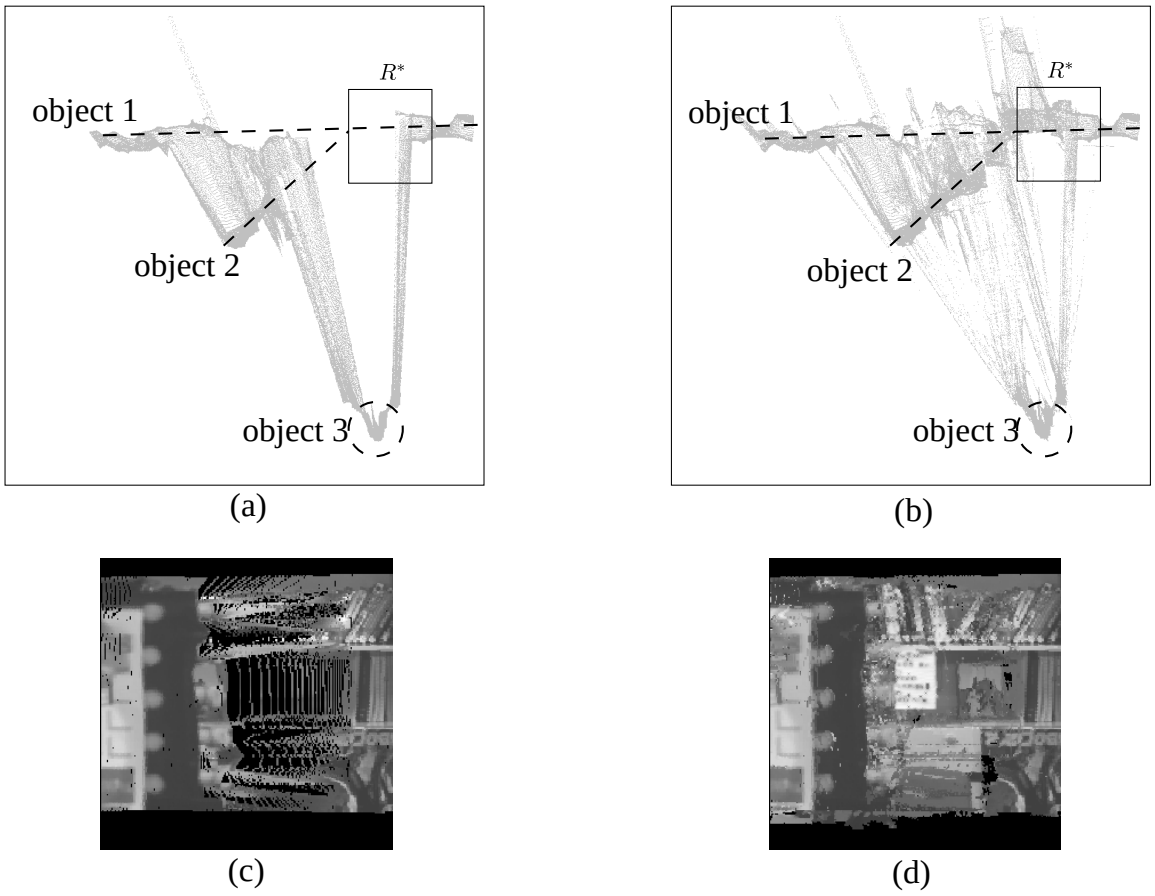


Figure 17: (a) The 3-D data contained in the octree reconstructed at viewpoints $A1$–$A4$ were projected to a plane parallel to the ground. (b) The 3-D data contained in the octree reconstructed at the twelve viewpoints were projected to a plane parallel to the ground. (c) An image synthesized at a viewpoint, V, located between viewpoints A and C by using the octree data shown in (a). (d) An image synthesized at the viewpoint, V, by using the octree data shown in (b).

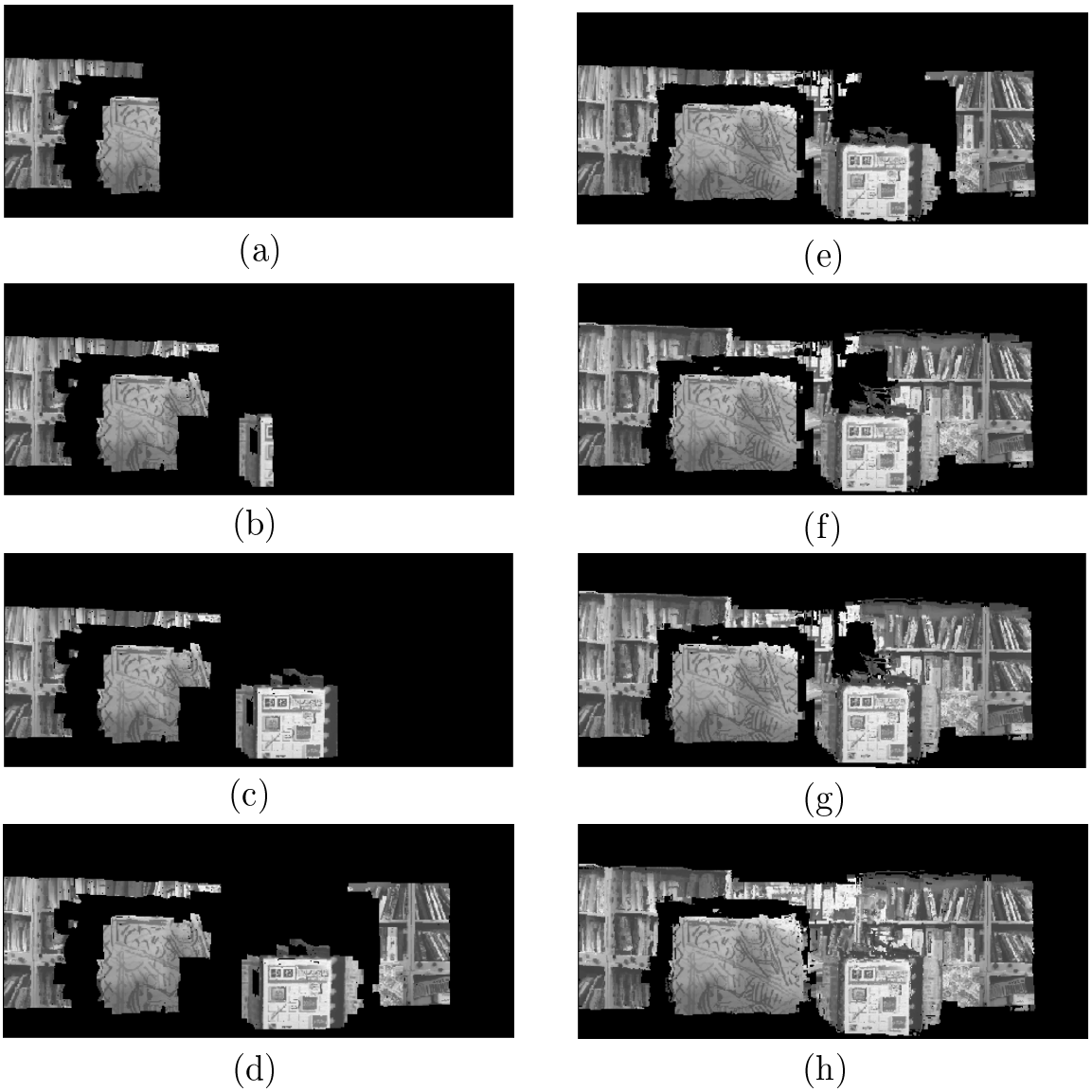


Figure 18: Synthesized images from a virtual viewpoint slightly overlooking the scene, where (a), (b), (c), (d), (e), (f), (g) and (h) are synthesized by using the IPO reconstructed after taking observations at viewpoints $A1$, $A1$–$A2$, $A1$–$A3$, $A1$–$A4$, $A1$–$B4$, $A1$–$C4$, $A1$–$D4$, and $A1$–$E4$, respectively.

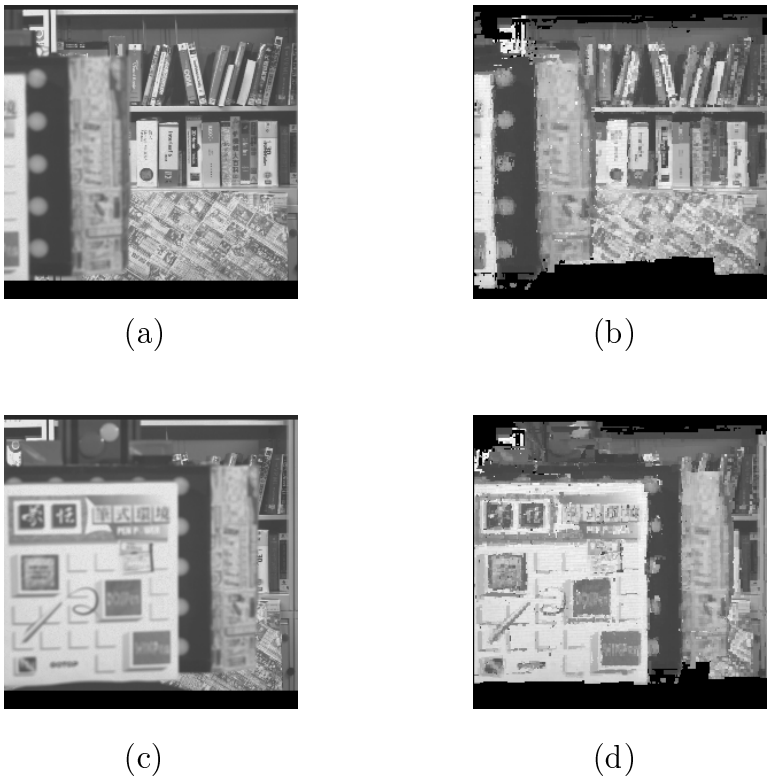


Figure 19: (a) The real image captured at a viewpoint, V, which is located between viewpoints $A4$ and $C2$. (b) The synthetic image generated for V by using the reconstructed IPO. (c) The real image captured by another viewpoint, U, located between viewpoints $A3$ and $B4$. (d) The synthetic image generated for U by using the reconstructed IPO.

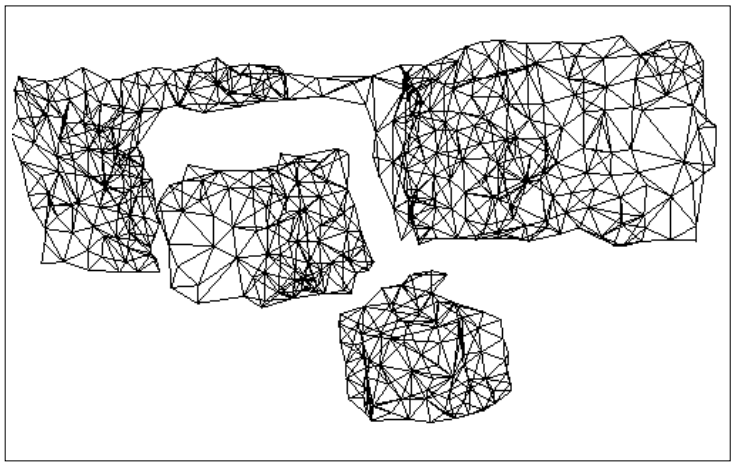


Figure 20: The triangular meshes obtained by converting the 3-D data contained in the reconstructed IPO.

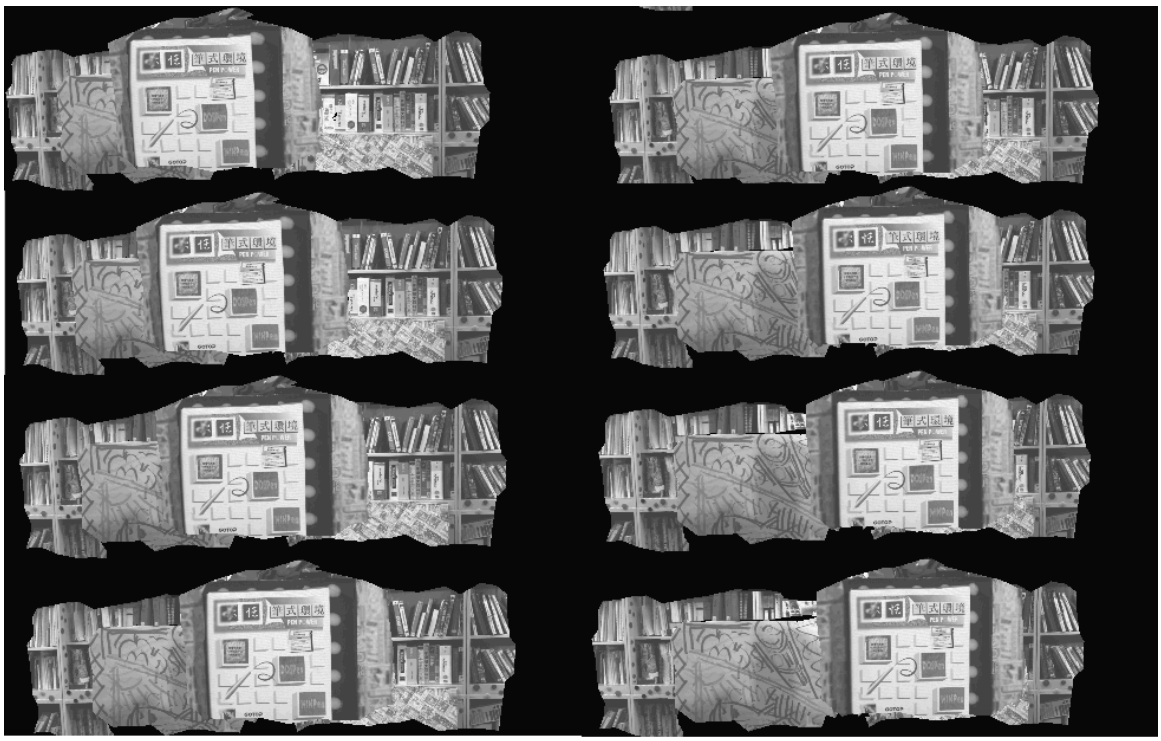


Figure 21: An image sequence generated by using the texture-mapped triangular meshes.

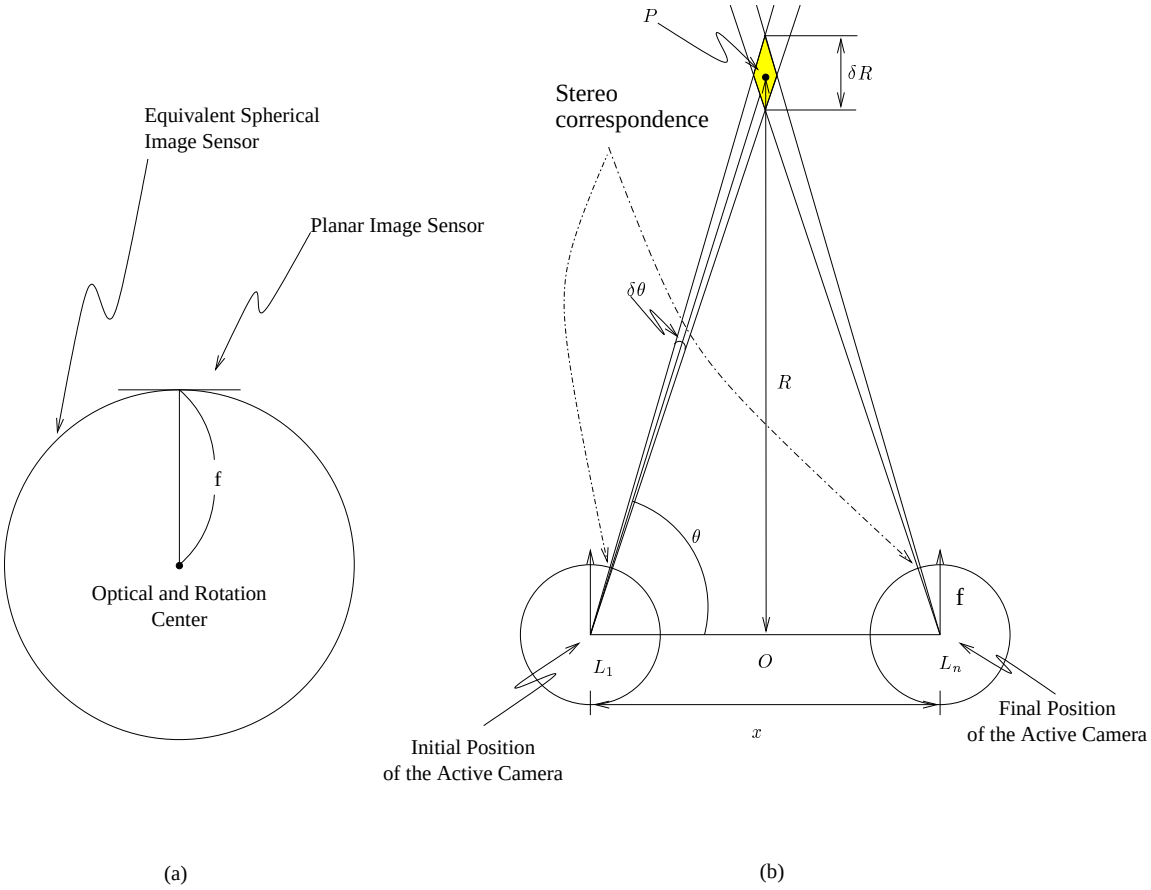


Figure 22: The depth uncertainty and the effective length of the camera local motion

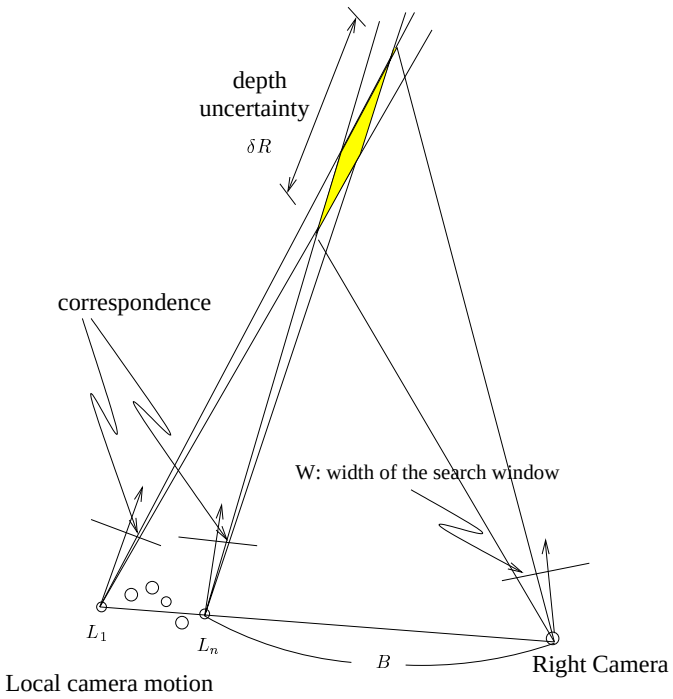


Figure 23: The relation between the depth uncertainty evaluated by local camera motion and the search region of stereo correspondence.