

Vision Sensor Planning for 3D Model Acquisition

S. Y. Chen, *Member, IEEE*, and Y. F. Li, *Senior Member, IEEE*

Abstract—A novel method is proposed in this paper for automatic acquisition of three-dimensional models of unknown objects by an active vision system, in which the vision sensor is to be moved from one viewpoint to the next around the target to obtain its complete model. In each step, sensing parameters are determined automatically for incrementally building the 3D target models. The method is developed by analyzing the target's trend surface, which is the regional feature of a surface for describing the global tendency of change. Whilst previous approaches to trend analysis are usually focused on generating polynomial equations for interpreting regression surfaces in three dimensions, this paper proposes a new mathematical model for predicting the unknown area of the object surface. A uniform surface model is established by analyzing the surface curvatures. Furthermore, a criterion is defined to determine the exploration direction and an algorithm is developed for determining the parameters of the next view. Implementation of the method is carried out to validate the proposed method.

Index Terms—vision sensor, viewpoint planning, sensor placement, 3D modeling, trend surface, surface prediction, model acquisition.

I. INTRODUCTION

USING machine vision to obtain the 3D model of an object is very important to many practical applications. In such tasks, when there is no prior information about the objects or environments, it is desired to automatically generate a multi-viewpoint plan at run-time. The goal of viewpoint decision or sensor placement is to gain knowledge about the unseen portions of the object while satisfying the placement constraints such as focus, field-of-view, occlusion, and collision avoidance. Such a strategy determines each subsequent vantage point and offers the benefit of reducing or eliminating the manual work that otherwise would be required for acquiring an object's surface features. A system without viewpoint planning typically has to utilize as many as tens of range images for a typical task, with significant overlap between them.

The research on actively moving a vision sensor for modeling objects [1] has been active for more than ten years.

Manuscript received November 10, 2004. This work was supported by the Research Grants Council of Hong Kong [Project no CityU1130/03E] and the National Natural Science Foundation of China [NSFC-60405009] and [ZJNSF-Y104185]. This paper was presented in part at the IEEE International Conference on Robotics and Automation, New Orleans, USA, April 2004.

S. Y. Chen is with Zhejiang University of Technology and National Laboratory of Pattern Recognition of CASIA, China (e-mail: sy@ieee.org).

Y. F. Li (*corresponding author) is with the Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong, Hong Kong (e-mail: meyfli@cityu.edu.hk).

Since 1988 when an early attempt was made in this aspect by Cowan and Kovesi [2], many works have been published on sensor placement or viewpoint planning. In the early stage, the work focused mainly on sensor modeling, analysis of sensor's optical and geometrical parameters, and sensor placement constraints. Later, relevant applications were investigated in vision-based inspection and object recognition using CAD models. The "generate-and-test" method and the synthesis method were the major topics explored at that stage.

While optimization is desired for model-based sensor placement [3], it is a problem for planning viewpoints for unknown objects or environment, which is important in many tasks using machine vision. The reported results of research work in this aspect were still far from practical applications. The majority of work carried out in this problem is mainly to find the best views to digitize an object without missing zones, and with a minimum number of views. For example, Papdopoulos-Orfanos and Schmitt [4] utilized a volumetric representation and worked on a solution to the next best view problem for a range scanner with narrow and short viewing field. Their work focused on collision avoidance as the small field of view makes the sensors navigate closely to the object.

Among the previous approaches to the modeling problem, "occlusion" and uncertainty have been strongly associated with viewpoint planning for some time. Kutulakos *et al.* [5] utilized the changes in the boundary between sensed surface and occlusion with respect to sensor position to recover a shape. A similar histogram-based technique was used by Maver and Bajcsy [6] to find the viewing vector that would illuminate the most edge features derived from occluded regions. Whaite and Ferrie [7] used a sensor model to evaluate the efficacy of the imaging process over a set of discrete orientations by ray-casting: the sensor orientation that would hypothetically best improve the model is selected for the next view. The work by Pito [8] removed the need to ray-cast from every possible sensor location by determining a subset of positions that would improve the current model. An uncertainty driven approach was investigated [9], to maximize the information gain for the next view.

On the next best view (NBV) problem for incremental object modeling, Zha *et al.* [10] addressed two main issues to determine the next-best-viewpoint: 1) a uniform tessellation of the spherical space and its mapping onto the 2D array; 2) incremental updating computations from evaluating viewpoints as the NBV. Yu *et al.* [11] proposed to determine the next pose of the range sensor by analyzing the intersections of planar surfaces obtained from the previous

images, for the unseen parts of the scene that can be observed most. They set up a sphere around the scene and the next view is selected that gives the largest unseen area. Pito *et al.* [8], [12] presented a solution for planning a range camera in the process of digitizing unknown parts. Arbel *et al.* [13] showed how entropy maps could be used to guide an active observer along an optimal trajectory and how a gaze-planning strategy could be formulated by using entropy minimization as a basis for choosing a next best view. In [14], Banta and Abidi described a system in which, ideally, a NBV will reveal the greatest quantity of previously unknown scene information. Reed *et al.* [15] determined the visibility volume, which is the volume of space within which a sensor has an unobstructed view of a particular target. In [16], [17], the scene was assumed to be composed only of polyhedral objects and cylinders. The technique proposed to solve the NBV problem was a depth-first search algorithm and the strategy ensured the completeness of the scene reconstruction.

Another factor to be considered in viewpoint planning is the constraints that must be satisfied. To this end, two distinct methods have been widely used: the weighted function method and tessellated space approach. The former [15], [18]-[22] employs a function that combines several components representing the placement constraints. This method is usually used in model-based planning tasks [23]. The latter method [8]-[10], [24], [25] is mainly for object modeling tasks. It tessellates a sphere or cylinder around the object to be modeled, producing a viewpoint space or look-up array [24]. The object surface is partitioned as void surface, seen surface, unseen surface, and uncertain surface. The working space is also partitioned into void volume and viewing volume. An algorithm is then developed to plan a sequence of viewpoints so that the whole object can be sampled. This method is effective for dealing with some small and simple objects, but it is difficult to model a large and complex object, e.g. an environment with many concave areas, as it cannot solve the problems of occlusion constraint.

Therefore, previous efforts were often made on finding the best next views by volumetric analysis or occlusion as a guide. However, since there does not exist any information about the unknown target, it is actually impossible to give the true best next view. Although sensor placement has been studied for more than 15 years and there are a number of relevant publications available, a large part of the bibliography deals with model-based tasks such as for inspection or recognition. The existing work on sensor placement for object modeling is usually implemented with restriction in a specific kind of applications. Further investigation is very necessary for this issue, especially for modeling of large objects or environments.

This paper proposes a new idea of sensor placement in sense of target driven and shape prediction. It involves decision of exploration direction and determination of the next view. The trend surface is used as a cue to predict the unknown portion of an object and the next best viewpoint is determined by the expected surface.

II. INCREMENTAL MODELING AND VIEWPOINT PLANNING

In 3-D modeling tasks, an incremental method is normally used for exploring the unknown portion of an object or environment. This results in a sequence of viewing actions for local model acquisition. Each successive sensing gives some new information that is registered and integrated in the global model.

During such a process, the first view can be set at an arbitrary pose. A 3-D depth image M_0 is then generated and registered as the initial object model O_0 . Next, a new viewing pose P_{i+1} ($i=0, 1, \dots, n$) has to be determined so that the modeling process will continue until the whole object is reconstructed. Let the global model reconstructed by previous i steps (from step 1 to i) be

$$M_i = M_i(u, v) = (x(u, v), y(u, v), z(u, v)), \quad (1)$$

where u and v are the coordinates on the digital image. Using a coordinate system whose origin is at the center of the vision sensor with its Z-axis pointing along the viewing direction, the 3-D depth image is represented as:

$$z = M_i^w(x, y), x, y \in \mathbb{R}. \quad (2)$$

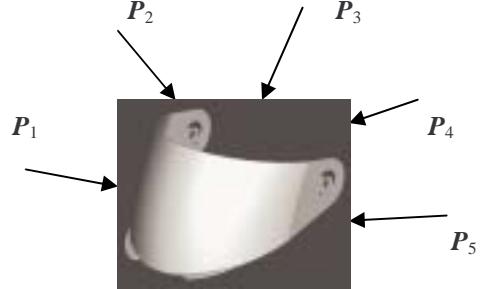


Figure 1. Multiple viewpoints for incremental modeling

Fig.1 illustrates an example in which five viewpoints are used for acquisition of the structural model, from P_1 to P_5 . The difficulty here is that there is no prior knowledge about the object. Therefore, each successive viewpoint (from P_2 to P_5) must be determined during the run-time. The key problem of sensor placement in such a task is to determine a most feasible viewpoint called the *Next-Best-View* or *Next-Best-Pose* [10], by utilizing the partially known global model that has already been built in the previous steps. Algorithms of sensor placement or viewpoint planning have to be developed in such vision systems for determining the sensor's position, orientation, and optical settings dynamically. In this paper, the next viewpoint is determined according to the surface trend of the known partial model.

III. TREND SURFACE FOR MODEL PREDICTION

A. Trend Analysis

Surface trend describes the global shape of a surface. Trend surface analysis is a global method for processing spatial data. Mathematically, a mapped surface can be separated into two components - that of the trend and the residuals from the trend. The trend is the regional feature of a surface, and the residuals are the local fluctuations of high frequent features (Fig. 2).

Trend surface analysis is often used for fitting and interpolating regression surfaces in three dimensions as smoothed representation of area data. It is assumed that the spatial distribution of a particular phenomenon can be represented by some form of continuous surface, usually a defined geometric function. The observed spatial pattern can be regarded as the sum of such a surface and a "random", or local, term. The surface is a function of two orthogonal coordinate axes which can be represented by

$$z = f(x, y) + e, \quad (3)$$

in which the variable z at the point (x, y) is a function of the coordinate axes, plus the error term e . This expression is the generalized form of the General Linear Model (GLM), which is the basis of most trend methods.

The function $f(x, y)$ is usually expanded or approximated by various terms to generate polynomial equations. The principles of trend surface analysis were set out initially by Agocs [26]. To develop complex, smoothed equations for geophysical data by expanding the summation term of the General Linear Model, Krumbein [27] defined the relationship between standard multivariate regression analyses and trend methods. This expansion was performed by incorporating power terms and cross-products of the x and y coordinates. For an n -order three-dimensional surface, the form of the power series is given by

$$f(u, v) = \sum_{i=0}^n \sum_{j=0}^i b_{ij} u^j v^{i-j}, \quad (4)$$

where u and v are the coordinates on an arbitrary orthogonal reference system, b_{ij} is the constant coefficient of the surface (b_{00} is the surface base).

The trend part is useful for predicting the unseen part of an object or environment and is thus used for determining the next viewpoint in this research. The residuals (local features)

do not affect viewpoint planning much, but they should be filtered out during the image processing.

Let a single surface M be split into two parts, M_1 and M_2 ,

$$M = M_1 \cup M_2.$$

If the surface M changes smoothly, both the trends of M_1 and M_2 should be approximately equal to the trend of M , i.e.

$$\text{Trend}(M) \approx \text{Trend}(M_1) \approx \text{Trend}(M_2). \quad (5)$$

Suppose the vision agent has already captured a part of the surface, say M_1 , but M_2 remains unknown. Then by computing the surface trend of M_1 , the surface shape of M_2 can be predicted. In this paper, we will not use (3) or (4) directly as the trend model for surface prediction, since it relies on interpreting regression of the known area. Instead, we will develop a new mathematical model for describing the surface trend, thus emphasizing on the prediction of the unknown area.

With the partially known model, the sensor pose for a next view can be decided according to the known information. Here two steps are used to make this decision. The first step is to determine the exploration direction and the second is to determine the sensor pose according to the surface trend.

B. Exploration Direction

At each successive action in the 3D surface acquisition, only one direction, called exploration direction, can be chosen for planning the next viewpoint. This direction needs to be determined in three steps. The first step is to detect boundary points of the object. It segments the target from the scene and finds the points that are near to unknown area but on the object surface. In the second step, these points are ranked by a rating function. The third step is to select a point with the highest rating to give the exploration direction. The first and third steps can be solved in a straightforward way. For the second step, the rating function is described as

$$R(P) = \frac{C_r E^{C_e}}{(n_{\text{order}} + 1)^{C_n}}, \quad (6)$$

where n_{order} is the surface order at point P , E is the expected volume of the unknown area near P . C_r , C_n , and C_e are adjustable coefficients and can be chosen according to application requirements. C_n reflects the importance of new object information and C_e reflects the importance of

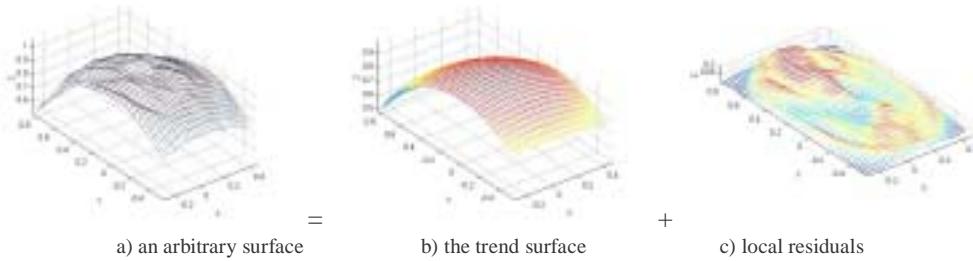


Figure 2. The trend is the regional feature of a surface

exploration reliability. C_r is a global scale factor or can be set to be 1. The result $R(P)$ is the rated value for a boundary point P .

The surface order (i.e. the smoothness of the surface) is used as the main criterion. The reason is that the trend surface can predict the unknown area accurately where the surface has a low order. Fig. 3 illustrates the selection criterion of the exploration direction.

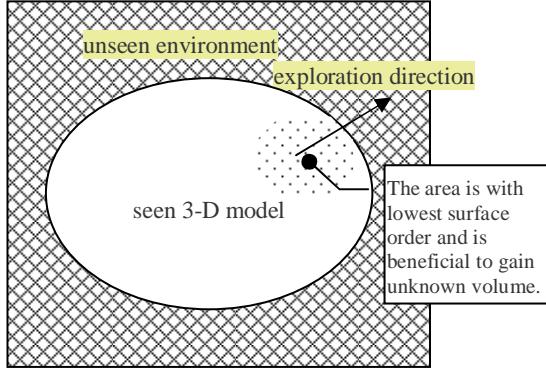


Figure 3. Exploration of object surface

The surface order is defined according to (4) with the same fitting error. To avoid computation of surface fitting, we can approximately compute the integral value of the curvatures in a small area, i.e.

$$n_{\text{order}}(u, v) \approx \iint_{x,y \in S(u,v)} k_{\min}(x, y) dx dy, \quad (7)$$

where $S(u, v)$ is the neighborhood area of point (u, v) . The area size of domain S is dependent on the sensor's Field-of-View (FOV), e.g. with $20 \times 20 \text{ mm}^2$ or $30 \times 30 \text{ pixels}$. $k_{\min}(x, y)$ is the curvature at point (x, y) along a specific direction.

It is only necessary to compute the surface orders in the areas near to the boundary of the known surface. The surface order in the center area of the known model does not affect the exploration direction. After the minimum surface order is obtained, i.e. $n_{\min} = \min\{n_{\text{order}}(u, v)\}$, the exploration direction is set to be along the outer direction to the unknown area.

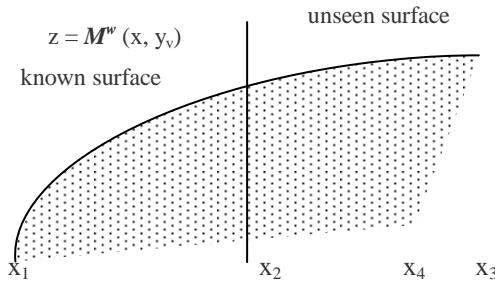


Figure 4. A curve on the sectional plane

C. Surface Prediction

Except for surface edges and object boundaries, since the curvature of a trend surface changes smoothly, the unknown part of object surface can be predicted by analyzing the curvature tendency of the known surface.

For a surface point, there are different curvatures along different directions, although the principal curvatures and Gaussian curvature are the most frequently used ones. To reduce the computation complexity, we may just compute the curve curvatures along the exploration direction. Without loss of generality, we can describe the mathematical formula along the horizontal direction (otherwise a coordinate transformation need be performed). Using a vertical sectional plane which is parallel to x -axis, at $y = y_v$, to cut through the 3-D surface, we obtain a surface curve (see Fig.4),

$$z_v = f_{yv}(x). \quad (8)$$

The curvature of this curve is

$$k(x, y_v) = z_v'' / [1 + (z_v')^2]^{3/2}. \quad (9)$$

Let $X_k = [x_1, x_2]$ be the domain of the known part of the surface curve. To predict the unseen surface, we use a linear regression of x on k and obtain a fitted curve c_v for approximating the curvature tendency on the curve z_v . Hence,

$$c_v(x) = a x + b, x \in [x_1, x_3], \quad (10)$$

where $[x_1, x_3]$ is the whole domain including both the known and unknown area, i.e. $[x_1, x_3] = [x_1, x_2] \cup [x_2, x_3]$. The two parameters a and b are determined by fitting the known part of surface curve, i.e.,

$$a = \frac{6(x_2 + x_1) \int_{x_1}^{x_2} k(x, y_v) dx - 12 \int_{x_1}^{x_2} x k(x, y_v) dx}{(x_1 - x_2)^3}, \quad (11)$$

and

$$b = [\int_{x_1}^{x_2} k(x, y_v) dx - \frac{1}{2}(x_2^2 - x_1^2)a]/(x_2 - x_1). \quad (12)$$

The curvature in the unseen area is expected to be:

$$c_v = \begin{cases} ax + b, & k(x, y_v) < k_{\max}, x \in [x_2, x_4], \\ k_{\max}, & k(x, y_v) \geq k_{\max} \end{cases} \quad (13)$$

where $x_2 < x_4 < x_3$ is a domain for satisfying the constraint that the object surface will be in the FOV of the sensor. k_{\max} is a given a threshold to limit the maximum curvature. It can be set a value in the range of $[1, 20]/\text{Sensing_Distance}$ (1/mm). Then the surface curve in the unseen part of the object will be a solution of the following equation:

$$\|z''\| / [1+(z')^2]^{3/2} - c_v = 0. \quad (14)$$

The solution of this differential equation is:

$$z = \pm \int \frac{(ax^2 + 2bx + 2C_1)}{\sqrt{4 - (ax^2 + 2bx + 2C_1)^2}} dx + C_2, \quad x < (k_{\max} - b)/a, \quad (15)$$

$$\text{or } z = \pm \sqrt{k_{\max}^2 - (x - C_3)^2} + C_4, \quad x \geq (k_{\max} - b)/a, \quad (16)$$

where C_i ($i = 1, 2, 3, 4$) are differential constants which can be determined according to the boundary conditions, such as $z(x_2) = z_2$ and $z'(x_2) = z'_2$. The sign "+" or "-" can also be determined by the known part of the surface curve (convex or concave). Since the predicted curve is based on the analysis of the tendency of known area, it is called the *trend curve*.

IV. VIEWPOINT PLANNING

A. Placement Parameters

In a sensor placement problem, we need to specify the sensor's placement parameters [3]. The placement parameters here include the sensor's position (x_p, y_p, z_p) and orientation (α, β, γ) . The placement constraints can include visibility, focus, field of view, viewing angle, resolution, overlap, occlusion, and some operational constraints such as kinematic reachability of the sensor pose and robot-environment collision. Let the resolution constraint be

$$r_p = 2\sqrt{(x_p - x_2)^2 + [z_p - z(x_2, y_v)]^2} \tan(\frac{\Omega}{2})/N < r_{\max}, \quad (17)$$

where N is the pixel number on a scanning line of the digital image and Ω is the angle of view of the sensor. Equation (17) describes one pixel on the image corresponding to how much size on the actual object surface.

To satisfy the constraints of sensor placement on resolution and FOV, the parameter x_4 in (13) is determined by a searching algorithm. Then the mid-point of a trend curve is:

$$Q_{yv} = (x_v, y_v, z_{mv}), \quad z_{mv} = f(x_v, y_v), \quad x_v = \frac{x_2 + x_4}{2}. \quad (18)$$

By moving the sectional plane to different positions, in the domain of $-Y_{fov} < y_v < Y_{fov}$, we get a series of surface trend curves. Connecting the mid-point of each such curve, we obtain a new curve:

$$L_{i+1} = L(Q_{yv}), \quad -Y_{fov} < y_v < Y_{fov}. \quad (19)$$

where L is the curve function and $(i+1)$ denotes the next view pose.

Calculating the centroid, the position of the reference point (i.e. the new scene center) is obtained:

$$O_{i+1} = (x_{i+1}^o, y_{i+1}^o, z_{i+1}^o), \quad (20)$$

$$\text{where } x_{i+1}^o = \frac{\int_{L_{i+1}} x L dl}{\int_{L_{i+1}} L dl}, \quad y_{i+1}^o = \frac{\int_{L_{i+1}} y L dl}{\int_{L_{i+1}} L dl}, \quad \text{and}$$

$$z_{i+1}^o = \frac{\int_{L_{i+1}} z L dl}{\int_{L_{i+1}} L dl}.$$

Now the sensor position and the viewing direction can be determined. To achieve the maximum viewing angle (i.e. the angle between the viewing direction and the surface tangent) for minimizing the reconstruction uncertainty, the viewing direction is chosen to be the inverse of the average normal on the predicted surface, that is,

$$V_{i+1} = -\frac{\iint N(x, y) dxdy}{\iint dxdy} = -\frac{\iint (\mu \mathbf{i}, \nu \mathbf{j}, \kappa \mathbf{k})(x, y) dxdy}{\iint dxdy} \\ = (-\mu_{i+1} \mathbf{i}, -\nu_{i+1} \mathbf{j}, -\kappa_{i+1} \mathbf{k}), \quad (21)$$

$$\text{where } \mu(x, y) = \frac{\partial f}{\partial x}, \quad \nu(x, y) = \frac{\partial f}{\partial y}, \quad \kappa(x, y) = -1, \quad \text{and } N(x, y)$$

is the surface normal on point (x, y, z) .

The sensor position $P_{i+1} = (x_{i+1}^p, y_{i+1}^p, z_{i+1}^p)$ for the next viewpoint is planned as a solution of the following set of equations:

$$\begin{cases} \frac{x_{i+1}^o - x_{i+1}^p}{\|O_{i+1} - P_{i+1}\|_2} = \frac{-\mu_{i+1}}{\|V_{i+1}\|_2} \\ \frac{y_{i+1}^o - y_{i+1}^p}{\|O_{i+1} - P_{i+1}\|_2} = \frac{-\nu_{i+1}}{\|V_{i+1}\|_2} \\ \frac{\|O_{i+1} - P_{i+1}\|_2 - \frac{r_{\max} N}{2 \tan \frac{\Omega}{2}}}{c_{cmp}} = c_{cmp}, \quad c_{cmp} \geq 0 \end{cases}, \quad (22)$$

where c_{cmp} is a positive constant for compensating the depth value range, Ω is the sensor's Angle-of-View, $\|V_{i+1}\|_2 = \sqrt{\mu_{i+1}^2 + \nu_{i+1}^2 + \kappa_{i+1}^2}$, and $\|O_{i+1} - P_{i+1}\|_2$ is the distance between O_{i+1} and P_{i+1} .

B. Placement Algorithm

The algorithm for the active sensor placement is summarized as below:

Algorithm 1 The algorithm for determining placement parameters

- Step 1. According the 3D surface or partial model obtained from the first view or previous views, find the boundary positions for determining the exploration directions;
- Step 2. For each candidate position, compute its surface order and the expected unknown volume. Then select an exploration direction which is determined by the rating function (7).
- Step 3. With the exploration direction, compute the surface trend or curve trend at the selected position.
- Step 4. Determine the constants a and b by (11) and (12).
- Step 5. Determine C_i and the sign of z in (15) and (16).
- Step 6. Determine x_4 by Algorithm 1;
- Step 7. Determine the mid-point for each trend curve by (18);
- Step 8. Determine the scene center by (20);
- Step 9. Determine the looking direction by (21);
- Step 10. Determine the eye position by (22);

Using this method, the sensor orientation is determined in such a way that it views perpendicularly downward the predicted object surface. The distance to the object is determined so that it enables acquisition of the maximum volume of unknown surface while satisfying some constraints such as resolution and FOV. Finally, the placement parameters of the vision sensor is given as a vector:

$$\mathbf{S}_{i+1} = (x_{i+1}^P, y_{i+1}^P, z_{i+1}^P, \mu_{i+1}, \nu_{i+1}, \kappa_{i+1}), \quad (23)$$

This placement vector is based on the local coordinate system. It needs to be converted to the world coordinate system by multiplying a transformation matrix.

C. The Modeling Process

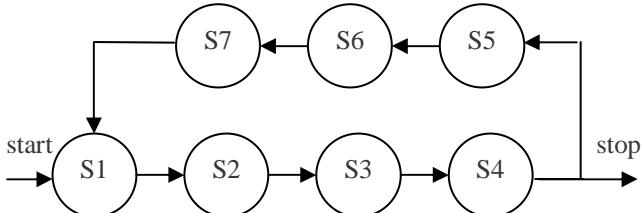


Figure 5. The modeling process

Finally the iterative modeling process is illustrated in Fig. 5, in which the symbols S_i ($i=1, 2, \dots, 6$) represent the states of:

- S1: Acquisition of a view
- S2: Reconstruction of the 3-D local model
- S3: Registration and fusion with the global model
- S4: Model analysis and completion condition checking
- S5: Ranking for selection of exploration directions
- S6: Computing trend surface and determining next view
- S7: Moving the robot to the new viewpoint.

Since surface data sets (triangular meshes or point clouds) are from different viewpoints, they are necessary to register and fuse with the global model in S3. A split and merge algorithm is used to perform such a registration and fusion process. The termination condition in S4 is determined according to a model-completion criterion. In implementation, the known partial model surface is tessellated into a set of small regions. The unknown hole is defined which is the area without data but is neither in known empty volumes nor pointing to unreachable space. An algorithm then detects the unknown holes (uncertain areas) near a small region and estimates their sizes. If the diameter of an unknown hole is larger than certain value d_h (e.g. 10 mm) which is application dependent, a new viewpoint should be planned. Otherwise, the reconstruction process will terminate.

V. ERROR ANALYSIS

It should be noted that the surface predicted by the trend is only the possible surface shape. The resulting error is analyzed here under the assumption that the actual surface is composed of typical 1st or 2nd order curves. We first define a *prediction error*. The *absolute prediction error* is defined as the difference between the expected surface and the actual object surface, i.e.,

$$E = \iint_{M_{i+1}} \| z(x, y) - z_{\text{exp}}(x, y) \| dx dy, \quad (24)$$

where M_{i+1} is the domain for the scene to be observed by the next view. A *relative error* is defined as:

$$E' = \iint_{M_{i+1}} \| \frac{z(x, y) - z_{\text{exp}}(x, y)}{z(x, y)} \| dx dy \times 100\%. \quad (25)$$

Since the shape of a surface curve is computed on the sectional plane, we only need to analyze errors on a 2-dimensional coordinate system for the 1st and 2nd order curves, i.e., straight lines, circular curves, parabolas, elliptical curves, and hyperbolas. These curves are widely used in practical industrial design and most object surfaces can be decomposed to these curves. It is not necessary to consider the original position of a curve because it does not affect the curvature of the object surface. Therefore we always let the original point of a curve be (0, 0).

For a straight line, e.g. $z = c_1x + c_2$, we have $a=0$ and $b=0$ according to (11) and (12). Referring to (15), the predicted curve function $z_p(x)$ is found to be the same as the actual one. Therefore the error to predict a straight line is

$$E_{\text{line}} = \int_{x_2}^{x_4} \| z - z_{\text{exp}} \| dx = 0.$$

Similarly, for a circular curve $z = \sqrt{R^2 - x^2}$, we can also get that $E_{\text{circle}} \equiv 0$. This means that there is no error between the predicted surface and the actual surface if the object surface is composed of circular curves or linear curves. Therefore we can accurately place the vision sensor for the next view pose to perceive maximum information. Example objects include cuboids and balls at arbitrary poses, and cylinders and cones at restricted poses.

The typical 2nd order curves are parabola: $z = A_0 x^2$,

$$\text{elliptic: } z = \frac{1}{B_0} \sqrt{1 - A_0^2 x^2},$$

$$\text{and hyperbola: } z = \frac{1}{B_0} \sqrt{1 + A_0^2 x^2}.$$

Since (11) is non-integrable when $a \neq 0$ and $b \neq 0$, for these three curves, a 2nd order Taylor polynomial function is used to approximate it. That is:

$$z_p(x + \Delta) = z_p(x) + z'_p(x)\Delta + \frac{1}{2} z''_p(x)\Delta^2 + R_3\Delta^3,$$

where,

$$z'_p = \frac{(ax^2 + 2bx + 2C_1)}{\sqrt{4 - (ax^2 + 2bx + 2C_1)^2}}, \text{ and}$$

$$z''_p = \frac{8(ax + b)}{[4 - (ax^2 + 2bx + 2C_1)^2]^{3/2}}.$$

In practical implementation, the term $R_3\Delta^3$ is neglected as $\Delta \ll 1$ and $R_3\Delta^3 \approx 0$. Therefore the predicted curve can be calculated numerically. The variables a and b are determined according to (11) and (12). The constant C_1 is also determined with the boundary condition $z_p'(x_2) = z'(x_2)$, i.e.,

$$C_1 = \frac{z'(x_2)}{\sqrt{[z'(x_2)]^2 + 1}} - \frac{1}{2} ax_2^2 - bx_2. \quad (26)$$

Then the absolute errors and relative errors of these curves are determined by averaging the difference between the actual z values and predicted z values. Simulation has been performed to observe the results of the absolute and relative errors of a parabola, elliptic, hyperbola, and a circle for a , b , C_1 , assuming $A_0=1$ and $B_0=0.5$. As a verification of the algorithm, the results show that the prediction error is always zero if the actual curve is a part of a circle, with the results of $a = 0$, $b = 1$, $C_1 = 0$, and $E_{\text{circle}} = 0$. The results also showed that the errors were very small (relative error was at the order of 10^{-7}) for predicting other 2nd order curves. Therefore we can accurately place the vision sensor for the next view pose to observe objects composed of such surfaces, when using the

expected curve predicted by the uniform surface trend model (15). However, for each different curve, the constants (a , b , and C_1) are different and they should be dynamically determined according to the known part of the curves.

VI. EXPERIMENTS

A. Implementation Considerations

To implement the proposed method in a practical system, we have to consider many other factors and conditions. In fact, this method has to be integrated with other techniques and algorithms for automating the modeling process, as illustrated in Fig. 5. This condition makes difficult in evaluation of the proposed method and currently we can only provide some simple examples with a little human interference during the modeling process.

One problem has to be noted is the noise filtering. Since the curvature on the object surface is sensitive to noise and the local features of a 3D image do not affect the surface trend much, a low-pass filter is applied to the image so that we can obtain a smoothed surface. Therefore, for the trend which describes the shape of a considerable large area, it is actually not sensitive to noise. We are also considering some image processing methods to filter noise and compute the shape in a more stable way.

Equations (11) and (12) are described as continuous functions. When applied to digital image processing, they can be written as,

$$a = \frac{m \sum_{i=1}^m x_i k_i - \sum_{i=1}^m x_i \sum_{i=1}^m k_i}{m \sum_{i=1}^m x_i^2 - (\sum_{i=1}^m x_i)^2}, \quad (11')$$

$$b = (\sum_{i=1}^m k_i - a \sum_{i=1}^m x_i) / m. \quad (12')$$

where m is the number of total points on the known part of the object surface.

Another problem is on calculation of curvatures. The curvature of each point on the known part of object surface should be calculated so that the surface trend can be determined for predicting the unknown part. Equation (6) is not suitable for computing the curvature on a digital image because the errors in computing z' and z'' will be significant. In this research, we determine the curvature of a point $P(x(i), z(i))$ by using three adjacent points (triplet), i.e. $P(x(i-1), z(i-1))$, $P(x(i), z(i))$, and $P(x(i+1), z(i+1))$. Each triplet defines a circle and the curvature $k(i)$ is the inverse of its radius.

B. Practical Examples

Several experiments were carried in our laboratory for construction of object models. The range data are obtained by

a structured light system set up in this research, which mainly consists of a projector and a camera. The projector is a palm-sized digital projector, PLUS U3-1080. It is connected to a computer and is controlled to generate some gray-encoded stripe-light patterns for 3D reconstruction [28]. The CCD camera (PULNIX TMC-9700) has a 1-inch sensor and a 25 mm lens. A fixture is designed for mounting the structured light system on the end-effector of a 6DOF robot (STAUBLI RX-90B) with ± 0.02 mm repeatability. This enables the 3D sensor to be freely moved to an arbitrary position in the workspace.



Figure 6. The objects to be reconstructed

Fig. 6 illustrates two objects for the model construction in the experiment. In both cases, we set the resolution constraint to be $r_{\max}=0.85$ mm. The first one is demonstrated with the procedure in more details here. It was incrementally built by four views. The first view is assumed to be taken from the top view. To determine a next view for acquiring some unseen information of the object, we used trend surface method and developed a program to compute the expected surface curves. Then the trend is computed and the next viewpoint is determined.

The experimental results in the incremental construction of the first object are shown below to illustrate the computation at each step. A new surface was acquired at each view and it was integrated with the existing ones to form a partial model. The exploration direction and sensor placement were determined by the proposed method. The placement parameters of each view are set with a viewpoint vector in the format of $[x \ y \ z \ ai \ bj \ ck]$, representing the six parameters of the 3D position and orientation. Also a registration vector contains six parameters for the surface integration so that the range images are transformed and registered to a common coordinate system. The first three of the six parameters define the surface's reference point and the other three define the surface orientation by 3-axis rotation. In this way, the model is improved by eliminating the overlapped/redundant points and stitching non-data areas.

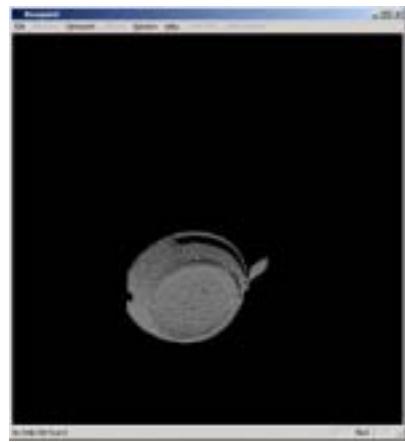


Figure 7. The 3D surface obtained from the first view

The corresponding placement parameters of the first view (Fig. 7) are

$$\text{Viewpoint}_1 = (0, 0, 457.9333, 0, 0, -1)$$

$$\text{Reg}_1 = \mathbf{O}_{\text{global}} = (0, 0, 0, 0, 0, 0)$$

where the viewpoint vector has a format of $[x \ y \ z \ ai \ bj \ ck]$, representing the six parameters of the 3D position and orientation. Reg_i contains some registration parameters for the surface integration so that the range images are transformed and registered to a common coordinate system. The first three of the six parameters define the surface's reference point and the other three define the surface orientation by 3-axis rotation. In this way, the model is improved by eliminating the overlapped/redundant points and stitching non-data areas.

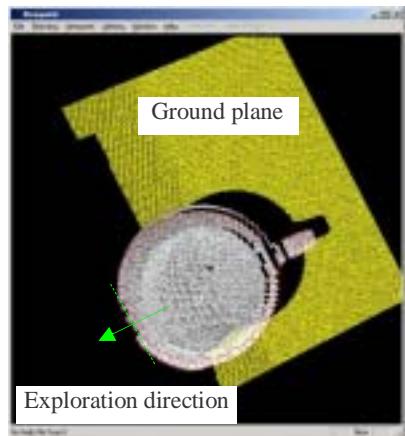


Figure 8. The points were detected as the candidates of exploration directions. The decision was made by their rating values.

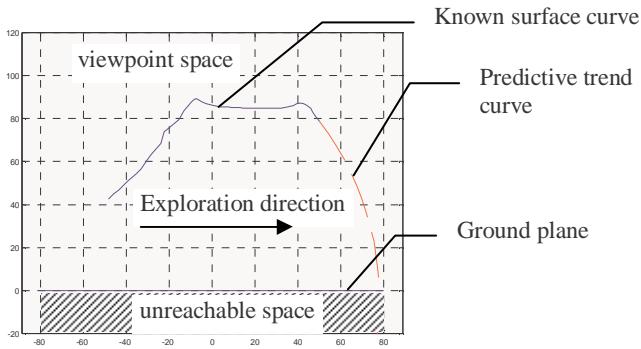


Figure 9. An predictive trend curve

Along the exploration direction, a trend curve (the dashed red curve in Fig. 9) is computed from the known 3D surface. The space under the curve and ground plane is marked as unreachable.

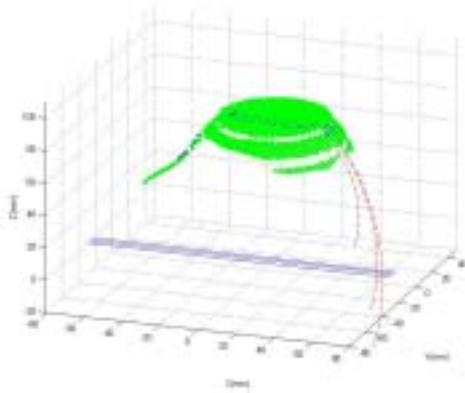


Figure 10. Several trend curves are obtained to form a trend surface so that the decision of a next viewpoint will be made more reliable

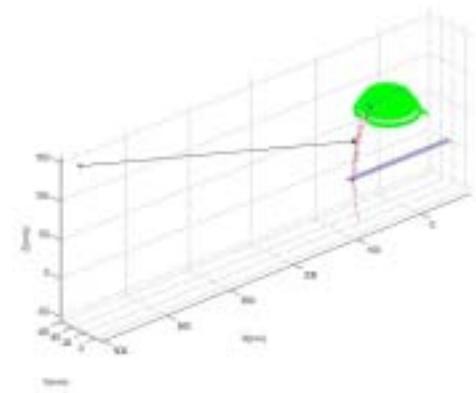


Figure 11. The planned next viewpoint

The sensor position and looking direction are planned according the predicted curves. The corresponding placement parameters are:

`Set_scene_center = [63.9950, -23.0346, 40.4942].`

`Viewpoint2 = (502.9636, -26.2364, 158.2319, -0.965838, 0.00704474, -0.259052).`

`Reg2 = (-72.06, 0.00, 40.10, 32.01, -89.89, -30.80).`

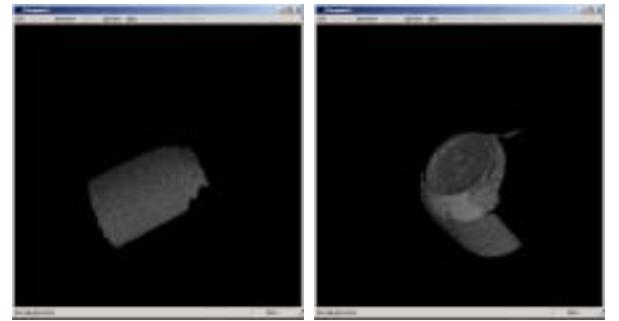


Figure 12. The 3D surface was obtained from the second view (left) and integrated with the first view to form a partial model (right) of the object.

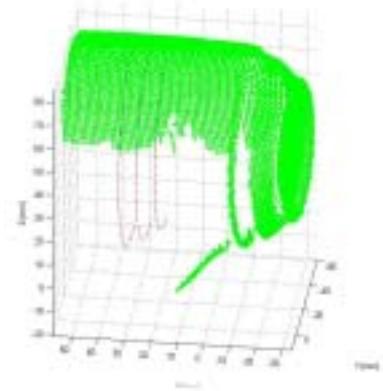


Figure 13. Trend curves in the second step

Similar to the situation in the first step, candidate points are selected to find an exploration direction. Along this direction, several trend curves are computed to predict the unknown part of the surface.

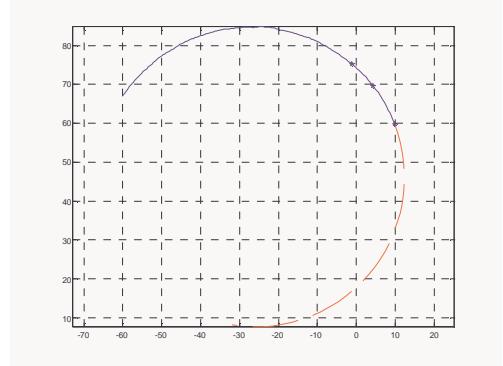


Figure 14. One of the trend curve as in Fig. 13. Since the original curve is circular, the prediction is very accurate.

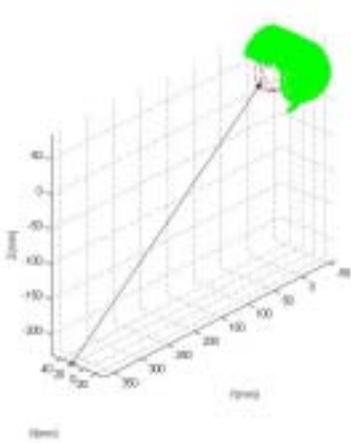


Figure 15. The next viewpoint

The sensor pose is also determined according the predicted curves. The corresponding placement parameters are:

Set_scene_center = [-8.4859 60.0199 33.7807]
Viewpoint3 = (21.7862, 366.7046, -228.8904, -0.0747589,
-0.757378, 0.648683)
Reg3 = (30.21,-116.07,40.02,89.83,31.10,-89.92)

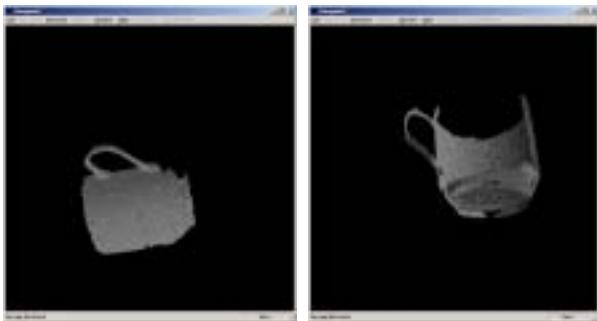


Figure 16. The 3D surface was obtained from the third view and integrated with all known views.

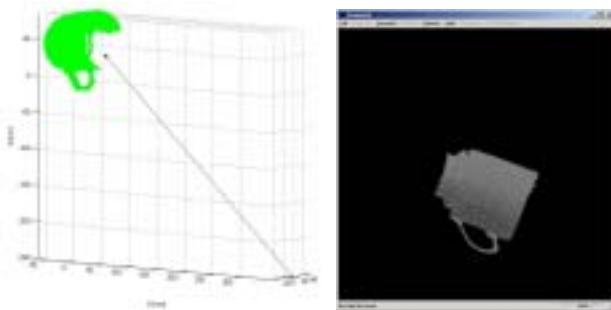


Figure 17. The next viewpoint and 3D surface obtained

With the model in Fig. 16, it is necessary to take further viewpoint decision and surface acquisition. These were done similarly to the previous steps. The sensor parameters for the fourth view are determined as (Fig. 17):

Set_scene_center = [16.3405, -10.1124, 35.0603]

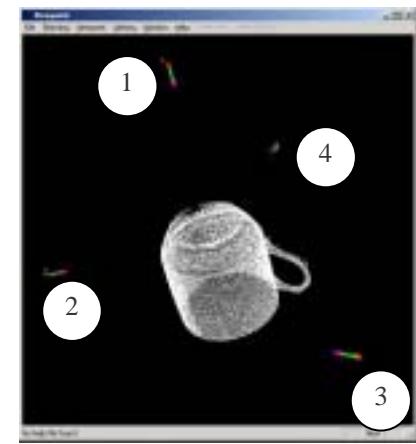


Figure 18. The complete model and planned viewpoints

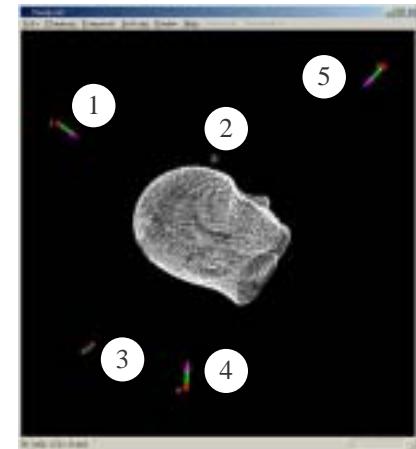


Figure 19. The model reconstructed from the second object

Finally the complete model was obtained by integrating all the four views. Their relative distribution to the object model in 3D space is given in Fig. 18. Also Fig. 19 shows the other object with five viewpoints. 3D Animation of the results is available on our web: <http://vision.research.sychen.com/>. In our experiments, the implementation was performed on a PC with 750 MHz CPU. The 3D surface acquisition was achieved by a unique color encoding method. The computation time for planning a next viewpoint was about three to five seconds. It should be noted that the planning results are dependent on the first view. By different initial viewpoints, the results are also different and there may be one or two more views planned for each object.

C. Discussion

For surface edges and object boundaries, as the curvature around these areas changes abruptly, there will be significant errors in computing the surface trend. For example, if there is surface edge located in the seen area, the domain of known part for determining constants a and b should be restricted to a smaller area that does not contain any edges. The exploration direction may also need to be changed by analyzing the seen object surface.

When modeling a small-sized object (compared with the sensor's field-of-view), the trend surface method may not work very well since the whole object will be contained in a single view and the surface trend has a high order. That makes it unreliable to predict the unknown area. In such cases, a sensor based solution [8] instead of target-driven method may be used for the modeling task.

The basic idea of this research is to find any possible cues of object shape for predicting unknown areas. The surface trend is an option that is suitable for many objects. Here, the trend may not be computed by all area of the known part. We only choose a suitable surface part usually without object boundary or edges inside, so that the trend is reasonable and predictable. For example, for a polyhedral object, the surface trend will be a single surface plane. Only when the image boundary is on an edge of polygonal plane, the next view can not be reasonably planned. This will certainly cause a not good viewpoint, but rarely often happen in practice. Many cases are avoided by our selection method of exploration direction. Of course, we don't expect only trend method is enough for completion of the whole modeling task. We have to integrate many other methods together so that the whole modeling process is executed in an "adaptive" way. That is also why we currently have difficulties to provide some complicated examples that are fully automatically performed.

VII. CONCLUSION

In this paper, we proposed a trend surface model which was used to predict the unseen part of objects or environments. In this way, the next viewpoint can be determined for on-line acquisition of the range data until the whole structure of the object or environment is reconstructed.

The trend surfaces and trend curves were computed from the curvature tendency. While determining the next viewpoint, multiple sensor placement constraints were considered in this paper, such as resolution, field-of-view, and viewing angle. Such a trend model can accurately predict the unknown surface of the object if the surface is composed of 1st-order and 2nd-order curves and surfaces. Experiments were carried out to demonstrate the method presented in this paper.

Our future work in the research will deal with reliable detection of the boundary positions on a complex partial model, so that they will be evaluated for selecting a best candidate of exploration direction. Other uncertain conditions will also be considered to make to the reconstruction process

more reliable so that an autonomous robot system can work without any human interference.

REFERENCES

- [1] P. K. Allen, I. Stamos, etc., "3D Modeling of Historic Sites using Range and Image Data", ICRA 2003, Sep. 14-19, 2003 Taipei, pp. 145-150.
- [2] C. K. Cowan and P. D. Kovesi, "Automatic sensor placement from vision task requirements", IEEE Trans. on PAMI, Vol. 10, No. 3, May 1988, pp. 407-416.
- [3] S. Y. Chen and Y. F. Li, "Automatic Sensor Placement for Model-Based Robot Vision", IEEE Trans. on Systems, Man and Cybernetics, Part B, Vol. 33, No. 1, 2004, pp. 393-408.
- [4] D. Papadopoulos-Orfanos and F. Schmitt, "Automatic 3-D digitization using a laser rangefinder with a small field of view", Int. Conf. on Recent Advances in 3-D Digital Imaging and Modeling, Ottawa, Ontario, Canada, May 1997, pp. 60-67.
- [5] K. N. Kutulakos and C.R. Dyer, "Global surface reconstruction by purposive control of observer motion", IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, Seattle, 1994, pp. 331-338.
- [6] J. Maver and R. Bajcsy, "Occlusions as a guide for planning the next view", IEEE Trans. on PAMI, Vol. 15, No. 5, May 1993, pp. 417-433.
- [7] P. Whaite and F.P. Ferrie, "Autonomous exploration: driven by uncertainty", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 19, No.3, March 1997, pp. 193-205.
- [8] R. Pito, "A solution to the next best view problem for automated surface acquisition", IEEE Trans. on PAMI, Vol. 21 (10), 1999, pp. 1016-30.
- [9] Y. F. Li and Z. Liu, Uncertainty-Driven Viewpoint Planning for 3D Object Measurements, Proc. IEEE International Conference on Robotics and Automation, Taipei, Taiwan, Sept. 2003, pp.127-132.
- [10] H. Zha, K. Morooka, T. Hasegawa, T. Nagata, "Active modeling of 3-D objects: planning on the next best pose (NBP) for acquiring range images", Int. Conf. on Recent Advances in 3-D Digital Imaging and Modeling, Ottawa, Ontario, Canada, May 1997, pp. 68-75.
- [11] Z. Yu, K. Wang and R. G. Yang, "Next best view of range sensor", IEEE 22nd Int. Conf. on Industrial Electronics, Control, and Instrumentation, vol. 1, Taipei, Taiwan, 1996, pp. 185-188.
- [12] R. Pito, "A sensor-based solution to the "next best view" problem", 13th Int. Conf. on Pattern Recognition, vol. 1, Vienna, Austria, Aug. 1996, pp. 941-945.
- [13] T. Arbel and F. P. Ferrie, "Viewpoint selection by navigation through entropy maps", Seventh IEEE Int. Conf. on Computer Vision, vol. 1, Kerkyra, Greece, 1999, pp. 248-254.
- [14] J. E. Banta and M.A. Abidi, "Autonomous placement of a range sensor for acquisition of optimal 3-D models", 22nd IEEE Int. Conf. on Industrial Electronics, Control, and Instrumentation, Taipei, Taiwan, Aug. 1996, vol. 3, pp. 1583-1588.
- [15] M. Reed, P. Allen, "Constraint-based sensor planning for scene modeling", IEEE Int. Symposium on Computational Intelligence in Robotics and Automation, Monterey, California, 1999, pp. 131-136.
- [16] E. Marchand and F. Chaumette, "Active sensor placement for complete scene reconstruction and exploration", IEEE Int. Conf. Robotics and Automation, Albuquerque, USA, April 1997, pp. 743-750.
- [17] E. Marchand and F. Chaumette, "Active vision for complete scene reconstruction and exploration", IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 21, no. 1, Jan. 1999, pp. 65-72.
- [18] K. A. Tarabanis, R. Y. Tsai, P. K. Allen, "The MVP sensor planning system for robotic vision tasks", IEEE Trans. on Robotics and Automation, Vol. 11, No. 1, 1995, pp. 72-85.
- [19] B. Triggs, C. Laugier, "Automatic camera placement for robot vision tasks", IEEE Int. Conf. on Robotics and Automation, Nagoya, 1995, pp. 1732-1737.
- [20] X. G. Gu, M. M. Marefat, F. W. Ciarallo, "A robust approach for sensor placement in automated vision dimensional inspection", IEEE Int. Conf. on Robotics and Automation, Detroit, May 1999, pp. 2602-2606.

- [21] C. K. Cowan, A. Bergman, "Determining the camera and light source location for a visual task", IEEE Int. Conf. on RA, vol. 1, Scottsdale, 1989, pp. 509-514.
- [22] K. A. Tarabanis, P. K. Allen, R. Y. Tsai; "A survey of sensor planning in computer vision", IEEE Trans. on Robotics and Automation, Vol. 11, No.1, 1995, pp. 86-104.
- [23] E. Trucco, M. Umasuthan, A. M. Wallace, V. Roberto, "Model-based planning of optimal sensor placements for inspection", IEEE Trans. on RA, Vol. 13, No.2, 1997, pp. 182-194.
- [24] K. Morooka, H. Zha, T. Hasegawa, "Computations on a spherical view space for efficient planning of viewpoints in 3-D object modeling", 2nd Int. Conf. on 3-D Digital Imaging and Modeling, Ottawa, Canada, 1999, pp. 138-147.
- [25] L. M. Wong, C. Dumont, M. A. Abidi, "Next best view system in a 3-D object modeling task", IEEE Int. Symposium on Computational Intelligence in Robotics and Automation, Monterey, California, 1999, pp. 306 -311.
- [26] W. B. Agocs, "Least squares residual anomaly determination", Geophysics, Vol. 16, 1951, pp. 686-696.
- [27] W. C. Krumbein, "The 'sorting out' of geological variables illustrated by regression analysis of factors controlling beach firmness", J. of Sedimentary Petrology, Vol. 29, 1959, pp. 575-587.
- [28] Li, Y. F. and Chen, S. Y., "Automatic Recalibration of an Active Structured Light Vision System", IEEE Trans. on Robotics and Automation, Vol. 19, No.2, 2003, pp. 259-268.
- [29] S. Y. Chen and Y. F. Li, "Active Viewpoint Planning for Model Construction", IEEE International Conference on Robotics and Automation, New Orleans, LA, USA, April, 2004, pp. 4411-4416.
- [30] M. K. Reed, P. K. Allen, I. Stamos, "3-D modeling from range imagery: an incremental method with a planning component", Int. Conf. on Recent Advances in 3-D Digital Imaging and Modeling, Ottawa, Ontario, Canada, May 1997, pp. 76-83.
- [31] M. Lozano, M. Devy, J.M. Sanchez, "Perception Planning for an Exploration Task of a 3D Environment", 16th Int. Conf. on Pattern Recognition (ICPR 2002), Vol. III, Québec City, Canada, August 2002, pp. 704-707.
- [32] W.R Scott, G. Roth, and J.F. Rivest, View planning for automated three-dimensional object reconstruction and inspection. ACM Computing Surveys, Vol. 35, 2003, pp.64-96.
- [33] A. Georgiev and P. K. Allen, "Localization Methods for a Mobile Robot in Urban Environments", IEEE Trans. on Robotics and Automation, V. 20, N. 5, Oct. 2004, pp. 851-864.

S. Y. Chen (M'01) received the Ph.D. degree from the Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong, Hong Kong, in 2003.

He joined Zhejiang University of Technology, China, in Jan. 2004 where he is currently an Associate Professor in the Department of Information Engineering. Since July 2004, he has been invited as a guest researcher in the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His research interests include computer vision, robotics, 3-D object modeling, and medical image analysis.

Y. F. Li (M'91-SM'01) received the Ph.D. degree from the Robotics Research Group, Department of Engineering Science, University of Oxford, Oxford, U.K., in 1993.

From 1993 to 1995, he was a Postdoctoral Research Associate in the Artificial Intelligence and Robotics Research Group, Department of Computer Science, University of Wales, Aberystwyth, U.K. He joined City University of Hong Kong, Hong Kong, in 1995 where he is currently an Associate Professor in the Department of Manufacturing Engineering and Engineering Management. His research interests include robot vision, sensing, and sensor-based control for robotics.