



0031-3203(95)00076-3

ERROR ANALYSIS OF 3D SHAPE CONSTRUCTION FROM STRUCTURED LIGHTING*

ZAIMING YANG and Y. F. WANG†

Department of Computer Science, University of California, Santa Barbara, CA 93106, U.S.A.

(Received 29 December 1993; in revised form 10 May 1995; received for publication 2 June 1995)

Abstract—In this paper, we present a detailed model and analysis of several error sources and their effects on measuring three-dimensional (3D) surface properties using the structured lighting technique. The analysis is based on a general system configuration and identifies three types of error sources—system modeling error, image processing error and experimental error. Absolute and relative error bounds in obtaining 3D surface orientation and curvature measurements using structured lighting are derived in terms of the system parameters and likely error sources. In addition to the quantization error, other likely error sources in system modeling and experimental setup are also considered. Even though our analysis is on structured lighting, the results are readily applicable to other triangulation-based techniques such as stereopsis. Finally, our analysis focuses on error in inferring surface orientation and principal surface curvature. Such analyses, to our knowledge, have never been attempted before.

Image processing

Structured light

Orientation

Curvature

Error analysis

1. INTRODUCTION

The problem of reconstructing 3D surface structures from their 2D projections is an important research topic in computer vision. Over the past two decades, a variety of techniques have been developed to infer 3D surface structures from 2D images using different imaging devices, shape cues and heuristics.^(1–3) These techniques can rely on ambient light reflection (passive sensing) or can employ a light source to actively probe the environment (active sensing). They have also relied on many image shape cues—such as stereo disparity, image brightness and surface pattern—to recover the depth, orientation and curvature of an imaged surface.

To study the feasibility of these 3D shape reconstruction techniques in industrial applications, it is imperative that their accuracy be understood. That is, for each technique, rigid modeling and analysis of the inherent error sources and their effects on 3D shape inference are needed. However, error analysis of all types of sensors used in machine vision is beyond the scope of this paper. Our discussion will be limited to the structured light-sensing technique, which we have some experience with. Thus, the goal of this paper is to identify likely error sources and investigate their effects on computing surface properties using the structured light sensing technique. More precisely, errors in using structured lighting to infer surface orientation and principal surface curvatures are analysed. Error

bounds are derived in terms of various system parameters and error sources. Simulation was conducted to verify the correctness of the analysis.

Structured lighting is an active sensing technique which projects a spatially modulated pattern to encode the image object for analysis.^(4–15) Traditional structured light techniques use a grid pattern and rely on the triangulation principle in the analysis, i.e. the observed pattern is matched with the projected one, and the 3D position of the pattern—hence that of the encoded surface—is recovered through triangulation. More recently, it was shown that it is possible to analyse the orientation and curvature of the projected pattern to infer the orientation and curvature of the encoded surfaces.^(10,11,13,14)

Although many image analysis techniques have been developed using structured lighting, no formal modeling and analysis of errors in structured lighting have been attempted, except, maybe, for references (16, 17). Frobin^(16,17) considered the image processing error in his reconstruction equation and modeled such an error as uncorrelated Gaussian noise at each pixel location. He then computed the 3D surface position using the least-squares minimization, with sensor data weighted by the inverse of the observation error. However, Frobin's analysis was on computing 3D surface position using structured lighting and the analysis results are not applicable to the surface orientation and curvature computations.

Other researches on error analysis in computer vision were mainly concerned with the analysis of the stereopsis technique and considered only the quantization error.^(18–22) Duda and Hart⁽²³⁾ gave a brief treat-

* This research was supported in part by a grant from the National Science Foundation, IRI-8908627.

† Author for correspondence.

ment on the subject. McVey and Lee⁽²⁰⁾ performed a worst-case error analysis of the stereopsis technique, so did Solina.⁽²⁶⁾ Matthies and Shafer⁽¹⁹⁾ used 3D Gaussian distribution to model the triangulation error in the stereo vision. Blostein and Huang⁽¹⁸⁾ used a uniform distribution error model and derived the probability profile of the absolute positional error. Rodriguez and Aggarwal⁽²¹⁾ used an approach similar to that of in reference (18) but gave a formulation in terms of the relative error.

All the above analyses were based on a simplified stereo configuration, in which the lines of sight of the two cameras were parallel. Verri and Torre⁽²²⁾ assumed a slightly more general configuration and allowed independent coordinate systems to be associated with the left and right cameras. However, their analysis results were applicable only to the plane in the scene which was of the same distance to the origins of the left and right coordinate systems.

In contrast to the above, our analysis assumes a general system configuration, and in addition to the quantization error, other likely error sources in system modeling and experimental setup are also considered. Although our analysis is on structured lighting, the results are readily applicable to other triangulation-based techniques such as the stereopsis. Furthermore, our analysis focuses on error in inferring surface orientation and principal surface curvature; such analyses, to our knowledge, have never been attempted before.

The remainder of this paper is organized as follows: Section 2 identifies likely error sources in the structured light-sensing technique. Sections 3 and 4 analyse the error in inferring 3D surface orientation and principal surface curvatures. Section 5 presents some simulation results. Finally, Section 6 presents concluding remarks.

2. ERROR SOURCES

As shown in Fig. 1, in the structured light technique, one of the cameras in a passive stereo setup is replaced with a projection device. A grid pattern marked on a slide is projected to "encode" the imaged object's surface for analysis. The perceived pattern may comprise k sets of parallel stripes, $k \geq 1$. The encoded scene is recorded using a camera. The perceived surface pattern in a structured light image can be used to compute three types of surface properties: the 3D position, orientation and principle surface curvatures of the imaged surfaces.

We will use the following notational conventions throughout the discussion: We denote the computed value of a surface measurement as x^* , where x can be the depth, orientation or curvature of the imaged surface. It can also be used to denote the computed value of some intermediate parameter in recovering a desired surface property. The corresponding theoretical values (or the ground truth) of x^* is denoted as

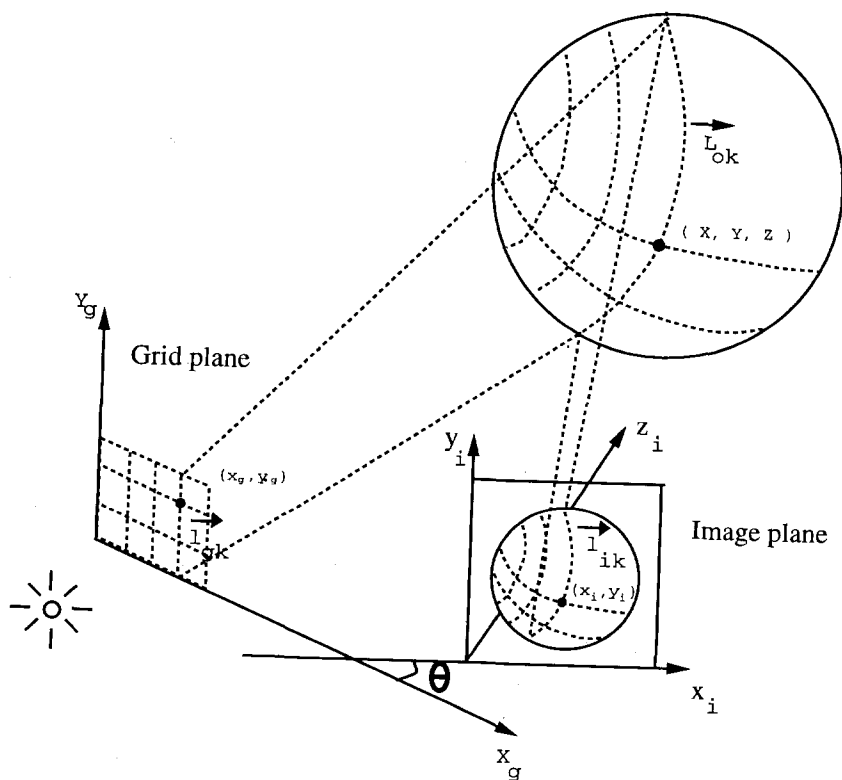


Fig. 1. System configuration.

x (without an asterisk). We use the Greek letter Δ as the prefix for the absolute error and the Greek letter δ as the prefix for the relative error. That is, $\Delta x = |x^* - x|$ and $\delta x = |(x^* - x)/x|$.

In our analysis, we first identify likely sources of error and establish their theoretical error upper bounds. The error bounds are then used in the analysis in Sections 3 and 4. We identify three types of error sources: system modeling error, image processing error and experimental error, which are discussed below.

2.1. System modeling error

An optical system is usually characterized by one of the following two models: perspective projection or parallel projection. The former is more accurate (it produces the correct projection foreshortening effect: objects closer to the camera appear larger in an image), but requires a division in computing the 2D projection coordinates. The parallel projection model may be assumed in a shape inference process to simplify certain analyses. For example, by adopting a parallel projection model, we were able to avoid solving the difficult feature correspondence problem in our analysis of the structured lighting techniques.^(10,11,13,14) However, error introduced by using a less faithful model need be considered, because, in the real world, projectors and cameras do not behave according to an ideal parallel projection model where all the projection rays are parallel. Rather, a projection ray makes an angle φ with respect to the optical axis:

$$\tan|\varphi| = \frac{|L|}{D} = \frac{|L|}{d} \approx \frac{|L|}{f}, \quad (1)$$

where $|L|$ denotes the size of the imaged object and $|l|$ that of its image; f is the focal length of the lens; d is the distance from the slide or the film to the lens (image distance); D is, in the case of a slide projector, the distance from the lens to the projection screen and, in the case of a video camera, the distance from the lens to the imaged object (object distance). In parallel projection, it is assumed that $f \gg |l|$ and φ approaches zero. Hence, rays emitted from the projector or incident on the film all become parallel.

In reality, φ is never exactly zero. Approximating a nonzero φ by zero introduces error in surface measurements. This error is defined to be the system modeling error and is a function of the angle φ . The theoretical error upper bound is therefore:

$$\Phi = \max|\varphi - 0| = \max|\varphi| \approx \tan^{-1} \frac{W}{2f}, \quad (2)$$

where W is the diagonal length of, in the case of a projector, the slide, and, in the case of a camera, the film.

2.2. Image processing error

The analysis of a structured light image depends heavily on the 2D features—such as positions of stripe junctions and orientations and curvatures of projected

stripes—extracted from images. This feature extraction process suffers from an image processing error that consists of two separate error sources: error due to quantization and error due to mislocating features in the image plane.

An image coordinate can differ from its true value because the quantization process restricts image pixels to lie on an integer grid. It was shown in references (18, 21) that the quantization error can be at most half a pixel, $\pm \eta/2$, in both the x and y directions, where η denotes the size of a pixel. Then it is trivially shown that the total quantization error, e_q , is bounded above by $\eta/\sqrt{2}$.

Another usually more critical error is that image features might not be properly located. For example, in a typical structured light image, the observed pattern may not be sharply focused. Hence, the observed stripes may be wider than a pixel. Referring to Fig. 2, w_g denotes the width of a stripe in the projection slide, w_o denotes the width of the stripe projected on the imaged object's surface, and w_i denotes the width of the perceived stripe in the image plane. Let θ_g be the angle between the object surface and the light sheet from the projector and θ_i the angle between the object surface and the viewing direction, then the following relations hold:

$$\frac{d_g w_g}{f_g} = w_o \cos\left(\theta_g - \frac{\pi}{2}\right), \quad \frac{d_i w_i}{f_i} = w_o \cos\left(\theta_i - \frac{\pi}{2}\right),$$

therefore, dividing the above two expressions we have:

$$\frac{d_i w_i f_g}{f_i d_g w_g} = \frac{\cos(\theta_i - \pi/2)}{\cos(\theta_g - \pi/2)},$$

or

$$w_i = \frac{d_g f_i \cos(\theta_i - (\pi/2))}{d_i f_g \cos(\theta_g - (\pi/2))} w_g = \frac{d_g f_i \sin \theta_i}{d_i f_g \sin \theta_g} w_g, \quad (3)$$

where d_g and d_i are the distances from the projector and the camera to the object, respectively.

The width w_i of a projected stripe in the image plane is computed using equation (3). When w_i is more than one pixel wide, we select the coordinate of a pixel which lies under the perceived stripe and report that coordinate as the position of the stripe in the image plane. Such a selection may not be unique as there may be as many as $\lceil w_i/\eta \rceil$ possible choices and an improper selection will misplace the stripe. This misplacement error is denoted as e_f or the feature location error, which, in general, can be larger than the quantization error. The image processing error is defined as:

$$e_r = e_q + e_f. \quad (4)$$

It remains to be shown how large the error in locating image features (e_f) can become. Here, we claim that for a projected stripe of n pixels wide, $n \geq 1$, the feature location error is minimized if the coordinate of the projected stripe's central pixel is used. We introduce the following proposition (Proof of the proposition can be found in the Appendix):

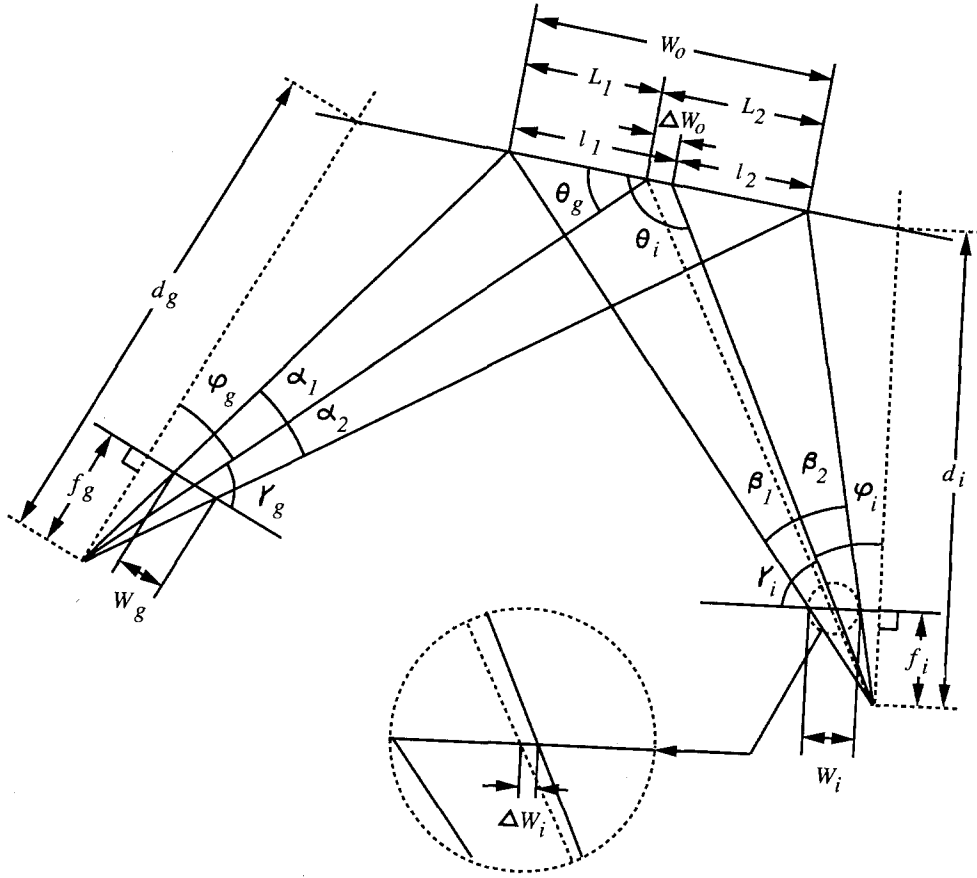


Fig. 2. Feature location error.

Proposition 1. (feature location error). If points on the central line along a perceived stripe are selected and reported as the position of the stripe in the image plane, then the feature location error introduced is:

$$e_f = |x^* - x| \leq \frac{d_g f_i \max(\tan \alpha, \tan \beta)}{2d_i f_g \sin^2(\theta_g)} w_g, \quad (5)$$

where x denotes the (true) coordinate of a point on the projected curve and x^* is the coordinate of the approximation point, i.e. the center pixel; w_g is the stripe width in the projection slide, 2α equals the projection angle and 2β equals the viewing angle (Fig. 2).

The validity of Proposition 1 can be intuitively verified as follows: When all the variables in equation (5) are held constant while varying, say, f_i , it can be seen from Fig. 2 that w_i increases as f_i increases. Hence, feature location error increases as a result of increasing f_i . Similarly, increasing d_g while holding all other variables constant results in an increase of w_o , and hence, an increase of w_i and the feature location error (Fig. 2). Increasing d_i and f_g has the opposite effect of decreasing the size of w_i , and hence the feature location error decreases. Therefore, we conclude that a smaller

image processing error results if the grid marks in the projection slide are thin (i.e. w_g is small) and/or the projector is positioned in such a way that the projection direction is close to the normal direction to the imaged object surface (i.e. $\theta_g \approx \pi/2$).

2.3. Experimental error

We define errors, other than those introduced in system modeling and image processing, as experimental errors. In a structured light experiment, various types of errors may occur because of inappropriate operation and calibration of the imaging equipments. For example, if the patterned slide is not positioned level to the ground, orientation of the stripes in the pattern may be off by a few degrees from the calibrated value. The parameters of the imaging devices may not be absolutely accurate either, and the focal length of the camera lens may differ slightly from the marked value. Other examples of experimental errors include: numerical errors resulted from truncation and numerical approximation, failure to accurately measure the relative position of the projector and the camera, distortion of the grid pattern, lens distortion, etc. These errors vary from one experiment to the next, depend on

the accuracy of the imaging equipments and may also depend on the experience level of the experimenter. We do not consider this type of error in our analysis, except for that stemmed from numerical methods which can be characterized.

3. ERROR ANALYSIS OF ORIENTATION MEASUREMENT

Since inferring 3D surface position using structured lighting relies on the same triangulation principle employed in stereopsis, and error analysis of stereopsis has been addressed by various researchers,^(18–22) we will not present a duplicate analysis here. Rather, we will concentrate our discussion on analysing error in inferring 3D surface orientation and principle surface curvature using structured lighting. Such analyses, to our knowledge, have not been attempted before. In the main body of the paper we will go through the analysis step-by-step and state various propositions without proving them. Proofs of the propositions used in the analysis can be found in the Appendix.

3.1. Surface orientation inference technique

The process of computing the orientation of an imaged surface from structured lighting is illustrated in Fig. 3 using a planar surface patch. If the stripe orientations \mathbf{L}_{o1} and \mathbf{L}_{o2} —where 1 and 2 denote the first and second sets of projection stripes—on a planar surface patch, \mathbf{P}_o , are recovered, then the surface normal direction \mathbf{P}_o can be readily computed by taking the cross product of \mathbf{L}_{o1} and \mathbf{L}_{o2} :

$$\mathbf{P}_o = (P_{ox}, P_{oy}, P_{oz}) = \frac{\mathbf{L}_{o1} \times \mathbf{L}_{o2}}{|\mathbf{L}_{o1} \times \mathbf{L}_{o2}|}. \quad (6)$$

Hence, surface orientation inference from structured lighting is to locate a pair of projected stripes on the imaged object's surface and use them to compute \mathbf{L}_{o1} and \mathbf{L}_{o2} . Referring to Fig. 3 again, in a structured light image, let the stripe orientations observed in the image plane \mathbf{P}_i be \mathbf{l}_{i1} and \mathbf{l}_{i2} , and those in the grid projection plane \mathbf{P}_g be \mathbf{l}_{g1} and \mathbf{l}_{g2} . Using a parallel projection model, the stripe orientations \mathbf{L}_{o1} and \mathbf{L}_{o2} on the imaged object surface can be computed using the following equations:

$$\begin{aligned} \mathbf{L}_{o1} &= (\mathbf{P}_g \times \mathbf{l}_{g1}) \times (\mathbf{P}_i \times \mathbf{l}_{i1}) \\ \mathbf{L}_{o2} &= (\mathbf{P}_g \times \mathbf{l}_{g2}) \times (\mathbf{P}_i \times \mathbf{l}_{i2}). \end{aligned} \quad (7)$$

If the imaging configuration has been calibrated, \mathbf{P}_g is known, so are \mathbf{l}_{g1} , \mathbf{l}_{g2} and \mathbf{P}_i . Furthermore, \mathbf{l}_{i1} and \mathbf{l}_{i2} can be measured in the image plane. Hence, \mathbf{L}_{o1} , \mathbf{L}_{o2} , and \mathbf{P}_o can be computed using equations (6) and (7). The process to recover the orientation of a curved surface is similar to that for a planar surface and is described in more details in references (10, 11).

To compute visible surface orientation using the technique discussed above, three vector products are carried out in equations (6) and (7) based on the observed pattern orientation in the image plane. Therefore, in our analysis, we first study the error in

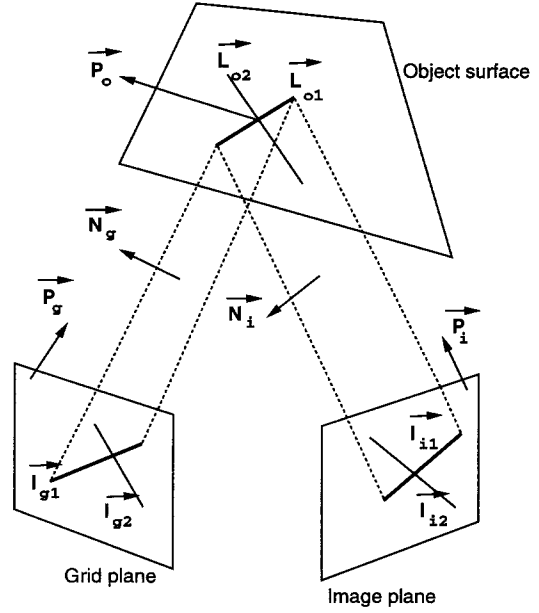


Fig. 3. Surface orientation computation.

computing the tangential direction of a projected stripe (Section 3.2). We then show how this error propagates through the vector product computation (Section 3.3). Finally, we combine the analysis results presented in Section 3.2 and 3.3 to analyse the error in recovering visible surface orientation using structured lighting (Section 3.4).

3.2. Error in computing stripe tangential direction

In the following analysis, we assume that n sampling points (x_k, y_k) , $k = 1, \dots, n$, are located along a projected stripe in the image plane, and the chord formed by connecting any three adjacent points are not vertical (i.e. any three adjacent points have different x coordinates). If a chord of length three is vertical, a similar analysis by expressing x as a function of y , $x(y)$, instead of y as a function of x , $y(x)$, can be performed. We compute the tangent vector at a stripe junction (x_i, y_i) using the midpoint rule (or central difference), i.e. the direction of the stripe tangent at point (x_i, y_i) is approximated by the direction of the chord connecting points (x_{i-1}, y_{i-1}) and (x_{i+1}, y_{i+1}) . Appropriate adjustment, using forward and backward difference for the tangent computation, is performed at the two end points.

The error in computing the stripe tangential directions using the midpoint rule is mainly due to two factors: (1) the midpoint approximation is not entirely correct. It is well established^(24, 25) that the error in central difference approximation is proportional to the length of the interval $|x_{i+1} - x_{i-1}|$ and to the maximal third derivative $y^{(3)}(\xi)$ of the curve at a point ξ inside the interval $[x_{i-1}, x_{i+1}]$, and (2) the y positions at x_{i+1} and x_{i-1} are not entirely reliable due to error in image

processing. In the following proposition, we establish the upper bound of the angular deviation, $\Delta\alpha$, between the true tangent vector and that computed using the midpoint rule.

Proposition 2 (error in computing stripe tangential direction). The direction of the tangent vector computed using the midpoint rule on a set of discrete points at (x_i, y_i) is within $\Delta\alpha$ to the true tangent of the curve $y(x)$ at (x_i, y_i) , where:

$$\Delta\alpha \leq \tan^{-1} \frac{4E_{y'}}{4 - E_{y'}^2}, \quad (8)$$

and $E_{y'}$ is the upper bound of $\Delta y'(x)$ which is:

$$E_{y'} = \frac{E_r}{h} + \frac{h^2}{6} M_f, \quad (9)$$

where E_r is the maximum image processing error in measuring a y coordinate; h is the length of the sampling interval; M_f is defined as:

$$M_f = \text{Max} |y^{(3)}(x)|, \quad (10)$$

for $x \in [x_{i-1}, x_{i+1}]$.

An interesting observation can be made about the proposition. Observing in equation (9) the size of the first term is inversely proportional to h , while that of the second term is squarely proportional to h . The first term is due to the error in image processing. Since we assume that the error in reading the y coordinate is bound above by a constant E_r regardless of the x location, the closer the two end points the larger the effect E_r has on the orientation measurement. The second term is due to numerical truncation, i.e. the midpoint rule is more accurate over a short interval, hence, a small h is preferred. Based on the above observations, we can derive the optimal sampling interval as follows: In order to determine a value h which minimizes $E_{y'}$, we find the stationary point of $E_{y'}$, or $E_{y'}'(h) = -(E_r/h^2) + (h/3)M_f = 0$. We see that $E_{y'}'(h) = 0$ if $h = (3E_r/M_f)^{1/3}$. Therefore, by properly selecting the sampling interval h , we may lower the error in computing $E_{y'}$. Furthermore, since $E_{y'} \ll 1$, we have:

$$\frac{4E_{y'}}{4 - E_{y'}^2} \rightarrow E_{y'},$$

when $E_{y'} \rightarrow 0$. Therefore, due to the fact that $E_{y'} \ll 1$, $\Delta\alpha \leq E_{y'}$ or $\Delta\alpha$ is also minimized.

3.3. Error propagation in vector product

The computed stripe orientation are used to recover the imaged surface orientation through equations (6) and (7), which involve three vector cross-products. Hence, the next logical step is to show how error in the stripe orientation propagates in a vector product. In the following discussion, all vectors are unit vectors except where noted. To simplify the notation, we also use (\mathbf{a}, \mathbf{b}) as a short hand for the angle between vectors \mathbf{a} and \mathbf{b} .

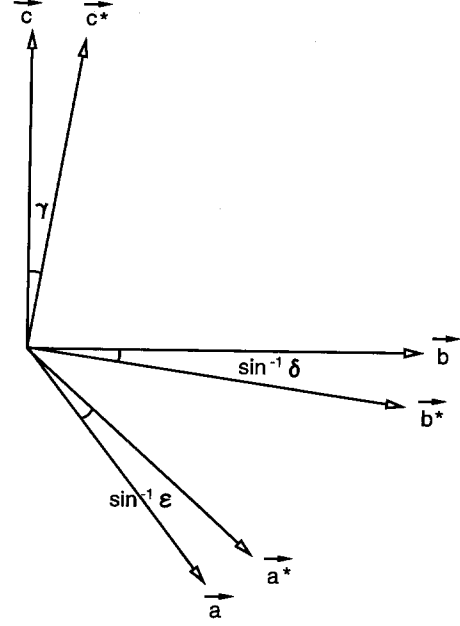


Fig. 4. Error in vector cross-product.

Proposition 3 (error propagation in vector product). Referring to Fig. 4, let $\mathbf{a}, \mathbf{a}^*, \mathbf{b}, \mathbf{b}^*$ be unit vectors, and $\epsilon = \sin(\mathbf{a}, \mathbf{a}^*)$, $\delta = \sin(\mathbf{b}, \mathbf{b}^*)$. If $\mathbf{c} = \mathbf{a} \times \mathbf{b}$, $\mathbf{c}^* = \mathbf{a}^* \times \mathbf{b}^*$, and $\gamma = (\mathbf{c}, \mathbf{c}^*)$, we have:

$$\begin{aligned} \sin \gamma &\leq \frac{\max(\epsilon, \delta, \sqrt{\epsilon^2 + \delta^2 - 2\epsilon\delta \cos(\mathbf{a}^*, \mathbf{b}^*)})}{\sin(\mathbf{a}^*, \mathbf{b}^*)}, \\ &\leq \max(\sin(\mathbf{a}, \mathbf{a}^*), \sin(\mathbf{b}, \mathbf{b}^*)) \max\left(2\sqrt{\frac{2}{3}}, \frac{1}{\sin(\mathbf{a}^*, \mathbf{b}^*)}\right). \end{aligned} \quad (11)$$

Especially, when $\pi/6 \leq (\mathbf{a}^*, \mathbf{b}^*) \leq \pi/2$, we can obtain the following simplified inequality:

$$\sin \gamma \leq 2 \max(\sin(\mathbf{a}, \mathbf{a}^*), \sin(\mathbf{b}, \mathbf{b}^*)). \quad (13)$$

In the above proposition, \mathbf{a}^* and \mathbf{b}^* denote measured stripe orientations which may not confirm with the true orientations \mathbf{a} and \mathbf{b} . Hence, the vector product $\mathbf{c}^* = \mathbf{a}^* \times \mathbf{b}^*$ also differs from the true vector product $\mathbf{c} = \mathbf{a} \times \mathbf{b}$. The maximum angular derivation $\gamma = (\mathbf{c}, \mathbf{c}^*)$ is established in the proposition.

3.4. Error in orientation measurement using structured lighting

We are now ready to analyse the error in measuring surface orientation with structured lighting. As mentioned in Section 3.1, surface normal \mathbf{P}_o is computed by taking three successive cross-products (Fig. 3):

$$\begin{aligned} \mathbf{P}_o &= \mathbf{L}_{o1} \times \mathbf{L}_{o2} \\ &= (\mathbf{N}_{g1} \times \mathbf{N}_{i1}) \times (\mathbf{N}_{g2} \times \mathbf{N}_{i2}) \\ &= ((\mathbf{P}_g \times \mathbf{I}_{g1}) \times (\mathbf{P}_i \times \mathbf{I}_{i1})) \times ((\mathbf{P}_g \times \mathbf{I}_{g2}) \times (\mathbf{P}_i \times \mathbf{I}_{i2})). \end{aligned} \quad (14)$$

The goal is thus to arrive at an expression for $(\mathbf{P}_o, \mathbf{P}_o^*)$, i.e. the angular deviation between the true surface normal (\mathbf{P}_o) and the computed one (\mathbf{P}_o^*) , using equation (14) and results established in the two previous sections. The analysis involves the following steps:

(1) In the first line of equation (14), the angle between \mathbf{L}_{o1} and \mathbf{L}_{o2} can be shown to be related to $(\mathbf{P}_g, \mathbf{P}_o)$, the angle between the normals to the grid plane and to the imaged surface, by the following equations:

$$\begin{aligned}\cos(\mathbf{L}_{o1}, \mathbf{L}_{o2}) &= \cos(\mathbf{L}_{o1}, \mathbf{P}_g) \cos(\mathbf{L}_{o2}, \mathbf{P}_g) \\ \cos(\mathbf{P}_g, \mathbf{P}_o) &= \sin(\mathbf{L}_{o1}, \mathbf{P}_g) \sin(\mathbf{L}_{o2}, \mathbf{P}_g)\end{aligned}\quad (15)$$

When the angle between the grid plane and the object surface approaches $\pi/2$, no pattern is shined on the surface and an analysis based on the perceived pattern orientation is impossible. By restricting the computation only to the case where $(\mathbf{P}_g, \mathbf{P}_o) \leq (11/24)\pi(82.5^\circ)$, we have $\pi/6 \leq (\mathbf{L}_{o1}^*, \mathbf{L}_{o2}^*) \leq \pi/2$ [from equation (15), assuming that the difference between \mathbf{L}_{oi} and \mathbf{L}_{oi}^* , $i = 1, 2$ is small] and hence by Proposition 3:

$$\sin(\mathbf{P}_o, \mathbf{P}_o^*) \leq 2\max(\sin(\mathbf{L}_{o1}, \mathbf{L}_{o1}^*), \sin(\mathbf{L}_{o2}, \mathbf{L}_{o2}^*)).$$

(2) In the second line of equation (14), the angle between \mathbf{N}_{gk} and \mathbf{N}_{ik} can be shown to be:

$$\begin{aligned}(\mathbf{N}_{gk}, \mathbf{N}_{ik}) &= \cos^{-1}(\cos \theta \cos \psi_{yk}^g \cos \psi_{yk}^i \\ &\quad + \sin \psi_{yk}^g \sin \psi_{yk}^i),\end{aligned}\quad (16)$$

where θ is the angle between the X axis of the grid coordinate system and that of the image coordinate system (Fig. 1), ψ_{yk}^g is the angle between the Y axis of the grid coordinate system and vectors \mathbf{l}_{gk} , ($k = 1, 2$) and ψ_{yk}^i is the angle between Y axis of the image coordinate system and vectors \mathbf{l}_{ik} , ($k = 1, 2$).

In general, ψ_y^i is proportional to ψ_y^g . When $\theta \gg 0$, $(\mathbf{N}_{gk}, \mathbf{N}_{ik})$ increases as ψ_y^g decreases. Intuitively, when ψ_y^g and ψ_y^i approach $\pi/2$, \mathbf{N}_{ik} and \mathbf{N}_{gk} become parallel, and when ψ_y^g and ψ_y^i approach 0, the angle $(\mathbf{N}_{gk}, \mathbf{N}_{ik})$ is no less than θ . If we have two sets of orthogonal grid marks, then ψ_{y1}^g decreases as ψ_{y2}^g increases. Hence, the best choice should be $\psi_{y1}^g = \psi_{y2}^g = \pi/4$. Therefore, if we choose $\psi_{y1}^g = \psi_{y2}^g = \pi/4$, then $(\mathbf{N}_{gk}, \mathbf{N}_{ik}) \geq \pi/6$. This yields (again by Proposition 3):

$$\begin{aligned}\sin(\mathbf{L}_{ok}, \mathbf{L}_{ok}^*) &\leq 2\max(\sin(\mathbf{N}_{gk}, \mathbf{N}_{gk}^*), \sin(\mathbf{N}_{ik}, \mathbf{N}_{ik}^*)), \\ &\quad k = 1 \text{ or } 2.\end{aligned}\quad (17)$$

Combining equations (16) and (17), we have:

$$\begin{aligned}\sin(\mathbf{P}_o, \mathbf{P}_o^*) &\leq 2\max(\sin(\mathbf{L}_{o1}, \mathbf{L}_{o1}^*), \sin(\mathbf{L}_{o2}, \mathbf{L}_{o2}^*)) \\ &\leq 4\max(\sin(\mathbf{N}_{g1}, \mathbf{N}_{g1}^*), \sin(\mathbf{N}_{g2}, \mathbf{N}_{g2}^*), \\ &\quad \sin(\mathbf{N}_{i1}, \mathbf{N}_{i1}^*), \sin(\mathbf{N}_{i2}, \mathbf{N}_{i2}^*)).\end{aligned}\quad (18)$$

(3) Finally, the angle between \mathbf{P}_g^* and \mathbf{l}_{gk}^* , as well as that between \mathbf{P}_i^* and \mathbf{l}_{ik}^* , $k = 1$ or 2 , are almost $\pi/2$ (they will be exactly $\pi/2$ if the parallel projection model is assumed true). Hence, the condition in Proposition 3 is satisfied (that the angle between the two cross-product vectors are larger than $\pi/3$). We have [using the

$(\mathbf{a}^*, \mathbf{b}^*) = \pi/2$ approximation in equation (40)]:

$$\begin{aligned}\sin(\mathbf{N}_{gk}, \mathbf{N}_{gk}^*) &\leq \sqrt{\sin^2(\mathbf{P}_g, \mathbf{P}_g^*) + \sin^2(\mathbf{l}_{gk}, \mathbf{l}_{gk}^*)} \\ &\leq \sqrt{\sin^2 \Phi_g + \sin^2(\mathbf{l}_{gk}, \mathbf{l}_{gk}^*)} \\ \sin(\mathbf{N}_{ik}, \mathbf{N}_{ik}^*) &\leq \sqrt{\sin^2(\mathbf{P}_i, \mathbf{P}_i^*) + \sin^2(\mathbf{l}_{ik}, \mathbf{l}_{ik}^*)} \\ &\leq \sqrt{\sin^2 \Phi_i + \sin^2(\mathbf{l}_{ik}, \mathbf{l}_{ik}^*)},\end{aligned}\quad (19)$$

where we use Φ_g and Φ_i to denote the maximum modeling error, i.e. the error in using a parallel model to approximate the slide projection process and the image formation process, respectively. Substituting equation (19) into (18), we obtain:

$$\begin{aligned}\sin(\mathbf{P}_o, \mathbf{P}_o^*) &\leq 4\max(\sqrt{\sin^2 \Phi_g + \sin^2(\mathbf{l}_{gk}, \mathbf{l}_{gk}^*)}, \\ &\quad \sqrt{\sin^2 \Phi_i + \sin^2(\mathbf{l}_{ik}, \mathbf{l}_{ik}^*)}), \quad k = 1, 2.\end{aligned}\quad (20)$$

Furthermore, the item $\sin^2(\mathbf{l}_{gk}, \mathbf{l}_{gk}^*)$ is an experimental error, which denotes the deviation of the stripe orientation in the projector from its calibrated value. It can be made smaller than all the other error terms by carefully placing the patterned slide into the projector. Especially, when $\Phi_g \approx \Phi_i$, the equation is simplified as:

$$\sin(\mathbf{P}_o, \mathbf{P}_o^*) \leq 4\sqrt{\sin^2 \Phi_i + \max(\sin^2(\mathbf{l}_{ik}, \mathbf{l}_{ik}^*)), \quad k = 1, 2.}\quad (21)$$

In equation (21), one notices that there are two dominating error sources in surface orientation computation using structured lighting: Φ_i , the parallel projection modeling error (Section 2) and $(\mathbf{l}_{ik}, \mathbf{l}_{ik}^*)$, the error in computing stripe tangential directions (Section 3.2). The effect of the former can be reduced when a camera lens with a large focal length is used. The latter is composed of errors due to image processing and numerical differentiation. The effect of this term can be made small when the sampling interval is carefully chosen.

4. ERROR ANALYSIS OF CURVATURE MEASUREMENT

4.1. Principal surface curvature inference technique

To compute principal surface curvatures using structured lighting, curvatures of the projected stripes on the imaged object's surface are computed first. These curvatures are then related to those of the normal sections that share the same tangential directions as the projected stripes. Principal curvatures and their directions at the stripe junction under consideration are then recovered using Euler's theorem.^(13,14)

Referring to Fig. 1, we first extract the projected pattern \mathbf{l}_{ik} from a structured-light coded image and project it back onto the object surface. A curve fitting process is then performed on the back-projected stripe, \mathbf{L}_{ok} , in space. The curvature, κ_k , of the back-projected space curve \mathbf{L}_{ok} is then calculated.

Next, the curvature of the normal section which shares the same tangential direction as the projected curve \mathbf{L}_{ok} at a stripe junction \mathbf{J} is computed. Referring

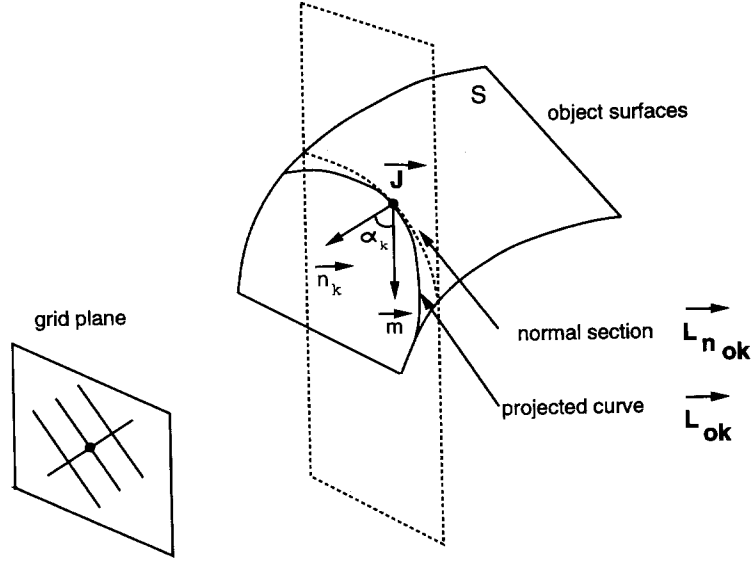


Fig. 5. The relation of a curve on the object surface and the normal section which shares the same tangential direction.

to Fig. 5, a normal section at a point J on a surface S is the intersection of S with a plane which contains the normal to S at J . A well-known theorem (Meusnier's theorem) in differential geometry relates the curvature κ_k of L_{ok} to the curvature, κ_{nk} , of the normal section L_{nok} , which shares the same tangential direction as L_{ok} at J . Meusnier's theorem⁽²⁷⁾ states that the following relation holds (Fig. 5):

$$\kappa_{nk} = \kappa_k \cos \alpha_k, \quad (22)$$

where α_k is the angle between the surface normal \mathbf{m} and the principal normal direction \mathbf{n}_k of the curve L_{ok} at the grid junction J (Fig. 5).

The principal surface curvatures are then computed from the curvatures of the normal sections at the stripe junction under consideration. Euler's theorem⁽²⁷⁾ relates the normal curvature along a particular tangential direction to the extremal curvatures, κ_1 and κ_2 , at the same point by:

$$\kappa_{nk} = \kappa_1 \cos^2 \theta_k + \kappa_2 \sin^2 \theta_k, \quad (23)$$

where θ_k is the angle between the tangential direction of L_{ok} and that of κ_1 .

From the above discussion, we see that κ_{nk} can be inferred from κ_k [equation (22)] and κ_k in turn can be computed based on the observed image pattern. Hence at a stripe junction, each stripe in the projection pattern provides one equation to solve for the unknowns κ_1 and κ_2 . In general, three sets of stripes are needed to solve for κ_1 , κ_2 and the direction of κ_1 (the direction of κ_2 is always perpendicular to that of κ_1 at a regular surface point). Although for certain simple surfaces, such as spherical, cylindrical and conic surfaces, it was proven that two sets of stripes are sufficient for shape recovery. Details of the process can be found in references (13, 14).

In this section, we analyse the error in computing surface curvatures using structured lighting. In our analysis, the error which arises in computing the curvature of a projected curve is studied first. Recall that a projected curve on the imaged object's surface results from the intersection of the surface with a *planar* light sheet from the structured light projection, hence, the projected curves on the object's surface can be curved, but are always planar. After a suitable coordinate transform, a projected curve can be represented in a convenient form of $y = f(x)$ in a small neighborhood. Furthermore, we make the assumption that the imaged surface is locally C^2 (i.e. with continuous derivatives up to the second order), and hence, the projected curves are also locally C^2 . Curvature computation proceeds as follows: firstly, a set of discrete sampling points are extracted along a projected curve; secondly, three adjacent sampling points are fitted with a conic arc; and finally, curvature is computed using the coefficients from conic fitting.

Our analysis proceeds as follows: in Lemma 1 (Section 4.2), we establish an important result concerning the above curvature computation process based on local curve fitting. We show that if two C^2 curves share three common points over a closed interval—which is the case with a projected stripe and its conic fitting over the three adjacent points used in the curvature computation—then there exists at least one point in the interval where the two curves share the same curvature. Lemma 1 is then used to establish Proposition 4 which gives the upper bound—over the whole curve fitting interval—in the deviation of the curvature measurements between a projected stripe and its conic fitting.

Up to now we have not considered possible errors in image processing. As stated repeatedly, when points

are sampled along a projected curve, the extracted point locations along the curve are subjected to errors in image processing. In Proposition 5 we modify Proposition 4 to include the effects of the image processing error. Finally, in Section 4.3, we show how the curvature error propagates in applying Meusnier's and Euler's theorems and establish the main result for analysing surface curvature computation using structured lighting.

4.2. Error in computing the curvature of the projected stripes

First, we introduce the following lemma:

Lemma 1 (curvatures of two intersecting curves). Suppose that we are given two C^2 curves $y = f(x)$ and $y = g(x)$, $g'(x) \neq 0$, both defined on a closed interval $[a, b]$ with coincident end points, i.e. $f(a) = g(a)$, $f(b) = g(b)$. If another point c inside the interval $[a, b]$ can be found where the two curves intersect, i.e. $f(c) = g(c)$, then there exists at least one point in the interval where the two curves share the same curvature, or $\exists \xi \in (a, b)$, $\kappa_f(\xi) = \kappa_g(\xi)$.

Lemma 1 is a generalization of the famous Mean Value Theorem⁽²⁸⁾ and its proof is given in the Appendix. This lemma establishes that if a conic fitting $g(x)$ is used to approximate a projected stripe $f(x)$ in such a way that they share three common sampling points (x_k, y_k) , $k = i - 1, i$ and $i + 1$, then there must exist a point in the interval (x_{i-1}, x_{i+1}) where the two curves share identical curvature measurement. However, this particular point need not be x_i , the place where we approximate κ_f by κ_g . Hence, the lemma alone is not sufficient for the analysis. The following proposition (Proposition 4) establishes the error bound for the whole curve fitting interval.

Proposition 4 (curvature approximation w/o image processing error). Suppose that we have two C^2 curves $y = f(x)$ and $y = g(x)$ defined on a closed interval $[a, b]$ and an increasing sequence of points (of length at least three), x_0, \dots, x_{n-1} , $a = x_0 < x_1 < \dots < x_{n-1} = b$ and $f(x_i) = g(x_i)$, for $i = 0, \dots, n - 1$. Furthermore, denote the upper bound of the change of curvature inside interval $[a, b]$ as M_f and M_g for curves $f(x)$ and $g(x)$, respectively, i.e. $|\kappa'_f(x)| \leq M_f$, $|\kappa'_g(x)| \leq M_g$, and $\max |x_{i+1} - x_i|$, $0 \leq i < n - 1$, as h , then we have:

$$|\kappa_f(x) - \kappa_g(x)| \leq 2h(M_f + M_g), \quad \text{for } a < x < b. \quad (24)$$

When $x = x_i$, we have:

$$|\kappa_f(x) - \kappa_g(x)| \leq h(M_f + M_g). \quad (25)$$

Basically, Proposition 4 states that if $f(x)$ represents a projected stripe, then a conic fitting $g(x)$ can be computed and κ_g is used to approximate κ_f . Since $g(x)$ is computed using three sampling points on $f(x)$, it follows from Proposition 4 that an upper bound can be established on the approximation of κ_f by κ_g .

One obvious choice of $g(x)$ is a circular arc which passes through points $(x_{i-1}, f(x_{i-1}))$, $(x_i, f(x_i))$ and $(x_{i+1}, f(x_{i+1}))$. This choice is based on the following

reasons: Firstly, it is evident that $M_g = 0$ if a circular arc is used, hence equation (24) is simplified further. Secondly, for simple parametric surfaces such as planar, spherical, conic and cylindrical patches, the projected stripes can be easily shown to possess the shape of a circle, an ellipse, a parabola or a hyperbola,⁽²⁹⁾ all of which can be approximated well by a set of circular arcs over a small neighborhood (for instance, four circular arc segments can be used to approximate an ellipse in descriptive geometry⁽³⁰⁾).

Now, we consider the effect of the image processing error. When the sampling point positions are subjected to noise corruption in image processing, $g(x)$ does not share three points with the ideal projection curve $f(x)$, but with the noise-corrupted version $f^*(x)$. The error in curve approximation is then due to two factors: (1) the use of a conic section $g(x)$ to approximate $f(x)$ which may or may not be a conic (Proposition 4) and (2) the use of $f^*(x_{i-1})$, $f^*(x_i)$, $f^*(x_{i+1})$ in the curve fitting process. We introduce the following proposition.

Proposition 5 (curvature approximation with image processing error). If a projected curve $f(x)$ is approximated by a set of circular arcs, with one fitting of $g(x)$ over three adjacent sampling points $(x_{i-1}, f^*(x_{i-1}))$, $(x_i, f^*(x_i))$ and $(x_{i+1}, f^*(x_{i+1}))$, $1 < i < n$, in the neighborhood of junction $(x_i, f^*(x_i))$, where $f^*(x)$ denotes the noise-corrupted version of $f(x)$, then the absolute error in approximating the curvature $\kappa_f(x)$ at $(x_i, f(x_i))$ by $\kappa_g(x)$ is bound by:

$$\Delta\kappa \leq hM_f + \frac{4E_r}{h^2}, \quad (26)$$

and the relative error is:

$$\delta\kappa \leq R \left(hM_f + \frac{4E_r}{h^2} \right), \quad (27)$$

where h is the length of the sampling interval, R is the radius of the circular arc used to approximate $f^*(x)$ at junction $(x_i, f^*(x_i))$. M_f is defined in Proposition 4 and E_r is the upper bound of the image processing error.

4.3. Error in curvature measurement using structured lighting

In the previous section, we analysed the error in computing the curvature of the projected stripes, now we show how the error propagates in applying Meusnier's and Euler's theorems in computing the principal surface curvatures from structured lighting.

In Meusnier's theorem, κ_{nk} is the projection of κ_k in the plane containing the normal section \mathbf{L}_{nok} . Hence, the error in computing κ_{nk} is due to the error in computing κ_k and in the projection process:

$$\begin{aligned} \kappa_{nk}^* &= \kappa^* \cos^* \alpha_k \\ &= (\kappa_k + \Delta\kappa_k)(\cos \alpha_k + \Delta\cos \alpha_k), \end{aligned}$$

thus

$$\begin{aligned} \Delta\kappa_{nk} &= \kappa_{nk}^* - \kappa_{nk} \\ &\approx \kappa_k \Delta \cos \alpha_k + \cos \alpha_k \Delta\kappa_k, \end{aligned}$$

and

$$\begin{aligned}\delta\kappa_{n_k} &= \frac{\Delta\kappa_{n_k}}{\kappa_{n_k}} \\ &\approx \delta\kappa_k + \delta\cos\alpha_k.\end{aligned}\quad (28)$$

The first term in equation (28) is given in the previous section and the second term denotes the relative error in computing $\cos\alpha_k$. Recall that α_k is the angle between the normal vector \mathbf{n}_k and the surface normal \mathbf{m} at the junction under consideration (Fig. 5). By denoting one other stripe at the junction as \mathbf{L}_{oj} , and the normal to the light sheet containing stripe \mathbf{L}_{ok} as \mathbf{N}_{gk} , we can write $\cos\alpha_k$ as:

$$\begin{aligned}\cos\alpha_k &= \mathbf{n}_k \cdot \mathbf{m} \\ &= (\mathbf{L}_{ok} \times \mathbf{N}_{gk}) \cdot (\mathbf{L}_{ok} \times \mathbf{L}_{oj}) \\ &= (\mathbf{L}_{ok} \cdot \mathbf{L}_{ok})(\mathbf{N}_{gk} \cdot \mathbf{L}_{oj}) - (\mathbf{L}_{ok} \cdot \mathbf{L}_{oj})(\mathbf{N}_{gk} \cdot \mathbf{L}_{ok}) \\ &= |\mathbf{L}_{ok}|^2(\mathbf{N}_{gk} \cdot \mathbf{L}_{oj}) - (\mathbf{L}_{ok} \cdot \mathbf{L}_{oj}) \cdot 0 \\ &= |\mathbf{L}_{ok}|^2(\mathbf{N}_{gk} \cdot \mathbf{L}_{oj}) \\ &= |\mathbf{L}_{ok}|^2 \cos(\mathbf{N}_{gk}, \mathbf{L}_{oj}) |\mathbf{L}_{oj}| \\ &= \frac{|\mathbf{L}_{ok}|^2 \cos(\mathbf{N}_{gk}, \mathbf{L}_{oj})}{|\mathbf{L}_{ok}|^2 \sin(\mathbf{L}_{ok}, \mathbf{L}_{oj})} \\ &= \frac{\cos(\mathbf{N}_{gk}, \mathbf{L}_{oj})}{\sin(\mathbf{L}_{ok}, \mathbf{L}_{oj})}.\end{aligned}\quad (29)$$

Hence, by denoting angles $(\mathbf{N}_{gk}, \mathbf{L}_{oj})$ and $(\mathbf{L}_{ok}, \mathbf{L}_{oj})$ as α_1 and α_2 , we have

$$\begin{aligned}\Delta\cos\alpha_k &= \left| \frac{\cos(\alpha_1 + \Delta\alpha_1)}{\sin(\alpha_2 + \Delta\alpha_2)} - \frac{\cos\alpha_1}{\sin\alpha_2} \right| \\ &\leq \frac{|\sin\alpha_2 \cos(\alpha_1 + \Delta\alpha_1) - \sin(\alpha_2 + \Delta\alpha_2) \cos\alpha_1|}{\sin^2\alpha_2} \\ &\approx \frac{|\sin(\alpha_2 - \alpha_1 - \Delta\alpha_1) - \sin(\alpha_2 - \alpha_1 + \Delta\alpha_2)|}{2\sin^2\alpha_2} \\ &\leq \frac{|\Delta\alpha_1 + \Delta\alpha_2|}{2\sin^2\alpha_2} \\ &\approx \frac{|(\mathbf{N}_{gk}, \mathbf{N}_{gk}^*)| + |(\mathbf{L}_{ok}, \mathbf{L}_{ok}^*)|}{2\sin^2(\mathbf{L}_{ok}, \mathbf{L}_{oj})}.\end{aligned}$$

Then the relative error in computing $\cos\alpha_k$ is:

$$\delta\cos\alpha_k \leq \frac{|(\mathbf{N}_{gk}, \mathbf{N}_{gk}^*)| + |(\mathbf{L}_{ok}, \mathbf{L}_{ok}^*)|}{2\sin(\mathbf{L}_{ok}, \mathbf{L}_{oj})\cos(\mathbf{N}_{gk}, \mathbf{L}_{oj})}, \quad (30)$$

where $(\mathbf{N}_{gk}, \mathbf{N}_{gk}^*)$, $(\mathbf{L}_{ok}, \mathbf{L}_{ok}^*)$ and $(\mathbf{L}_{oj}, \mathbf{L}_{oj}^*)$ are the angular errors in computing the directions of vectors \mathbf{N}_{gk} , \mathbf{L}_{ok} and \mathbf{L}_{oj} , respectively. As discussed in Section 3.4, $(\mathbf{N}_{gk}, \mathbf{N}_{gk}^*) \approx (\mathbf{P}_g, \mathbf{P}_g^*) \leq \Phi_g$ [equation (19)]. The error in computing \mathbf{L}_{ok} and \mathbf{L}_{oj} is given in equation (17).

The last step is to investigate the error in computing the principal surface curvature using Euler's theorem. As indicated in references (13, 14), three equations in three unknowns κ_1 , κ_2 and θ_1 are solved for the principal curvatures and their directions for general types of surfaces. While for the four parametric surfaces, planes, spheres, cones and cylinders, either $\kappa_1 = \kappa_2$ (for planes and spheres) or $\kappa_2 = 0$ (or cones and cylinders). Hence we end up with two equations in terms of

two unknowns κ_1 and θ_1 . Here we restrict our discussion to the four primitive surface types. The analysis of general free-form surfaces is similar, but the resulting expressions are much more complicated.

Cones and cylinders are simple surfaces of revolution which are obtained by rotating a line about an axis of revolution. The fact that the ruling is linear infers the special property $\kappa_2 = 0$ for cones and cylinders. This special property enables us to compute κ_1 and θ_1 using equation (23) with two sets of stripes:

$$\begin{cases} \kappa_{n_1} = \kappa_1 \cos^2\theta_1; \\ \kappa_{n_2} = \kappa_1 \cos^2(\theta_1 + \theta_{1,2}), \end{cases}$$

where we use $\theta_{1,2}$ as a shorthand for (L_{o1}, L_{o2}) . Solving the above equations, we obtain:

$$\kappa_1 = \frac{1}{\sin^2\theta_{1,2}} (\kappa_{n_1} - 2\cos\theta_{1,2}\sqrt{\kappa_{n_1}\kappa_{n_2}} + \kappa_{n_2}). \quad (31)$$

Now let:

$$u(\kappa_{n_1}, \kappa_{n_2}) = \kappa_{n_1} + \kappa_{n_2} - 2\cos\theta_{1,2}\sqrt{\kappa_{n_1}\kappa_{n_2}},$$

the relative error of κ_1 is:

$$\begin{aligned}\delta\kappa_1 &= \left| \frac{\kappa_1^* - \kappa_1}{\kappa_1} \right| \\ &\approx \left| \frac{u(\kappa_{n_1}^*, \kappa_{n_2}^*) \sin^2\theta_{1,2} - 1}{u(\kappa_{n_1}, \kappa_{n_2}) \sin^2\theta_{1,2}^*} - 1 \right| \\ &= \left| \frac{\sin^2\theta_{1,2} \Delta u(\kappa_{n_1}, \kappa_{n_2}) - u(\kappa_{n_1}, \kappa_{n_2}) \Delta \sin^2\theta_{1,2}}{u(\kappa_{n_1}, \kappa_{n_2}) \sin^2\theta_{1,2}^*} \right| \\ &\approx |\delta u(\kappa_{n_1}, \kappa_{n_2}) - \delta \sin^2\theta_{1,2}| \\ &\approx \left| \frac{\Delta\kappa_{n_1} + \Delta\kappa_{n_2} - 2\cos\theta_{1,2}\Delta\sqrt{\kappa_{n_1}\kappa_{n_2}}}{\kappa_{n_1} + \kappa_{n_2} - 2\cos\theta_{1,2}\sqrt{\kappa_{n_1}\kappa_{n_2}}} - \frac{\Delta \sin^2\theta_{1,2}}{\sin^2\theta_{1,2}} \right| \\ &\approx \left| \frac{\Delta\kappa_{n_1} + \Delta\kappa_{n_2} - \cos\theta_{1,2}\sqrt{\kappa_{n_1}\kappa_{n_2}}(\delta\kappa_{n_1} + \delta\kappa_{n_2})}{\kappa_{n_1} + \kappa_{n_2} - 2\cos\theta_{1,2}\sqrt{\kappa_{n_1}\kappa_{n_2}}} - \frac{\sin(2\theta_{1,2} + \Delta\theta_{1,2}) \sin\Delta\theta_{1,2}}{\sin^2\theta_{1,2}} \right| \\ &\approx \left| \frac{(\kappa_{n_1} - \cos\theta_{1,2}\sqrt{\kappa_{n_1}\kappa_{n_2}})\delta\kappa_{n_1} + (\kappa_{n_2} - \cos\theta_{1,2}\sqrt{\kappa_{n_1}\kappa_{n_2}})\delta\kappa_{n_2}}{\kappa_{n_1} + \kappa_{n_2} - 2\cos\theta_{1,2}\sqrt{\kappa_{n_1}\kappa_{n_2}}} - 2\cot\theta_{1,2} \sin\Delta\theta_{1,2} \right| \\ &\approx \left| \frac{(\kappa_{n_1} + \kappa_{n_2} - 2\cos\theta_{1,2}\sqrt{\kappa_{n_1}\kappa_{n_2}})(\delta\kappa_{n_1} + \delta\kappa_{n_2})}{4} \times \frac{(\kappa_{n_1} + \kappa_{n_2} - 2\cos\theta_{1,2}\sqrt{\kappa_{n_1}\kappa_{n_2}})}{-2\cot\theta_{1,2} \sin\Delta\theta_{1,2}} \right| \\ &= \left| \frac{(\delta\kappa_{n_1}, \delta\kappa_{n_2})}{4} - 2\cot\theta_{1,2} \sin\Delta\theta_{1,2} \right| \\ &\leq \frac{(\delta\kappa_{n_1}, \delta\kappa_{n_2})}{4} + 2\cot\theta_{1,2} \sin|\Delta\theta_{1,2}|. \quad (32)\end{aligned}$$

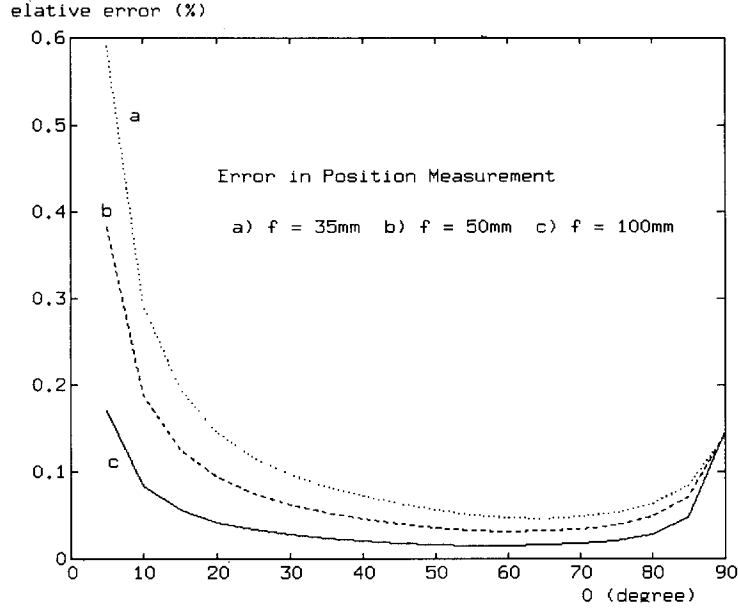


Fig. 6. 3D position error curve.

As can be seen in equation (32), $\delta\kappa_1$ comprises two terms. The first term is the relative error in computing the curvature of the normal sections that share the same tangential direction as the projected stripes [equation (28)]. This term itself is the sum of two error factors: $\delta\kappa_k$, error in computing the curvature of projected stripes which was discussed in Section 4.2, and $\delta\cos\alpha_k$, error due to the projection process in Meusnier's theorem which was given in equation (30). Both $\delta\cos\alpha_k$ and the second term $2\cot\theta_{1,2}\sin|\Delta\theta_{1,2}|$ are functions of the angle between the tangential directions of two stripes at the junction under consideration. The relationship between $\theta_{1,2}$ and $(\mathbf{P}_g, \mathbf{P}_o)$ —the angle between the normal to grid plane and the imaged surface normal—is given in equation (15).

5. SIMULATION RESULTS

For the simulation results presented here, we assume the following system parameters:

$$f_i = f_g = 100 \pm 0.5 \text{ mm},$$

$$D \approx 5 \text{ m},$$

$$W = 13.312 \text{ mm} \quad (1024 \text{ pixels}),$$

$$w_g \approx 39 \mu\text{m} \quad (3 \text{ pixels wide}),$$

where f_i and f_g are the focal lengths of the camera and the projector, D is the distance from the camera to the imaged object, W is the size of the image CCD array and w_g is the pattern width in the projection slide. These values are chosen to be their typical settings used in our experiment. W confirms with typical vendor specs in trade journals.

In the first simulation, we examine the effect angle θ —the angle between the x axes of the projector and the camera—has on the computation of 3D surface position. In Fig. 6, we plot the maximum relative error in computing surface depth as a function of θ which assumes a range 5° – 90° . Three cases—where the focal lengths of the camera and the projector assume the values of 35, 50 and 100 mm (curves a, b and c in Fig. 6, respectively)—are shown. As can be seen from Fig. 6, for all three curves, the largest error occurs when θ approaches zero, which implies that a stereo configuration where the optical axes of the two cameras are parallel is not an ideal choice as far as 3D position accuracy is concerned. Intuitively speaking, a large θ reduces the size of the uncertainty region where a feature point can lie and hence reduces the maximum relative error. Also, larger focal lengths do reduce the position error. The reason is that large focal lengths reduce the size of the field of view, and allow the same spatial image resolution to be used to describe a smaller 3D volume. Hence, higher feature resolution is achieved to reduce positioning error.

The second simulation gives the maximum angular error in computing surface orientation as a function of the angle θ . Again in Fig. 7, three curves are shown with different (P_g, P_o) angle which denotes the angle between the imaged surface normal and the projection direction (or the grid plane normal direction). As mentioned before, the desired configuration is to have the grid plane facing the imaged object surface, otherwise the width of the projected pattern increases [equation (3)], and so is the relative error in feature extraction [equation (5)]. Hence, a large (P_g, P_o) reduces the accuracy. It can also be seen in Fig. 7 that a large

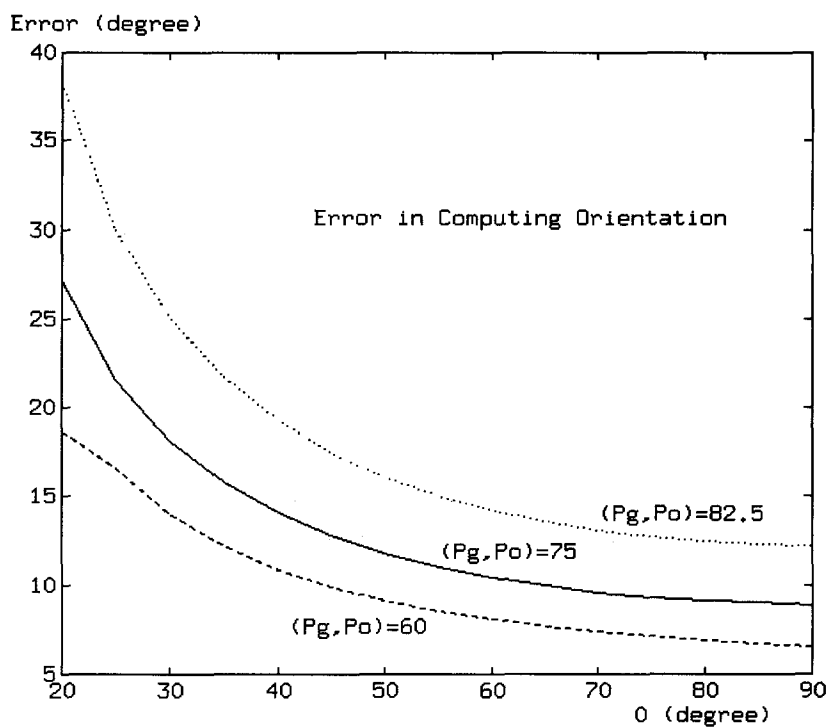


Fig. 7. 3D orientation error curve.

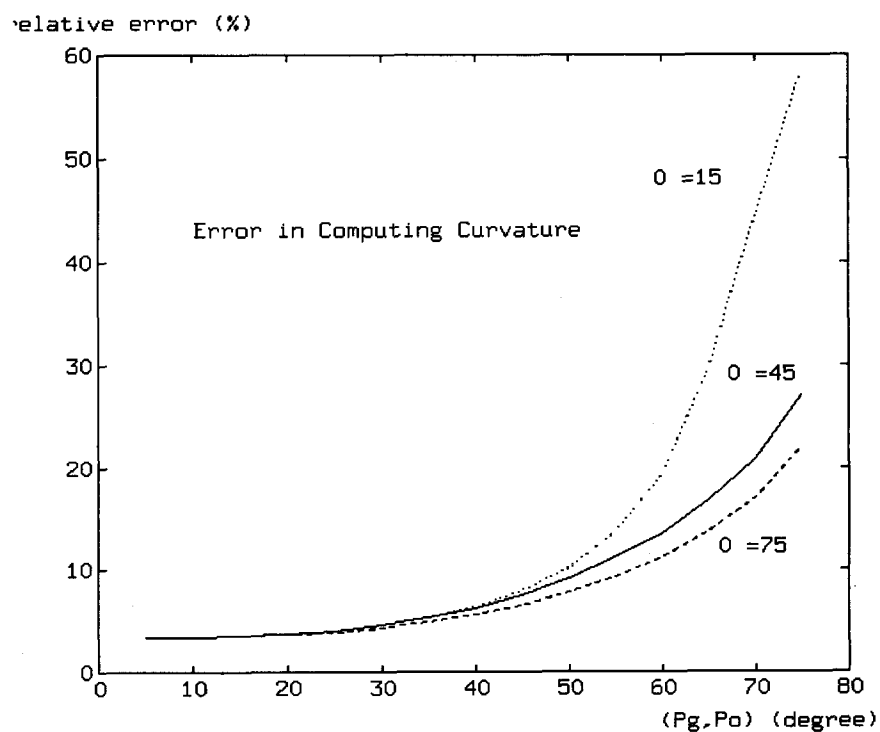


Fig. 8. 3D curvature error curve.

θ angle helps reducing the maximum angular error as we have predicted.

Finally, we perform a simulation to analyse the error in computing principal surface curvature. We use equation (32) to plot the curves in Fig. 8. As we have predicted, a large θ stabilizes the computation which a large (P_g, P_o) increases the error for the same reasons as stated above.

6. CONCLUSIONS

In summary, a detailed study of a number of error sources and their effects on measuring 3D surface properties using the structured lighting technique is presented in this paper. The derived error expressions give the absolute and relative error bounds in obtaining 3D surface position, orientation and curvature measurements. Error expressions are given in terms of the system parameters and likely error sources. We show that careful choice and calibration of the system parameters can greatly reduce the error in computing surface properties using structured lighting. Therefore, the designer of a structured light sensing system can use the derived expressions to fine tune the choice of the system parameters. Several suggestions on reducing the error in inferring surface properties using structured lighting can be drawn from this research:

- Use lenses of large focal length. Large focal length reduces the size of the field of view, allows the same spatial image resolution to be used to describe a smaller 3D volume. Hence, a higher feature resolution is obtained.
- Choosing a large angle between the optical axes of the projector and the camera helps reducing the size of the uncertainty region in locating a point in space.
- Differentiation is a process notorious for being noise-sensitive. The problem only becomes worse when high-order derivatives are used. A useful approach to stabilize high-order differentiations is to first apply conic fitting and smoothing, then derivatives can be computed based on the fitting coefficients. This method can greatly improve the accuracy in computing surface curvatures.
- The tangential direction computation is used in both the orientation and curvature measurements. It is also noise-sensitive. Carefully selecting the size of the sampling interval is one way to reduce error. Curve fitting to smooth out the undulation caused by noise before numerical differentiation is also important.
- Face the projector to the normal direction of the imaged surface area. This will help keeping the angle between the two projected stripes on the object surface large. Hence, computing surface normal as the vector product of the tangential directions of the projected stripes will be more stable.

REFERENCES

1. J. K. Aggarwal and C. H. Chien, 3-D structures from 2-D images, in *Advances in Machine Vision*, J. L. Sanz ed., pp. 64–121. Springer-Verlag, Berlin (1989).
2. H. G. Barrow and J. M. Tenenbaum, Computational vision, *Proc. IEEE* **69**, 572–595 (1981).
3. M. Brady, Computational approaches to image understanding, *ACM Comput. Surveys* **14**(1), 3–70 (March 1982).
4. J. K. Aggarwal and Y. F. Wang, Inference of object surface structure from structured lighting—an overview, in *Machine Vision, Algorithms, Architectures, and Systems*, H. Freeman, ed., pp. 193–220 Academic Press, San Diego, California (1988).
5. K. L. Boyer and A. C. Kak, Color-encoded structured light for rapid range sensing, *IEEE Trans. PAMI* **9**(1), 14–28 (January 1987).
6. W. Frobin and E. Hierholzer, Rasterstereography: A photogrammetric method for measurement of body surfaces, *Photogram. Eng. Remote Sensing* **47**, 1717–1724 (December 1981).
7. J. Le Moigne and A. M. Waxman, Projected light grids for short range navigation of autonomous robots, *Proc. 7th Int. Conf. Pattern Recognition* 203–206, (30 July–2 August 1984).
8. J. L. Posdamer and M. D. Altschuler, Surface measurement by space-encoded projected beam systems, *Comput. Graphics Image Process.* **18**, 1–17 (1982).
9. M. Potmesil and H. Freeman, Curved surface representation utilizing data extracted from multiple photographic images, *Proc. Workshop Represent. Three-Dimens. Objects*, pp. H-1–H-26. Philadelphia, 1–2 (May 1979).
10. Y. F. Wang, A. Mitiche and J. K. Aggarwal, Inferring local surface orientation with the aid of grid coding, *Proc. Third Workshop Comput. Vis. Represent. Control*, pp. 96–104. Bellaire, Michigan, 13–16 (October 1985).
11. Y. F. Wang, A. Mitiche and J. K. Aggarwal, Computation of surface orientation and structure of objects using grid coding, *IEEE Trans. PAMI*, **PAMI-9**(1), 129–137 (January 1987).
12. Y. F. Wang and J. K. Aggarwal, Geometric modeling using active sensing—an overview, *IEEE Control Syst. Mag.* **3**(2), 7–13 (1981).
13. Y. F. Wang and P. Liang, A new method for computing intrinsic surface properties, *IEEE Proc. CVPR*. San Diego, (4–8 June 1989).
14. Y. F. Wang, Characterizing 3-D surface structures from visual images, *IEEE Trans. PAMI*, **PAMI-13**(1), 52–60 (January 1991).
15. P. M. Will and K. S. Pennington, Grid coding: A preprocessing technique for robot and machine vision, *Art. Intell.* 319–329 (1971).
16. W. Frobin, Calibration and model reconstruction in analytical close-range stereophotogrammetry, Part I: Mathematical fundamentals, *Photogram. Eng. Remote Sensing* **48**, 67–72 (January 1982).
17. W. Frobin, Calibration and model reconstruction in analytical close-range stereophotogrammetry, Part II: special evaluation procedures for rasterstereography and moire topography, *Photogram. Eng. Remote Sensing* **48**, 215–220 (February 1982).
18. S. D. Blostein and T. S. Huang, Error analysis in stereo determination of 3-D point positions, *IEEE Trans. PAMI* **9**(6), 752–765 (November 1987).
19. L. Matthies and S. A. Shafer, Error modeling in stereo navigation, *IEEE Robot. Automat.* **3**(3), 239–248 (1987).
20. E. S. McVey and J. W. Lee, Some accuracy and resolution aspects of computer vision distance measurements, *IEEE Trans. PAMI*, **4**(6), 646–649 (November 1982).
21. J. J. Rodriguez and J. K. Aggarwal, Quantization error in stereo imaging, *IEEE Proc. CVPR*, pp. 153–158. Ann Arbor, Michigan, (5–9 June 1988).

22. A. Verri and V. Torre, Absolute depth estimate in stereopsis, *J. Opt. Soc. Am.* 3(3), 297–299 (1986).
23. R. Duda and P. Hart, *Pattern Classification and Scene Analysis*. Wiley-Interscience, New York (1973).
24. R. L. Burden and J. D. Faure, *Numerical Analysis*. Prindle, Weber & Schmidt, Boston, Massachusetts (1985).
25. G. M. Phillips and P. J. Taylor, *Theory and Applications of Numerical Analysis*. Academic Press, New York (1973).
26. F. Solina, Errors in stereo due to quantization, Technical Report MS-CIS-85-34, Department of Computer and Information Science, University of Pennsylvania, Philadelphia (1985).
27. D. J. Struik, *Differential Geometry*, 2nd edn. Addison-Wesley Publishing Company, Inc. Reading, Massachusetts (1961).
28. H. Sagan, *Advanced Calculus*. Houghton Mifflin, Boston, Massachusetts (1974).
29. G. C. Preston and A. R. Lovaglia, *Modern Analytic Geometry*, Harper & Row, Publishers, New York (1971).
30. B. L. Wellman, *Technical Descriptive Geometry*. McGraw Hill, New York (1957).

APPENDIX A: PROOFS OF PROPOSITIONS AND LEMMAS

Proposition 1 (feature location error). If points on the central line along a perceived stripe are selected and reported as the position of the stripe in the image plane, then the feature location error introduced is:

$$e_f = |x^* - x| \leq \frac{d_g f_i \max(\tan \alpha, \tan \beta)}{2d_i f_g \sin^2(\theta_g)} w_g,$$

where x denotes the (true) coordinate of a point on the projected curve and x^* is the coordinate of the approximation point, i.e. the center pixel; w_g is the stripe width in the projection slide, 2α equals the projection angle and 2β equals the viewing angle (Fig. 2).

In order to prove the above proposition, the following lemma is needed.

Lemma 2 (bisecting angle of a triangle). Given a triangle ABC and a line segment \overline{AD} which divides angle $\angle CAB$ into α_1 and α_2 and side \overline{BC} into two segments \overline{BD} and \overline{DC} , then the lengths of \overline{BD} and \overline{DC} satisfy the following relation:

$$\frac{|\overline{BD}|}{|\overline{DC}|} = \frac{\sin \alpha_1 \cdot \sin(\gamma - \alpha_2)}{\sin \alpha_2 \cdot \sin(\gamma + \alpha_1)}, \quad (33)$$

where $\gamma \equiv \angle ADB$. Furthermore, if \overline{AD} bisects side \overline{BC} into two segments of equal length or $|\overline{BD}| = |\overline{DC}|$, then the bisecting angles α_1 and α_2 approach each other as γ approaches $\pi/2$. Proof of Lemma 2 is straightforward and will not be presented here. Now we are ready to prove Theorem 1.

Proof. Consider a light sheet resulting from the projection of a stripe in the slide onto the imaged object surface. Referring to Fig. 2, in a typical structured light experiment ϕ_g , the angle between the projection direction and the optical axis of the projector, and ϕ_i , the angle between the viewing direction and the optical axis of the camera, are small [because $f \gg |l|$ in equation (1)]. This implies that γ_g , the angle between the projection direction and the grid plane, and γ_i , the angle between the viewing direction and the image plane, are close to $\pi/2$. Hence, if the center line of a stripe is selected as the position of the stripe, then by Lemma 2, we have $\alpha_1 \approx \alpha_2 \approx \alpha$, $\beta_1 \approx \beta_2 \approx \beta$ (Fig. 2).

Apply Lemma 2 to the triangles formed by the grid projection and the image formation (Fig. 2), we have:

$$\frac{L_1}{L_2} = \frac{\sin(\theta_g - \alpha)}{\sin(\theta_g + \alpha)}, \quad \frac{l_1}{l_2} = \frac{\sin(\theta_i - \beta)}{\sin(\theta_i + \beta)}.$$

Since:

$$w_o = L_1 + L_2 = l_1 + l_2,$$

and

$$\Delta w_o = |L_1 - l_1| = \left| \frac{L_1}{L_1 + L_2} - \frac{l_1}{l_1 + l_2} \right| w_o,$$

where

$$\begin{aligned} \frac{L_1}{L_1 + L_2} &= \frac{\sin(\theta_g - \alpha)}{\sin(\theta_g + \alpha) + \sin(\theta_g - \alpha)} \\ &= \frac{\sin \theta_g \cos \alpha - \cos \theta_g \sin \alpha}{2 \sin \theta_g \cos \alpha} \\ &= \frac{1}{2}(1 - \cot \theta_g \tan \alpha), \end{aligned}$$

and

$$\frac{l_1}{l_1 + l_2} = \frac{1}{2}(1 - \cot \theta_i \tan \beta),$$

therefore, the relative error is:

$$\begin{aligned} \delta w &= \left| \frac{L_1}{L_1 + L_2} - \frac{l_1}{l_1 + l_2} \right| \\ &= \frac{1}{2} |\cot \theta_g \tan \alpha - \cot \theta_i \tan \beta| \\ &\leq \frac{1}{2} (|\cot \theta_g \tan \alpha| + |\cot \theta_i \tan \beta|) \\ &= \frac{1}{2} (|\cot \theta_g| |\tan \alpha| + |\cot \theta_i| |\tan \beta|) \\ &\leq \frac{1}{2} \max(|\tan \alpha|, |\tan \beta|) (|\cot \theta_g| + |\cot \theta_i|). \end{aligned}$$

Since

$$\begin{aligned} 0 \leq \theta_g \leq \frac{\pi}{2}, \quad 0 \leq \theta_i \leq \frac{\pi}{2}, \\ 0 \leq \alpha \leq \frac{\pi}{2}, \quad 0 \leq \beta \leq \frac{\pi}{2}, \end{aligned}$$

we have

$$|\tan \alpha| = \tan \alpha, \quad |\tan \beta| = \tan \beta,$$

$$|\cot \theta_g| + |\cot \theta_i| = |\cot \theta_g + \cot \theta_i|.$$

The above inequality becomes:

$$\begin{aligned} \delta w &\leq \frac{1}{2} \max(|\tan \alpha|, |\tan \beta|) (|\cot \theta_g| + |\cot \theta_i|) \\ &= \frac{1}{2} \max(\tan \alpha, \tan \beta) (|\cot \theta_g + \cot \theta_i|) \\ &= \frac{1}{2} \max(\tan \alpha, \tan \beta) \left| \frac{\sin(\theta_g - \theta_i)}{\sin \theta_g \sin \theta_i} \right| \\ &\leq \frac{1}{2} \max(\tan \alpha, \tan \beta) \left| \frac{1}{|\sin \theta_g \sin \theta_i|} \right|. \end{aligned}$$

δw deduced above holds both on the object surface in three dimensions and in the image plane. Hence, the feature location error in the image plane is:

$$e_f = w_i \cdot \delta w$$

$$\leq \frac{d_g f_i \max(\tan \alpha, \tan \beta)}{2d_i f_g \sin^2(\theta_g)} w_g. \quad \square$$

Proposition 2 (error in computing stripe tangential direction). The direction of the tangent vector computed using the midpoint rule on a set of discrete points at (x_i, y_i) is within $\Delta\alpha$ to the true tangent of the curve $y(x)$ at (x_i, y_i) , where:

$$\Delta\alpha \leq \tan^{-1} \frac{4E_y}{4 - E_y^2},$$

and E_y is the upper bound of $\Delta y'(x)$, which is:

$$E_y = \frac{E_r}{h} + \frac{h^2}{6} M_f,$$

where E_r is the maximum image processing error in measuring a y coordinate; h is the length of the sampling interval; M_f is defined as:

$$M_f = \max|y^{(3)}(x)|,$$

for $x \in [x_{i-1}, x_{i+1}]$.

Proof. According to the midpoint rule, the tangent to a curve C at point $p = (x_i, y_i)$ is approximated by a chord \overline{AB} , where $A = (x_{i-1}, y_{i-1})$ and $B = (x_{i+1}, y_{i+1})$ are two other points on C and $x_{i-1} + h = x_i = x_{i+1} - h$. Using Taylor's theorem,^(24,25) we have:

$$y'(x_i) = \frac{y_{i+1} - y_{i-1}}{2h} - \frac{h^2}{6} y^{(3)}(\xi), \quad (34)$$

where ξ lies in between x_{i-1} and x_{i+1} . The last term in equation (34) gives the truncation error:

$$e_t = \frac{h^2}{6} y^{(3)}(\xi), \quad (35)$$

which is caused by numerical differentiation. However, as just mentioned, e_t is not the only error source in a numerical tangential computation. There is another error source due to image processing, i.e. we may not locate y_{i-1} and y_{i+1} precisely. Denote the measured y values at x_{i-1} and x_{i+1} as y_{i-1}^* and y_{i+1}^* , then we have:

$$y'(x_i) = \frac{y_{i+1}^* - y_{i-1}^*}{2h} - e_t - e_m, \quad (36)$$

and

$$\begin{aligned} e_m &= \left| \frac{y_{i+1}^* - y_{i-1}^*}{2h} - \frac{y_{i+1} - y_{i-1}}{2h} \right| \\ &= \left| \frac{y_{i+1}^* - y_{i+1}}{2h} - \frac{y_{i-1}^* - y_{i-1}}{2h} \right| \\ &\leq \frac{1}{h} \max(|y_{i+1}^* - y_{i+1}|, |y_{i-1}^* - y_{i-1}|) \end{aligned} \quad (37)$$

denotes the error caused by image processing. If the upper error bound in image processing is $E_r = \max(e_r)$ [e_r is defined in equation (4)], i.e. $|y^* - y| \leq E_r$, for $x \in [x_{i-1}, x_{i+1}]$, then the total error in evaluating $y'(x)$ —considering both error in numerical approximation and in image processing—is bound by:

$$E_{y'} = \max(e_t) + \max(e_m) = \frac{h^2}{6} M_f + \frac{E_r}{h}.$$

Since $y'(x)$ denotes the slope of a curve, error $\Delta y'(x)$ induces an angular error $\Delta\alpha$, which is:

$$\begin{aligned} y'^* &= \tan \alpha^* \\ &= \tan(\alpha + \Delta\alpha) \\ &= \frac{\tan \alpha + \tan \Delta\alpha}{1 - \tan \alpha \tan \Delta\alpha} \\ &= \frac{y' + \tan \Delta\alpha}{1 - y' \tan \Delta\alpha}, \end{aligned}$$

hence, we obtain:

$$\begin{aligned} \Delta\alpha &= \tan^{-1} \frac{\Delta y'}{y' y'^* + 1} \\ &= \tan^{-1} \frac{\Delta y'}{(y'^*)^2 - \Delta y y'^* + 1} \\ &\leq \tan^{-1} \frac{E_{y'}}{(y'^*)^2 - E_{y'} y'^* + 1}. \end{aligned} \quad (38)$$

When $y'^* = -E_{y'}/2$, expression $(y'^*)^2 - E_{y'} y'^* + 1$ reaches its minimum value of $1 - E_{y'}^2/4$ and $\Delta\alpha$ reaches its maximum value of:

$$\tan^{-1} \frac{4E_{y'}}{4 - E_{y'}^2}. \quad \square$$

Proposition 3 (error propagation in vector product). Referring to Fig. 4, let \mathbf{a} , \mathbf{a}^* , \mathbf{b} , \mathbf{b}^* be unit vectors, and $\varepsilon = \sin(\mathbf{a}, \mathbf{a}^*)$, $\delta = \sin(\mathbf{b}, \mathbf{b}^*)$. If $\mathbf{c} = \mathbf{a} \times \mathbf{b}$, $\mathbf{c}^* = \mathbf{a}^* \times \mathbf{b}^*$ and $\gamma = (\mathbf{c}, \mathbf{c}^*)$, we have:

$$\begin{aligned} \sin \gamma &\leq \frac{\max(\varepsilon, \delta, \sqrt{\varepsilon^2 + \delta^2 - 2\varepsilon\delta \cos(\mathbf{a}, \mathbf{b}^*)})}{\sin(\mathbf{a}^*, \mathbf{b}^*)}, \\ &\leq \max(\sin(\mathbf{a}, \mathbf{a}^*), \sin(\mathbf{b}, \mathbf{b}^*)) \max\left(2\sqrt{\frac{2}{3}}, \frac{1}{\sin(\mathbf{a}^*, \mathbf{b}^*)}\right). \end{aligned}$$

Especially, when $\pi/6 \leq (\mathbf{a}^*, \mathbf{b}^*) \leq \pi/2$, we can attain the following simplified inequality:

$$\sin \gamma \leq 2 \max(\sin(\mathbf{a}, \mathbf{a}^*), \sin(\mathbf{b}, \mathbf{b}^*)).$$

To prove the proposition, we introduce the following lemma.

Lemma 3 maximum of function $F(x, y) = x^2 + y^2 - 2\lambda xy$. Given $0 \leq \lambda \leq 1$, the function $F(x, y) = x^2 + y^2 - 2\lambda xy$ defined on $[0, \varepsilon] \times [0, \delta]$, $\varepsilon, \delta > 0$ attains its maximum at the three corner points $(\varepsilon, 0)$, $(0, \delta)$ and (ε, δ) of the rectangular $[0, \varepsilon] \times [0, \delta]$ and:

$$F(x, y) \leq \max(\varepsilon^2, \delta^2, \varepsilon^2 + \delta^2 - 2\lambda\varepsilon\delta). \quad (39)$$

Proof of the above lemma is straightforward and will not be presented here. Now we are ready to prove Proposition 3.

Proof. We have:

$$\begin{aligned} |\mathbf{c}| |\mathbf{c}^*| \sin \gamma &= |\mathbf{c} \times \mathbf{c}^*| \\ &= |(\mathbf{a} \times \mathbf{b}) \times (\mathbf{a}^* \times \mathbf{b}^*)| \\ &= |[(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{b}^*] \mathbf{a}^* - [(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{a}^*] \mathbf{b}^*| \\ &= \sqrt{|[(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{b}^*] \mathbf{a}^* - [(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{a}^*] \mathbf{b}^*|^2} \\ &= \sqrt{|(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{b}^*|^2 + |(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{a}^*|^2 - 2|(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{b}^*| |(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{a}^*| \cos(\mathbf{a}^*, \mathbf{b}^*)}. \end{aligned}$$

Now perform a coordinate rotation so that \mathbf{a} and \mathbf{b} lie on the X - Y plane, then $\mathbf{c} = \mathbf{a} \times \mathbf{b}$ is aligned along the Z direction with a length of $\sin(\mathbf{a}, \mathbf{b})$. One can easily verify that the projected length of \mathbf{a}^* in the Z direction is between 0 [if $(\mathbf{a}, \mathbf{a}^*)$ lies in the X - Y plane] and $\sin(\mathbf{a}, \mathbf{a}^*)$ [if $(\mathbf{a}, \mathbf{a}^*)$ is along the z -axis], thus the dot product $(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{a}^*$ satisfies:

$$0 \leq |(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{a}^*| \leq |\sin(\mathbf{a}, \mathbf{b}) \sin(\mathbf{a}, \mathbf{a}^*)| = |\sin(\mathbf{a}, \mathbf{b}) \varepsilon|.$$

Similarly, we have for \mathbf{b}^* :

$$0 \leq |(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{b}^*| \leq |\sin(\mathbf{a}, \mathbf{b}) \sin(\mathbf{b}, \mathbf{b}^*)| = |\sin(\mathbf{a}, \mathbf{b}) \delta|.$$

Substituting x with $|(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{b}^*|$, y with $|(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{a}^*|$ and λ with $\cos(\mathbf{a}^*, \mathbf{b}^*)$ into Lemma 3, we have:

$$\begin{aligned} |\mathbf{c}| |\mathbf{c}^*| \sin \gamma &= |\mathbf{c} \times \mathbf{c}^*| \\ &= \sqrt{|(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{b}^*|^2 + |(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{a}^*|^2 - 2|(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{b}^*| |(\mathbf{a} \times \mathbf{b}) \cdot \mathbf{a}^*| \cos(\mathbf{a}^*, \mathbf{b}^*)} \\ &\leq \max(\sqrt{\varepsilon^2 \sin^2(\mathbf{a}, \mathbf{b})}, \sqrt{\delta^2 \sin^2(\mathbf{a}, \mathbf{b})}, \\ &\quad \sqrt{\varepsilon^2 \sin^2(\mathbf{a}, \mathbf{b}) + \delta^2 \sin^2(\mathbf{a}, \mathbf{b}) - 2\varepsilon\delta \sin^2(\mathbf{a}, \mathbf{b}) \cos(\mathbf{a}^*, \mathbf{b}^*)}) \\ &= \sin(\mathbf{a}, \mathbf{b}) \max(\varepsilon, \delta, \sqrt{\varepsilon^2 + \delta^2 - 2\varepsilon\delta \cos(\mathbf{a}^*, \mathbf{b}^*)}). \end{aligned}$$

$$f(a) = g(a), f(g) = g(b), \\ \exists c \in (a, b), f(c) = g(c).$$

Fact I

$$\exists \eta_i \in (a, b), i = 1, \dots, n, n \geq 2, \\ f'(\eta_i) = g'(\eta_i), \\ \eta_1 \leq c \leq \eta_n.$$

$$\begin{array}{ccccc} \neg P_1 & & & & \neg P_2 \\ \neg P_1 \rightarrow Q_1 & & \text{Fact II} & & \neg P_2 \rightarrow Q_2 \\ \neg P_1 \rightarrow R_1 & \longrightarrow & P_1: \exists x_1, x_1 \in (a, b), & \text{Fact III} & \neg P_2 \rightarrow R_2 \\ Q_1 \wedge R_1 \rightarrow P_1 & & \kappa_f(x_1) \geq \kappa_g(x_1). & & Q_2 \wedge R_2 \rightarrow P_2 \\ P_1 & & & & P_2 \end{array}$$

Fact IV

$$P: \exists \xi, \xi \in (a, b), \\ \kappa_f(\xi) = \kappa_g(\xi)$$

$$\text{Where } Q_1: \forall x \in (\eta_1, \eta_n), f'(x) > g'(x). \quad Q_2: \forall x \in (\eta_1, \eta_n), f'(x) < g'(x). \\ R_1: \forall x \in (a, \eta_1), f'(x) < g'(x). \quad R_2: \forall x \in (\eta_1, b), f'(x) > g'(x).$$

Fig. A1. Steps in proving Lemma 1.

Hence:

$$\begin{aligned} \sin \gamma &= \frac{|\mathbf{c} \times \mathbf{c}^*|}{|\mathbf{c}| |\mathbf{c}^*|} \\ &= \frac{|\mathbf{c} \times \mathbf{c}^*|}{\sin(\mathbf{a}, \mathbf{b}) \sin(\mathbf{a}^*, \mathbf{b}^*)} \\ &\leq \frac{\max(\varepsilon, \delta, \sqrt{\varepsilon^2 + \delta^2 - 2\varepsilon\delta \cos(\mathbf{a}^*, \mathbf{b}^*)})}{\sin(\mathbf{a}^*, \mathbf{b}^*)} \end{aligned}$$

We have proved inequality (11). If certain ranges of (\mathbf{a}, \mathbf{b}) are given, the value of $\sin \gamma$ is bound as follows:

$$\sin \gamma \leq \begin{cases} \frac{\max(\varepsilon, \delta)}{\sin(\mathbf{a}^*, \mathbf{b}^*)}, & 0 \leq (\mathbf{a}^*, \mathbf{b}^*) < \pi/3; \\ 2\sqrt{\frac{2}{3}} \max(\varepsilon, \delta), & \pi/3 \leq (\mathbf{a}^*, \mathbf{b}^*) < \pi/2; \\ \sqrt{\varepsilon^2 + \delta^2}, & (\mathbf{a}^*, \mathbf{b}^*) = \pi/2; \end{cases} \quad (40)$$

that is:

$$\sin \gamma \leq \max(\sin(\mathbf{a}, \mathbf{a}^*), \sin(\mathbf{b}, \mathbf{b}^*)) \max\left(2\sqrt{\frac{2}{3}}, \frac{1}{\sin(\mathbf{a}^*, \mathbf{b}^*)}\right).$$

Especially, when $\pi/6 \leq (\mathbf{a}^*, \mathbf{b}^*) \leq \pi/2$, the following can be directly derived from inequality (11):

$$\sin \gamma \leq 2 \max(\sin(\mathbf{a}, \mathbf{a}^*), \sin(\mathbf{b}, \mathbf{b}^*)).$$

Lemma 1 (curvatures of two intersecting curves). Suppose that we are given two C^2 curves $y = f(x)$ and $y = g(x)$, $g'(x) \neq 0$, both defined on a closed interval $[a, b]$ with coincident end points, i.e. $f(a) = g(a)$, $f(b) = g(b)$. If another point c inside the interval $[a, b]$ can be found such that the two curves intersect, i.e. $f(c) = g(c)$, then there exists at least one

point in the interval where the two share the same curvature, or $\exists \xi \in (a, b), \kappa_f(\xi) = \kappa_g(\xi)$.

Proof. The proof process is sketched graphically in Fig. A1 and consists of four steps. First we prove Fact (I), which asserts that there exists an increasing sequence of points η_i of length at least two in the open interval (a, b) , that is $a < \eta_1 < \dots < \eta_n < b$, $n \geq 2$, which satisfies $f'(\eta_i) = g'(\eta_i)$, $i = 1, \dots, n$, and $\eta_1 < c < \eta_n$.

Fact (I) will then be used to establish Facts (II) and (III), which state that in interval (a, b) there must exist at least one point where $\kappa_f \geq \kappa_g$ [Fact (II)] and another point where $\kappa_f \leq \kappa_g$ [Fact (III)]. Since both f and g are C^2 , Facts (II) and (III) together establish that there exists at least one point in the interval where f and g assume the same curvature measurement $\kappa_f = \kappa_g$ (Fact IV and the Lemma).

Proof of Fact. (I). Apply the Mean Value Theorem (Cauchy)⁽²⁸⁾ to $f(x)$ and $g(x)$ in the interval $[a, c]$ and we know that there must exist at least one point $\eta_{j_1} \in (a, c)$ such that:

$$\frac{f'(\eta_{j_1})}{g'(\eta_{j_1})} = \frac{f(c) - f(a)}{g(c) - g(a)}.$$

Furthermore, because f and g coincide on three points a, b and c , or $f(a) = g(a)$, $f(c) = g(c)$, we have $f'(\eta_{j_1}) = g'(\eta_{j_1})$. Notice that $g(c) - g(a)$ will not vanish. If $g(c) - g(a) = 0$, then by Rolle's Theorem⁽²⁸⁾ there must exist at least one point $\mu \in (a, c)$ such that $g'(\mu) = 0$. This contradicts with the assumption that $g'(x) \neq 0$ in (a, b) . Following a similar argument, one can easily show that there must exist at least one point $\eta_{j_2} \in (a, b)$ such that $f'(\eta_{j_2}) = g'(\eta_{j_2})$.

Certainly, η_{j_1} and η_{j_2} should be included in the sequence η_i which proves that such a sequence exists with at least two points. Since we can always choose η_1 and η_n in such a way that $\eta_1 \leq \eta_{j_1} < c$ and $c < \eta_{j_2} \leq \eta_n$, hence we have $\eta_1 < c < \eta_n$. This completes the proof of Fact (I).

Proof of Facts (II) and (III). Next, we prove Facts (II) and (III); the proof procedures of these facts are similar and only one proof [that of Fact (II)] will be presented here. That is, we want to prove the following proposition $P_1: \exists x_1 \in (a, b), \kappa_f(x_1) \geq \kappa_g(x_1)$.

We prove P_1 by contradiction. As sketched in Fig. 9, we will assume that $\neg P_1$ is true (i.e. P_1 is false). Then based on P_1 we can prove two additional facts Q_1 and R_1 which, when put together, imply P_1 . Hence, a contradiction results which implies that P_1 is true. Below we give the proof in four steps:

(1) First, let $\neg P_1: \forall x \in (a, b), \kappa_f(x) < \kappa_g(x)$ be our hypothesis.

(2) $\neg P_1 \rightarrow Q_1$, where $Q_1: \forall x \in (\eta_1, \eta_n), f'(x) > g'(x)$.

We prove the logical equivalent proposition $\neg Q_1 \rightarrow P_1$. If Q_1 is not true, then there must exist at least one point $\mu \in (\eta_1, \eta_n)$ such that $f'(\mu) \leq g'(\mu)$. The C^2 property guarantees the continuity of $f'(x)$ and $g'(x)$, hence there must exist a nonzero region around μ such that the relation of the first derivatives of f and g is preserved. That is, a region (u, v) exists, with $u \in [\eta_1, \eta_n]$, $v \in [\eta_1, \eta_n]$ and $u < \mu < v$ such that $f'(x) \leq g'(x)$ for all $x \in (u, v)$. Furthermore, we have $f'(u) = g'(u)$ and $f'(v) = g'(v)$ [i.e. f' and g' change sign at the end points of the region (u, v)]. Hence, by Rolle's Theorem, there must exist a point $\zeta \in (u, v) = f''(\zeta) = g''(\zeta)$. Furthermore, ζ is in the interval (u, v) and hence $f'(\zeta) \leq g'(\zeta)$. Substitute this into:

$$\kappa_{y(x)} = \frac{|y''(x)|}{(1 + y'(x)^2)^{3/2}},$$

we conclude that $\kappa_f(\zeta) \geq \kappa_g(\zeta)$, which is P_1 .

(3) $\neg P_1 \rightarrow R_1$, where $R_1: \forall x \in (a, \eta_1), f'(x) < g'(x)$.

Again, we prove the logical equivalent proposition $\neg R_1 \rightarrow P_1$. $\neg R_1$ implies that there exists at least one $\mu \in (a, \eta_1)$ for which $f'(\mu) \geq g'(\mu)$. Then since $c < \eta_n$, the resolution must hold in the interval (a, c) . Now define $s(x) = f(x) - g(x)$, then $s'(x) \geq 0$ on (a, c) , that is, $s(x)$ is a non-decreasing function. However, we know that $s(a) = s(c) = 0$, which implies $s(x) = f(x) - g(x) \equiv 0, \forall x \in (a, c)$. This contradicts with the hypothesis $\neg P_1$.

(4) $Q_1 \wedge R_1 \rightarrow P_1$.

Let us select a region (u, v) such that $u < \eta_1, \eta_1 < v < \eta_n$ and denote $t(x) = f'(x) - g'(x)$. It is shown in the above discussion that $t(u) < 0, t(v) > 0$. Hence, we have:

$$0 < t(v) - t(u) = t'(\mu)(v - u),$$

or

$$0 < \frac{t(v) - t(u)}{v - u} = t'(\mu),$$

where $\mu \in (u, v)$. Now let both u and v approach η_1 , or $u \rightarrow \eta_1, v \rightarrow \eta_1$, then the above expression approaches the derivative of $t(x)$ at η_1 :

$$0 \leq \lim_{u, v \rightarrow \eta_1} \frac{t(v) - t(u)}{v - u} = t'(\eta_1),$$

that is, $f''(\eta_1) \geq g''(\eta_1)$. Together with $f'(\eta_1) = g'(\eta_1)$ implies that P_1 is true. Hence, we obtain a contradiction. This completes our proof for the part of P_1 .

Using a similar proof procedure, one can prove $P_2: \exists x_2 \in (a, b), \kappa_g(x_2) \leq \kappa_f(x_2)$.

Proof of Facts (IV). Combine the two propositions P_1 and P_2 with the fact that both $\kappa_f(x)$ and $\kappa_g(x)$ are continuous functions on (a, b) , we obtain proposition P which states that for at least one point in the interval (a, b) , f and g must share the same curvature value. That completes the proof of the Lemma. \square

Proposition 4 (curvature approximation w/o image processing error). Suppose that we have two C^2 curves $y = f(x)$ and $y = g(x)$ defined on a closed interval $[a, b]$ and an increasing

sequence of points (of length at least three), x_0, \dots, x_{n-1} , $a = x_0 < x_1 < \dots < x_{n-1} = b$ and $f(x_i) = g(x_i)$, for $i = 0, \dots, n-1$. Furthermore, denote the upper bound of the change curvature inside interval $[a, b]$ as M_f and M_g for curves $f(x)$ and $g(x)$, respectively, i.e. $|\kappa'_f(x)| \leq M_f$, $|\kappa'_g(x)| \leq M_g$, and $\max|x_{i+1} - x_i|, 0 \leq i < n-1$, as h , then we have:

$$|\kappa_f(x) - \kappa_g(x)| \leq 2h(M_f + M_g), \text{ for } a < x < b.$$

When $x = x_i$, we have:

$$|\kappa_f(x) - \kappa_g(x)| \leq h(M_f + M_g).$$

Proof. Since the x_i sequence spans the whole interval $[a, b]$ and is of length at least of three, every point x which lies in the interval must fall in between a pair of points in the x_i sequence, i.e. $\forall x \in (a, b), \exists i$ such that $x \in (x_{i-1}, x_{i+1})$. By Lemma 1, there must exist a point ξ in the interval (x_{i-1}, x_{i+1}) such that $\kappa_f(\xi) = \kappa_g(\xi)$. Thus, we have:

$$\begin{aligned} |\kappa_f(x) - \kappa_g(x)| &\leq |\kappa_f(x) - \kappa_f(\xi)| + |\kappa_f(\xi) - \kappa_g(x)| \\ &= |\kappa_f(x) - \kappa_f(\xi)| + |\kappa_g(\xi) - \kappa_g(x)| \\ &= |\kappa'_f(\eta_1)(x - \xi)| + |\kappa'_g(\eta_2)(\xi - x)| \\ &\leq M_f|x - \xi| + M_g|\xi - x| \\ &\leq 2h(M_f + M_g). \end{aligned}$$

If $x = x_i$ then $|x - \xi| \leq h$. Hence, we have:

$$|\kappa_f(x) - \kappa_g(x)| \leq h(M_f + M_g). \quad (41)$$

\square

Proposition 5 (curvature approximation with image processing error). If a projected curve $f(x)$ is approximated by a set of circular arcs, with one fitting of $g(x)$ over three adjacent sampling points $(x_{i-1}, f^*(x_{i-1}))$, $(x_i, f^*(x_i))$, and $(x_{i+1}, f^*(x_{i+1}))$, $1 < i < n$, in the neighborhood of junction $(x_i, f^*(x_i))$, where $f^*(x)$ denotes the noise-corrupted version of $f(x)$, then the absolute error in approximating the curvature $\kappa_f(x)$ at $(x_i, f(x_i))$ by $\kappa_g(x)$ is bound by:

$$\Delta\kappa \leq hM_f + \frac{2E_r}{h^3},$$

and the relative error is:

$$\delta\kappa \leq R \left(hM_f + \frac{2E_r}{h^3} \right),$$

where h is the length of the sampling interval, R is the radius of the circular arc used to approximate $f^*(x)$ at junction $(x_i, f^*(x_i))$. M_f is defined in Proposition 4 and E_r is the upper bound of the image processing error.

Proof. The first term of equation (26) denotes the maximum error caused by using a conic arc $g(x)$ to approximate a projected stripe $f(x)$ which may be of a higher order. This term comes directly from Propositional 4 with $g(x)$ being a circular arc and $M_g = \max|\kappa'_g| = 0$.

The second term represents the image processing error. Consider the triangle formed by using the three points under consideration, $(x_{i-1}, f(x_{i-1}))$, $(x_i, f(x_i))$, and $(x_{i+1}, f(x_{i+1}))$, as vertices and denote the three sides of the triangle as a, b and c . Then the radius of the circle which passes through these three points is:

$$R = \frac{abc}{2S}, \quad (42)$$

where S is the area of the triangle abc and can be computed as:

$$S = \frac{1}{2} \begin{vmatrix} x_{i-1} & f(x_{i-1}) & 1 \\ x_i & f(x_i) & 1 \\ x_{i+1} & f(x_{i+1}) & 1 \end{vmatrix}. \quad (43)$$

Since $(x_{i-1}, f(x_{i-1}))$, $(x_i, f(x_i))$ and $(x_{i+1}, f(x_{i+1}))$ are three successive sample points extracted from the projection stripe

$f(x)$, we have $x_{i+1} - x_i = x_i - x_{i-1} = h$. If we denote $f(x_i) - f(x_{i-1})$ and $f(x_{i+1}) - f(x_i)$ as d_1 and d_2 , respectively. Then the following relations hold:

$$\begin{aligned} x_{i+1} - x_{i-1} &= 2h \\ f(x_{i+1}) - f(x_{i-1}) &= d_1 + d_2, \end{aligned}$$

and hence:

$$S = \frac{h}{2}(d_2 - d_1),$$

$$abc = \sqrt{(h^2 + d_1^2)(h^2 + d_2^2)(4h^2 + (d_1 + d_2)^2)}.$$

Substituting the above into equation (42), we have:

$$R = \frac{\sqrt{(h^2 + d_1^2)(h^2 + d_2^2)(4h^2 + (d_1 + d_2)^2)}}{2h(d_2 - d_1)}. \quad (44)$$

In computing R^* , we use the noise-corrupted version $f^*(x)$. The largest error in computing R^* occurs when $\Delta f(x_{i-1}) = \Delta f(x_{i+1}) = E_r$ and $\Delta f(x_i) = -E_r$ (or $\Delta f(x_{i-1}) = \Delta f(x_{i+1}) = -E_r$ and $\Delta f(x_i) = E_r$, but this will not affect the following analysis). Then equation (44) becomes:

$$\begin{aligned} R^* &= \frac{\sqrt{(h^2 + (d_1 + \Delta d_1)^2)(h^2 + (d_2 + \Delta d_2)^2)(4h^2 + (d_1 + d_2 + \Delta d_1 + \Delta d_2)^2)}}{2h(d_2 - d_1 + \Delta d_2 - \Delta d_1)} \\ &= \frac{\sqrt{(h^2 + (d_1 + 2E_r)^2)(h^2 + (d_2 - 2E_r)^2)(4h^2 + (d_1 + d_2)^2)}}{2h(d_2 - d_1 - 4E_r)} \end{aligned}$$

hence, given $E_r \ll d_i$ and $E_r \ll h$, we have:

$$\begin{aligned} |\Delta \kappa_g| &= |\kappa_g^* - \kappa_g| \\ &= \left| \frac{1}{R^*} - \frac{1}{R} \right| \\ &= \left| \frac{2h(d_2 - d_1 - 4E_r)}{\sqrt{(h^2 + (d_1 + 2E_r)^2)(h^2 + (d_2 - 2E_r)^2)(4h^2 + (d_1 + d_2)^2)}} \right. \\ &\quad \left. - \frac{2h(d_2 - d_1)}{\sqrt{(h^2 + d_1^2)(h^2 + d_2^2)(4h^2 + (d_1 + d_2)^2)}} \right| \\ &\approx \frac{8E_r}{\sqrt{(h^2 + d_1^2)(h^2 + d_2^2)(4h^2 + (d_1 + d_2)^2)}} \\ &< \frac{4E_r}{h^2} \end{aligned}$$

and

$$\begin{aligned} |\kappa_f(x) - \kappa_g^*(x)| &= |\kappa_f(x) - (\kappa_g(x) + \Delta \kappa_g(x))| \\ &< hM_f + |\Delta \kappa_g(x)| \\ &< hM_f + \frac{4E_r}{h^2}. \end{aligned}$$

Thus, we conclude that the absolute error in the curvature computation is:

$$\Delta \kappa < hM_f + \frac{4E_r}{h^2},$$

and the relative error is:

$$\delta \kappa < R \left(hM_f + \frac{4E_r}{h^2} \right). \quad \square$$

About the Author—ZAIMING YANG is a software engineer at Johnson and Johnson's LifeScan division in Milpitas, California. He received his B.S. degree in electrical engineering from Shanghai University of Technology, Shanghai, China, in 1982, and his MS degree in computer science from University of California, Santa Barbara, in 1991. His research interests include computer vision and distributed systems.

About the Author—YUAN-FANG WANG received his bachelor's degree in electrical engineering from National Taiwan University in 1981, and his master's and Ph.D. degrees in electrical and computer engineering from the University of Texas at Austin in 1983 and 1987, respectively. He joined the Department of Computer Science at the University of California at Santa Barbara in 1987, where he is currently an Associate Professor. Dr. Wang's research activities center around computer vision, image processing, parallel processing, computer graphics and user interface design. Dr Wang is a member of IEEE Computer Society and Robotics and Automation Society. He served on various program committees and chaired several technical sessions in professional conferences. He has been a member of the IEEE PAMI Technical Committee since 1987.