1999 Reprinted with permission of Kluwer Academic Publishers,
from the International Journal of Computer Vision - Vol. 35, No. 2, Nov./Dec. 1999 - pp. 129-149

3D photography using shadows in dual-space geometry

Jean-Yves Bouguet† and Pietro Perona†‡

† California Institute of Technology, 136-93, Pasadena, CA 91125, USA
‡ Università di Padova, Italy
{bouguetj,perona}@vision.caltech.edu

Abstract

A simple and inexpensive approach for extracting the three-dimensional shape of objects is presented. It is based on 'weak structured lighting'. It requires very little hardware besides the camera: a light source (a desk-lamp or the sun), a stick and a checkerboard. The object, illuminated by the light source, is placed on a stage composed of a ground plane and a back plane; the camera faces the object. The user moves the stick in front of the light source, casting a moving shadow on the scene. The 3D shape of the object is extracted from the spatial and temporal location of the observed shadow. Experimental results are presented on five different scenes (indoor with a desk lamp and outdoor with the sun) demonstrating that the error in reconstructing the surface is less than 0.5% of the size of the object. A mathematical formalism is proposed that simplifies the notation and keep the algebra compact. A real-time implementation of the system is also presented.

1 Introduction and motivation

One of the most valuable functions of our visual system is informing us about the shape of the objects that surround us. Manipulation, recognition, and navigation are amongst the tasks that we can better accomplish by seeing shape. Ever-faster computers, progress in computer graphics, and the widespread expansion of the Internet have recently generated interest in imaging both the geometry and surface texture of objects. The applications are numerous. Perhaps the most important ones are animation and entertainment, industrial design, archiving, virtual visits to museums, and commercial on-line catalogues.

In designing a system for recovering shape, different engineering tradeoffs are proposed by each application. The main parameters to be considered are cost, accuracy, ease of use and speed of acquisition. So far the commercial 3D scanners (e.g. the Cyberware scanner) have emphasized accuracy over the other parameters. Active illumination systems are popular in industrial applications where a fixed installation with controlled lighting is possible. These systems use motorized transport of the object and active (laser, LCD projector) lighting of the scene which makes them very accurate, but unfortunately expensive [2, 23, 26, 38, 43]. Furthermore most active systems fail under bright outdoor scenes except those based upon synchronized scanning. One such system has been presented by Riou in [33].

An interesting challenge for vision scientists is to take the opposite point of view: emphasize low cost and simplicity and design 3D scanners that demand little more hardware than a PC and a video camera by making better use of the data that is available in the images.

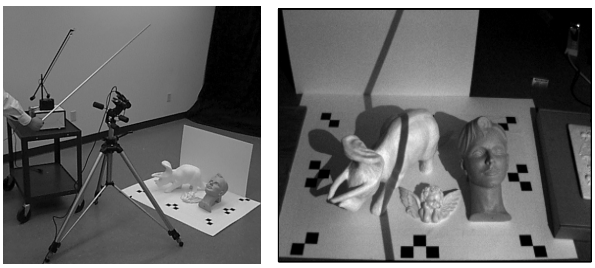


Figure 1: **The general setup of the proposed method:** The camera is facing the scene illuminated by the light source (top-left). The figure illustrates an indoor scenario when a desk lamp (without reflector) is used as light source. In outdoor the lamp is substituted by the sun. The objects to scan are positioned on the ground floor (horizontal plane), in front of a background plane. When an operator freely moves a stick in front of the light, a shadow is cast on the scene. The camera acquires a sequence of images $I(x, y, t)$ as the operator moves the stick so that the shadow scans the entire scene. A sample image is shown on the top right figure. This constitutes the input data to the 3D reconstruction system. The three dimensional shape of the scene is reconstructed using the spatial and temporal properties of the shadow boundary throughout the input sequence.

A number of passive cues have long been known to contain information on 3D shape: stereoscopic disparity, texture, motion parallax, (de)focus, shadows, shading and specularities, occluding contours and other surface discontinuities. At the current state of vision research stereoscopic disparity is the single passive cue that reliably gives reasonable accuracy. Unfortunately it has two major drawbacks: it requires two cameras thus increasing complexity and cost, and it cannot be used on untextured surfaces, which are common for industrially manufactured objects.

We propose a method for capturing 3D surfaces that is based on what we call 'weak structured lighting.' It yields good accuracy and requires minimal equipment besides a computer and a camera: a stick, a checkerboard, and a point light source. The light source may be a desk lamp for indoor scenes, and the sun for outdoor scenes. A human operator, acting as a low precision motor, is also required.

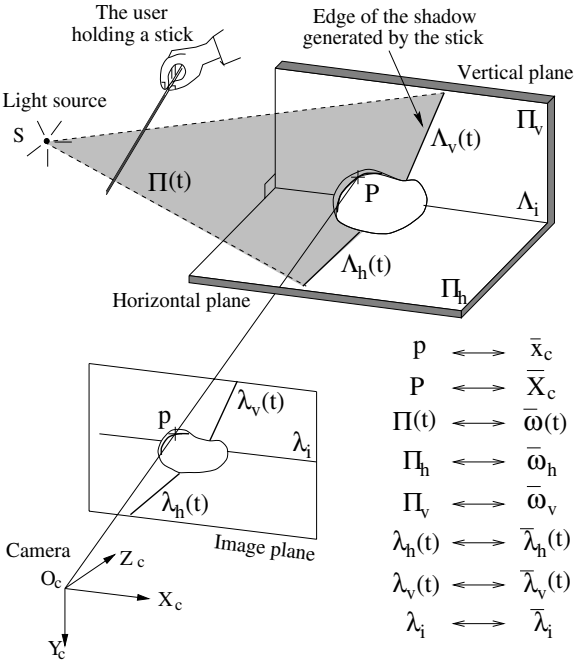


Figure 2: **Geometrical principle of the method**

We start with the description of the scanning method in Sec. 2, followed in Sec. 3 by a number of experiments that assess the convenience and accuracy of the system in indoor as well as outdoor scenarios. We end with a discussion and conclusions in Sec. 4. In addition, we show that expressing the problem in dual-space geometry [12] enables to explore and compute geometrical properties of three dimensional scenes with simple and compact notation. This formalism is discussed in the appendix together with a complete error analysis of the method.

2 Description of the method

The general principle consists of casting a moving shadow with a stick onto the scene, and estimating the three dimensional shape of the scene from the sequence of images of the deformed shadow. Figure 1 shows a typical setup. The objective is to extract scene depth at every pixel in the image. The point light source and the leading edge of the stick define, at every time instant, a plane; therefore, the boundary of the shadow that is cast by the stick on the scene is the intersection of this plane with the surface of the object. We exploit this geometrical insight for reconstructing the 3D shape of the object. Figure 2 illustrates the geometrical principle of the method. Approximate the light source with a point S, and denote by Π_h the horizontal plane (ground) and Π_v a vertical plane orthogonal to Π_h. Assume that the position of the plane Π_h in the camera reference frame is known from calibration (sec. 2.1). We infer the location of Π_v from the projection λ_i (visible in the image) of the intersection line Λ_i between Π_h and Π_v (sec. 2.2). The goal is to estimate the 3D location of the point P in space corresponding to every pixel p (of coordinates $\overline{x}_c$) in the image. Call t the time when the shadow boundary passes by a given pixel $\overline{x}_c$ (later referred to as the *shadow time*). Denote by $\Pi(t)$ the corresponding shadow plane at that time t. Assume that two portions of the shadow projected on the two planes Π_h and Π_v are visible on the image: $\lambda_h(t)$ and $\lambda_v(t)$. After extracting these two lines, we deduce the location in space of the two corresponding lines $\Lambda_h(t)$ and $\Lambda_v(t)$ by intersecting the planes $(O_c, \lambda_h(t))$ and $(O_c, \lambda_v(t))$ with Π_h and Π_v respectively. The shadow plane $\Pi(t)$ is then the plane defined by the two non-collinear lines $\Lambda_h(t)$ and $\Lambda_v(t)$ (sec. 2.5). Finally, the point P corresponding to $\overline{x}_c$ is retrieved by intersecting $\Pi(t)$ with the optical ray (O_c, p). This final stage is called triangulation (sec. 2.6). Notice that the key steps are: (a) estimate the shadow time $t_s(\overline{x}_c)$ at every pixel $\overline{x}_c$ (*temporal processing*), (b) locate the two reference lines $\lambda_h(t)$ and $\lambda_v(t)$ at every time instant t (*spatial processing*), (c) calculate the shadow plane, and (d) triangulate and calculate depth. These tasks are described in sections 2.4, 2.5 and 2.6.

Goshtasby *et al.* [22] also designed a range scanner using a shadow generated by a fine wire in order to reconstruct the shape of dental casts. In their system, the wire was motorized and its position calibrated.

Notice that if the light source is at a known location in space, then the shadow plane $\Pi(t)$ may be directly inferred from the point S and the line $\Lambda_h(t)$. Consequently, in such cases, the additional plane $\Pi_v(t)$ is not required. We describe here two versions of the setup: one containing two calibrated planes and an uncalibrated (possibly moving) light source; the second containing one calibrated plane and a calibrated light source.

2.1 Camera calibration

The goal of calibration is to recover the location of the ground plane Π_h and the *intrinsic* camera parameters (focal length, principal point and radial distortion factor). The procedure consists of first placing a planar checkerboard pattern on the ground in the location of the objects to scan (see figure 3-left). From the image captured by the camera (figure 3-right), we infer the intrinsic and extrinsic parameters of the camera, by matching the projections onto the image plane of the known grid corners with the expected projection directly measured on the image (extracted corners of the grid); the method is proposed by Tsai in [39]. We use a first order symmetric radial distortion model for the lens, as proposed in [11, 39, 25]. When using a single image of a planar calibration rig, the principal point (i.e. the intersection of the optical axis with the image plane) cannot be recovered [25, 37]. In that case it is assumed to be identical to the image center. In order to fit a full camera model (principal

point included), we propose to extend that approach by integrating multiple images of the planar grid positioned at different locations in space (with different orientations). This method has been suggested, studied and demonstrated by Sturm and Maybank in [37]. Theoretically, a minimum of two images is required to recover two focals (along x and y), the principal point coordinates, and the lens distortion factor. We recommend to use that method with three or four images for best accuracies on the intrinsic parameters [37]. In our experience, in order to achieve good 3D reconstruction accuracies, it is sufficient to use the simple approach with a single calibration image without estimating the camera principal point. In other words, the quality of reconstruction is quite insensitive to errors on the principal point position. A whole body of work supporting that observation may be found in the literature. We especially advise the reader most interested in issues on sensitivity of 3D Euclidean reconstruction results with respect to intrinsic calibration errors, to refer to recent publications on self-calibration, such as Bougnoux [5] or Pollefeys *et al.* [28, 31, 32].

For more general insights on calibration techniques, we refer the reader to the work of Faugeras [19] and others [10, 11, 14, 18, 36, 42]. A 3D rig should be used for achieving maximum accuracy.

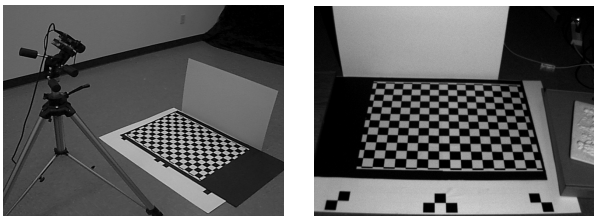


Figure 3: **Camera calibration**

2.2 Vertical plane localization Π_v

Call $\overline{\omega}_h$ and $\overline{\omega}_v$ respectively the coordinate vectors of Π_h and Π_v (refer to figure 2 and Appendix A for notation). After calibration, $\overline{\omega}_h$ is known. The two planes Π_h and Π_v intersect along the line Λ_i observed on the image plane at λ_i. Therefore, according to proposition 1 in Appendix A, $\overline{\omega}_h - \overline{\omega}_v$ is parallel to $\overline{\lambda}_i$, coordinate vector of λ_i, or equivalently, there exists a scalar α such that $\overline{\omega}_v = \overline{\omega}_h + \alpha\overline{\lambda}_i$. Since the two planes Π_h and Π_v are by construction orthogonal, we have $\langle \overline{\omega}_h, \overline{\omega}_v \rangle = 0$. That leads to the closed-form expression for calculating $\overline{\omega}_v$:

$$\overline{\omega}_v = \overline{\omega}_h - \frac{\langle \overline{\omega}_h, \overline{\omega}_h \rangle}{\langle \overline{\lambda}_i, \overline{\omega}_h \rangle} \overline{\lambda}_i.$$

Notice that in every realistic scenario, the two planes Π_h and Π_v do not contain the camera center O_c. Their coordinate vectors $\overline{\omega}_h$ and $\overline{\omega}_v$ in dual-space are therefore always well defined (see Appendix A and sections 2.6 and 2.7 for further discussions).

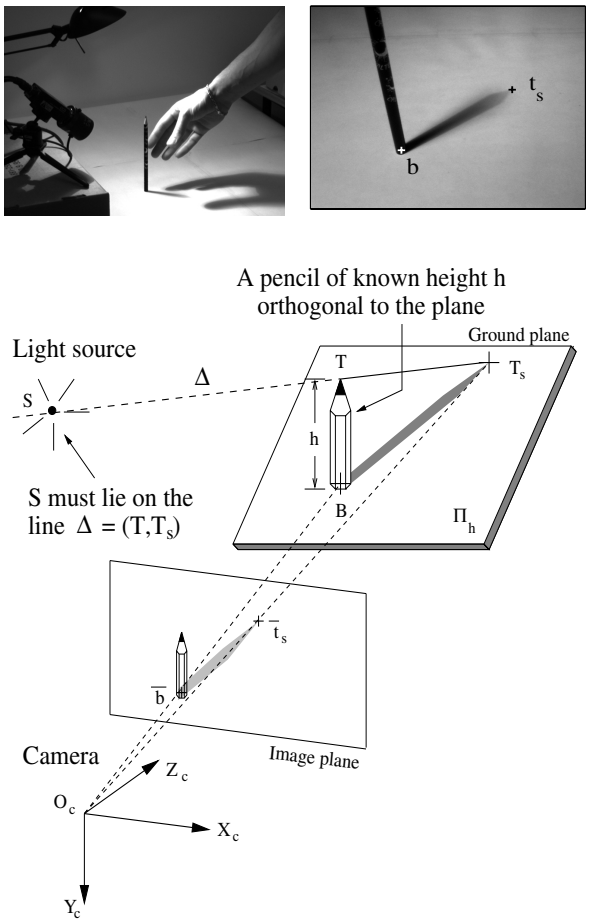


Figure 4: **Light source calibration**

2.3 Light source calibration

When using a single reference plane for scanning (for example Π_h without Π_v), it is required to know the location of the light source in order to infer the shadow plane location $\Pi(t)$ (see section 2.5 for details). Figure 4 illustrates a simple technique for calibrating the light source that requires minimal extra equipment: a pencil of known length. The operator stands a pencil on the reference plane Π_h (see fig. 4-top-left). The camera observes the shadow of the pencil projected on the ground plane. The acquired image is shown on figure 4-top-right. From the two points $\overline{b}$ and $\overline{t}_s$ on this image, one can infer the positions in space of B and T_s, respectively the base of the pencil, and the tip of the pencil shadow (see bottom figure). This is done by intersecting the optical rays $(O_c, \overline{b})$ and $(O_c, \overline{t}_s)$ with Π_h (known from camera calibration). In addition, given that the height of the pencil h is known, the coordinates of its tip T can be directly inferred from B. The point light source S has to lie on the line $\Delta = (T, T_s)$ in space. This yields one linear constraint on the light source position. By taking a second view, with the pencil at a

different location on the plane, one retrieves a second independent constraint with another line Δ'. A closed form solution for the 3D coordinate of S is then derived by intersecting the two lines Δ and Δ' (in the least squares sense). Notice that since the problem is linear, one can integrate the information from more than 2 views and make the estimation more accurate. If $N > 2$ images are used, one can obtain a closed form solution for the closest point $\tilde{S}$ to the N inferred lines (in the least squares sense). We also estimate the uncertainty on that estimate from the distance of $\tilde{S}$ to each one of the Δ lines. That indicates how consistently the lines intersect a single point in space. Refer to [7, 8, 6] for the complete derivations.

2.4 Spatial and temporal shadow edge localization

A fundamental stage of the method is the detection of the lines of intersection of the shadow plane $\Pi(t)$ with the two planes Π_h and Π_v; a simple approach to extract $\overline{\lambda}_h(t)$ and $\overline{\lambda}_v(t)$ may be used if we make sure that a number of pixel rows at the top and bottom of the image are free from objects. Then the two tasks to accomplish are: **(a)** Localize the edges of the shadow that are directly projected on the two orthogonal planes $\lambda_h(t)$ and $\lambda_v(t)$ at every discrete time t (every frame), leading to the set of all shadow planes $\Pi(t)$ (see sec. 2.5), **(b)** Estimate the time $t_s(\overline{x}_c)$ (*shadow time*) where the edge of the shadow passes through any given pixel $\overline{x}_c = (x_c, y_c)$ in the image (see figure 5). Curless and Levoy [16] demonstrated that such a spatio-temporal approach is appropriate for preserving sharp discontinuities in the scene as well as reducing range distortions. A similar temporal processing for range sensing was used by Gruss, Tada and Kanade in [23, 27].

Both processing tasks correspond to finding the edge of the shadow, but the search domains are different: one operates on the spatial coordinates (image coordinates) and the other one on the temporal coordinate. Although independent in appearance, the two search procedures need to be compatible: if at time t_0 the shadow edge passes through pixel $\overline{x}_c = (x_c, y_c)$, the two searches should find the exact same point (x_c, y_c, t_0) (in space/time). One could observe that this property does not hold for all techniques. One example is the image gradient approach for edge detection (e.g. Canny edge detector [13]). Indeed, the maximum spatial gradient point does not necessarily match with the maximum temporal gradient point (which is function of the scanning speed). In addition, the spatial gradient is a function both of changes in illumination due to the shadow and changes in albedo and changes in surface orientation. Furthermore, it is preferable to avoid any spatial filtering on the images (e.g. smoothing) which would produce blending in the final depth estimates, especially noticeable at

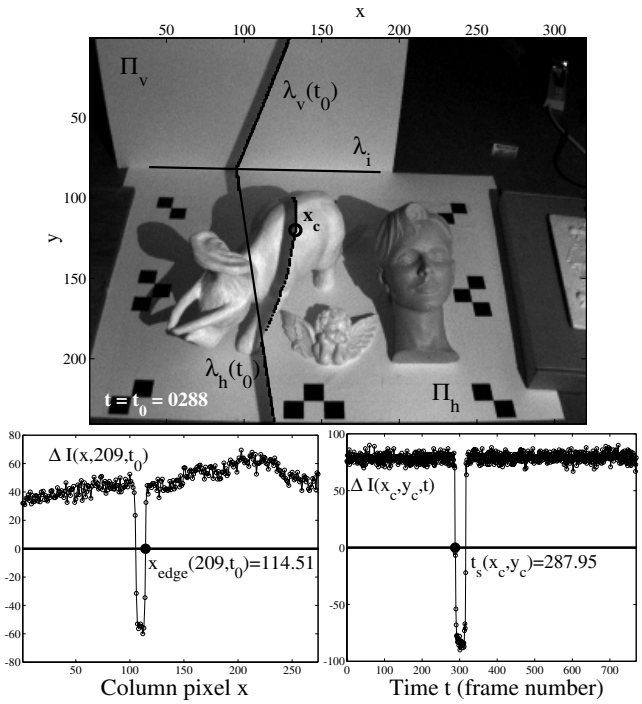


Figure 5: **Spatial and temporal shadow localization**

occlusions and surface discontinuities (corners for example). These observations were also addressed by Curless and Levoy in [16].

It is therefore necessary to use a unique criterion that would equally describe shadow edges in space (image coordinates) and time and is insensitive to changes in surface albedo and surface orientation. The simple technique we propose here that satisfies that property is spatio-temporal thresholding. This is based on the following observation: as the shadow is scanned across the scene, each pixel (x, y) sees its brightness intensity going from an initial maximum value $I_{\max}(x, y)$ (when there is no shadow yet) down to a minimum value $I_{\min}(x, y)$ (when the pixel is within the shadow) and then back up to its initial value as the shadow goes away. This profile is characteristic even when there is a fair amount of internal reflections in the scene [29, 41].

For any given pixel $\overline{x}_c = (x, y)$, define $I_{\min}(x, y)$ and $I_{\max}(x, y)$ as its minimum and maximum brightness throughout the entire sequence:

$$\begin{cases} I_{\min}(x,y) & \doteq & \min\limits_t \{I(x,y,t)\} \\ I_{\max}(x,y) & \doteq & \max\limits_t \{I(x,y,t)\} \end{cases} \quad (1)$$

We define the shadow edge to be the locations (in space-time) where the image $I(x, y, t)$ intersects with the threshold image $I_{\text{shadow}}(x, y)$ defined as the mean value between $I_{\max}(x, y)$ and $I_{\min}(x, y)$:

$$I_{\text{shadow}}(x,y) \doteq \frac{1}{2}\left(I_{\max}(x,y) + I_{\min}(x,y)\right) \quad (2)$$

This may be also regarded as the zero crossings of the difference image $\Delta I(x, y, t)$ defined as follows:

$$\Delta I(x, y, t) \doteq I(x, y, t) - I_{\text{shadow}}(x, y) \qquad (3)$$

The two bottom plots of fig. 5 illustrate shadow edge detection in the spatial domain (to calculate $\lambda_h(t)$ and $\lambda_v(t)$) and in the temporal domain (to calculate $t_s(\overline{x}_c)$). The bottom-left plot shows the profile of $\Delta I(x, y, t)$ along row $y = 209$ at time $t = t_0 = 288$ versus the column pixel coordinate x. The second zero crossing of that profile corresponds to one point $\overline{x}_{\text{edge}}(t_0) = (114.51, 209)$ belonging to $\lambda_h(t_0)$, the right edge of the shadow (computed at subpixel accuracy by linear interpolation). Identical processing is applied on 39 other rows for $\lambda_h(t_0)$ and 70 rows for $\lambda_v(t_0)$ in order to retrieve the two edges (by least square line fitting across the two sets of points on the image). Similarly, the bottom-right figure shows the temporal profile $\Delta I(x_c, y_c, t)$ at the pixel $\overline{x}_c = (x_c, y_c) = (133, 120)$ versus time t (or frame number). The shadow time at that pixel is defined as the first zero crossing location of that profile: $t_s(133, 120) = 287.95$ (computed at sub-frame accuracy by linear interpolation). Notice that the right edge of the shadow corresponds to the front edge of the temporal profile, because the shadow was scanned from left to right in all experiments. Intuitively, pixels corresponding to occluded regions in the scene do not provide any relevant depth information. Therefore, we only process pixels with contrast value $I_{\text{contrast}}(x, y) \doteq I_{\max}(x, y) - I_{\min}(x, y)$ larger than a pre-defined threshold I_{thresh}. This threshold was 30 in all experiments reported in this paper (recall that the intensity values are encoded from 0 for black to 255 for white). This threshold should be proportional to the level of noise in the image.

Due to the limited dynamic range of the camera, it is clear that one should avoid saturating the sensor, and that one would expect poor accuracy in areas of the scene that reflect little light towards the camera due to their orientation with respect to the light source and/or low albedo. Our experiments were designed to test the extent of this problem.

2.5 Shadow plane estimation $\Pi(t)$

Denote by $\overline{\omega}(t)$, $\overline{\lambda}_h(t)$ and $\overline{\lambda}_v(t)$ the coordinate vectors of the shadow plane $\Pi(t)$ and of the shadow edges $\lambda_h(t)$ and $\lambda_v(t)$ at time t. Since $\lambda_h(t)$ is the projection of the line of intersection $\Lambda_h(t)$ between $\Pi(t)$ and Π_h, then $\overline{\omega}(t)$ lies on the line passing through $\overline{\omega}_h$ with direction $\overline{\lambda}_h(t)$ in dual-space (from Appendix A). That line, denoted $\hat{\Lambda}_h(t)$, is the dual image of $\Lambda_h(t)$ in dual-space (see Appendix A). Similarly, $\overline{\omega}(t)$ lies on the line $\hat{\Lambda}_v(t)$ passing through $\overline{\omega}_v$ with direction $\overline{\lambda}_v(t)$ (dual image of $\Lambda_v(t)$). Therefore, in dual-space, the coordinate vector of the shadow plane $\overline{\omega}(t)$ is at the intersection between the two known lines $\hat{\Lambda}_h(t)$ and $\hat{\Lambda}_v(t)$. In the presence of noise these two lines will not exactly intersect (equivalently, the 3 lines λ_i, $\lambda_h(t)$ and $\lambda_v(t)$ do not necessarily intersect at one point on the image plane, or their coordinate vectors $\overline{\lambda}_i$, $\overline{\lambda}_h(t)$ and $\overline{\lambda}_v(t)$ are not coplanar). However, one may still identify $\overline{\omega}(t)$ with the point that is the closest to the lines in the least-squares sense. The solution to that problem reduces to:

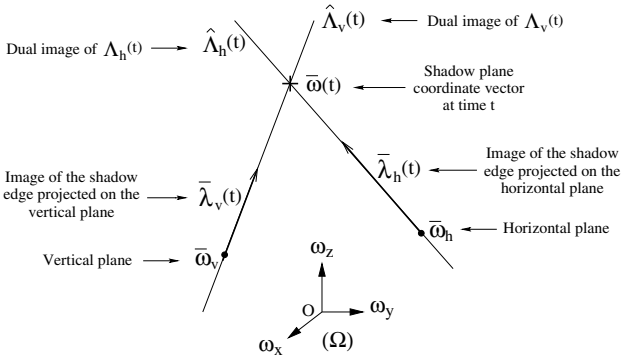


Figure 6: **Shadow plane estimation using two planes:** The coordinate vector of the shadow plane $\overline{\omega}(t)$ is the intersection point of the two dual lines $\hat{\Lambda}_h(t)$ and $\hat{\Lambda}_v(t)$ in dual-space (Ω). In presence of noise, the two lines do not intersect. The vector $\overline{\omega}(t)$ is then the best intersection point between the two lines (in the least squares sense).

$$\overline{\omega}(t) = \frac{1}{2}\left(\overline{\omega}_1(t) + \overline{\omega}_2(t)\right), \qquad (4)$$

with

$$\begin{array}{rcl} \overline{\omega}_1(t) & \doteq & \overline{\omega}_h + \alpha_h \overline{\lambda}_h(t) \\ \overline{\omega}_2(t) & \doteq & \overline{\omega}_v + \alpha_v \overline{\lambda}_v(t) \end{array} \qquad (5)$$

if $[\alpha_h \quad \alpha_v]^T = \mathbf{A}^{-1}\mathbf{b}$, where $\mathbf{A}$ and $\mathbf{b}$ are defined as follows (for clarity, the variable t is omitted):

$$\mathbf{A} \doteq \begin{bmatrix} \langle \overline{\lambda}_h, \overline{\lambda}_h \rangle & -\langle \overline{\lambda}_h, \overline{\lambda}_v \rangle \\ -\langle \overline{\lambda}_h, \overline{\lambda}_v \rangle & \langle \overline{\lambda}_v, \overline{\lambda}_v \rangle \end{bmatrix}, \quad \mathbf{b} \doteq \begin{bmatrix} \langle \overline{\lambda}_h, \overline{\omega}_v - \overline{\omega}_h \rangle \\ \langle \overline{\lambda}_v, \overline{\omega}_h - \overline{\omega}_v \rangle \end{bmatrix}$$

Note that the two vectors $\overline{\omega}_1(t)$ and $\overline{\omega}_2(t)$ are the orthogonal projections, in dual-space, of $\overline{\omega}(t)$ onto $\hat{\Lambda}_h(t)$ and $\hat{\Lambda}_v(t)$ respectively. The norm of the difference between these two vectors may be used as an estimate of the error in recovering $\Pi(t)$. If the two edges $\lambda_h(t)$ and $\lambda_v(t)$ are estimated with different reliabilities, a weighted least squares method may still be used.

Figure 6 illustrates the principle of shadow plane estimation in dual-space when using the two edges $\lambda_h(t)$ and $\lambda_v(t)$. This reconstruction method was used in experiments 1, 4 and 5.

Notice that the additional vertical plane Π_v enables us to extract the shadow plane location without requiring the knowledge of the light source position.

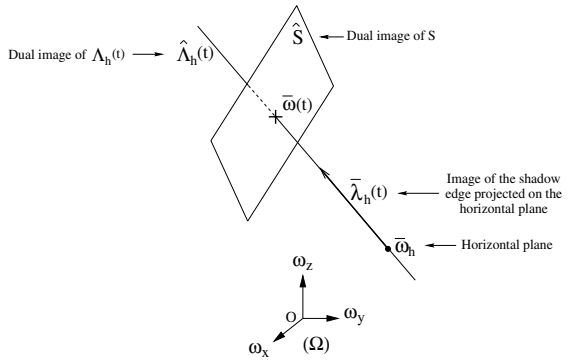


Figure 7: **Shadow plane estimation using one plane and the light source position:** In dual-space, the coordinate vector of the shadow plane $\overline{\omega}(t)$ is the intersection point of the line $\hat{\Lambda}_h(t)$ and the plane $\hat{S}$, dual image of the point light source S. This method requires the knowledge of the light source position. A light source calibration method is presented in section 2.3.

Consequently, the light source is allowed to move during the scan (this may be the case of the sun, for example).

When the light source is of fixed and known location in space, the plane Π_v is not required. Then, one may directly infer the shadow plane position from the line $\lambda_h(t)$ and from the light source position S:

$$\overline{\omega}(t) = \overline{\omega}_h + \alpha_h \overline{\lambda}_h(t) \tag{6}$$

where

$$S \in \Pi(t) \quad \Leftrightarrow \quad \langle \overline{\omega}(t), \overline{X}_S \rangle = 1 \quad \Leftrightarrow \quad \alpha_h = \frac{1 - \langle \overline{\omega}_h, \overline{X}_S \rangle}{\langle \overline{\lambda}_h(t), \overline{X}_S \rangle}$$

where $\overline{X}_S = [X_S \quad Y_S \quad Y_S]^T$ is the coordinate vector of the light source S in the camera reference frame. In dual-space geometry, this corresponds to intersecting the line $\hat{\Lambda}_h(t)$ with the plane $\hat{S}$, dual image of the source point S. This process is illustrated in figure 7. Notice that $\langle \overline{\lambda}_h(t), \overline{X}_S \rangle = 0$ corresponds to the case where the shadow plane contains the camera center of projection O_c. This is singular configuration that makes the triangulation fail ($\|\overline{\omega}(t)\| \to \infty$). This approach requires an additional step of estimating the position of S. Section 2.3 describes a simple method for light source calibration. This reconstruction method was used in experiments 2 and 3.

It is shown in Appendix B that $1 - \langle \overline{\omega}_h, \overline{X}_S \rangle = h_S / d_h$ where h_S and d_h are the orthogonal distances of the light source S and the camera center O_c to the plane Π_h (see figure 8). Then, the constant α_h may be written as:

$$\alpha_h = \frac{h_S/d_h}{\langle \overline{\lambda}_h(t), \overline{X}_S \rangle} = \frac{1/d_h}{\langle \overline{\lambda}_h(t), \overline{X}_S/h_S \rangle} \tag{7}$$

This expression highlights the fact that the algebra naturally generalizes to cases where the light source is located at infinity (and calibrated). Indeed, in those cases, the ratio $\overline{X}_S/h_S$ reduces to $\overline{d}_S/\sin\phi$ where $\overline{d}_S$ is the normalized light source direction vector (in the camera reference frame) and ϕ the elevation angle of the light source with respect to the plane Π_h (defined on figure 8). In dual-space, the construction of the shadow plane vector $\overline{\omega}(t)$ remains the same: it is still at the intersection of $\hat{\Lambda}_h(t)$ with $\hat{S}$. The only difference is that the dual image $\hat{S}$ is a plane crossing the origin in dual-space. The surface normal of that plane is simply the vector $\overline{d}_S$.

2.6 Triangulation

Once the shadow time $t_s(\overline{x}_c)$ is estimated at a given pixel $\overline{x}_c = [x_c \quad y_c \quad 1]^T$ (in homogeneous coordinates), one can identify the corresponding shadow plane $\Pi(t_s(\overline{x}_c))$ (with coordinate vector $\overline{\omega}_c \doteq \overline{\omega}(t_s(\overline{x}_c))$). Then, the point P in space associated to $\overline{x}_c$ is retrieved by intersecting $\Pi(t_s(\overline{x}_c))$ with the optical ray $(O_c, \overline{x}_c)$ (see figure 2):

$$Z_c = \frac{1}{\langle \overline{\omega}_c, \overline{x}_c \rangle} \quad \Longrightarrow \quad \overline{X}_c = Z_c \, \overline{x}_c = \frac{\overline{x}_c}{\langle \overline{\omega}_c, \overline{x}_c \rangle}, \tag{8}$$

if $\overline{X}_c = [X_c \quad Y_c \quad Z_c]^T$ is defined as the coordinate vector of P in the camera reference frame.

Notice that the shadow time $t_s(\overline{x}_c)$ acts as an index to the shadow plane list $\Pi(t)$. Since $t_s(\overline{x}_c)$ is estimated at sub-frame accuracy, the plane $\Pi(t_s(\overline{x}_c))$ (actually its coordinate vector $\overline{\omega}_c$) results from linear interpolation between the two planes $\Pi(t_0 - 1)$ and $\Pi(t_0)$ if $t_0 - 1 < t_s(\overline{x}_c) < t_0$ and t_0 integer:

$$\overline{\omega}_c = \Delta t \, \overline{\omega}(t_0 - 1) + (1 - \Delta t) \, \overline{\omega}(t_0),$$

where $\Delta t = t_0 - t_s(\overline{x}_c)$, $0 \leq \Delta t < 1$ (see figure 17).

Once the range data are recovered, a mesh is generated by connecting neighboring points in triangles. The connectivity is directly given by the image: two vertices are neighbors if their corresponding pixels are neighbors in the image. In addition, since every vertex corresponds to a unique pixel, texture mapping is also a straightforward task. Figures 9, 11, 12, 13 and 14 show experimental results.

Similarly to stereoscopic vision, when the baseline becomes shorter, as the shadow plane moves closer to the camera center triangulation becomes more and more sensitive to noise. In the limit, if the plane crosses the origin (or equivalently $\|\overline{\omega}_c\| \to \infty$) triangulation becomes impossible. This is why it is not a big loss that we cannot represent planes going through the origin with our parameterization. This observation will appear again in the next section on error analysis.

2.7 Design Issues - Error analysis

When designing the scanning system, it is important to choose a spatial configuration of the camera and the light source that maximizes the overall quality of reconstruction of the scene.

The analysis conducted in Appendix C leads to an expression for the variance $\sigma^2_{Z_c}$ of the error of the depth estimate Z_c of a point P belonging to the scene (equation 18):

$$\sigma^2_{Z_c} = Z_c^4 \left(\frac{\omega_x \cos\varphi + \omega_y \sin\varphi}{f_c \, \|\overline{\nabla} I(\overline{x}_c)\|} \right)^2 \sigma_I^2 \qquad (9)$$

where $\overline{x}_c$ is the coordinate vector of the projection p of P on the image plane, $\overline{\omega}_c = [\omega_x \quad \omega_y \quad \omega_z]^T$ is the shadow plane vector at time $t = t_s(\overline{x}_c)$, $\overline{\nabla} I(\overline{x}_c) = [I_x(\overline{x}_c) \quad I_y(\overline{x}_c)]^T = \|\overline{\nabla} I(\overline{x}_c)\| \, [\cos\varphi \quad \sin\varphi]^T$ is the spatial gradient vector of the image brightness at the shadow edge at $\overline{x}_c$ at time $t = t_s(\overline{x}_c)$ (in units of brightness per pixel), σ_I is the standard deviation of the image brightness noise (in units of brightness), and f_c is the camera focal length (in pixels).

Three observations may be drawn from equation 9. First, since $\sigma^2_{Z_c}$ is inversely proportional to $\|\overline{\nabla} I(\overline{x}_c)\|^2$, the reconstruction accuracy increases with the sharpness of the shadow boundary. This behavior has been observed in past experiments, and discussed in [8]. This might explain why scans obtained using the sun (experiments 4 and 5) are more noisy that those with a desk lamp (as the penumbra is larger with the sun by a factor of approximately 5). Second, notice that $\sigma^2_{Z_c}$ is proportional to $\|\overline{\omega}_c\|^2$ (through the terms ω_x^2 and ω_y^2), or, equivalently, inversely proportional to the square of the distance of the shadow plane to the camera center O_c. Therefore, as the shadow plane moves closer to the camera, triangulation becomes more and more sensitive to noise (see discussion in section 2.6). The third observation is less intuitive: one may notice that σ_{Z_c} does not explicitly depend on the local shadow speed at $\overline{x}_c$, at time $t = t_s(\overline{x}_c)$. Therefore, decreasing the scanning speed would not increase accuracy. However, for the analysis leading to equation 9 to remain valid (see Appendix C), the temporal pixel profile must be sufficiently sampled within the transition area of the shadow edge (the penumbra). Therefore, if the shadow edge were sharper, the scanning should also be slower so that the temporal profile at every pixel would be properly sampled. Decreasing further the scanning speed would benefit the accuracy only if the temporal profile were appropriately low-pass filtered or otherwise interpolated before extraction of $t_s(\overline{x}_c)$. This is an issue for future research.

An experimental validation of the variance expression (9) is reported in section 3 (figure 10).

In the case where the light source position is known, it is possible to write the "average" depth variance as a direct function of the variables defining the geometry of the system (Appendix C, equation 22):

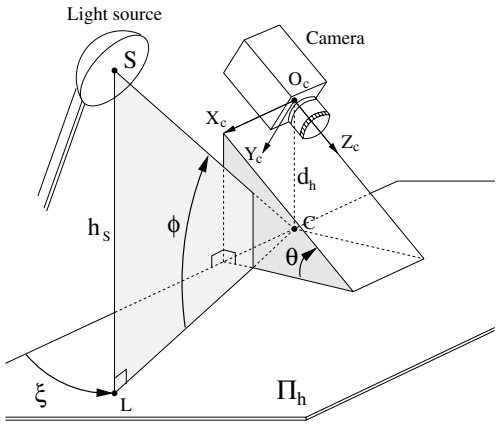


Figure 8: **Geometric setup:** The camera is positioned at a distance d_h away from the plane Π_h and tilted down towards it at an angle θ. The light source is located at a height h_S, with its direction defined by the azimuth and elevation angles ξ and ϕ in the reference frame attached to the plane Π_h. Notice that the sign of $\cos\xi$ directly relates to which side of the camera the lamp is standing: positive on the right, and negative on the left.

$$\sigma_{Z_c}|_{\text{average}} \approx d_h \, \frac{\tan\phi}{\sin^2\theta \, |\cos\xi|} \, \frac{\sigma_I}{f_c \, |I_x(\overline{x}_c)|} \qquad (10)$$

where the quantities d_h, θ, ϕ and ξ characterize the spatial configuration of the camera and the light source with respect to the reference plane Π_h (figure 8). Notice that this quantity may even be computed prior to scanning right after calibration.

In order to maximize the overall reconstruction quality, the position of the light source needs then to be chosen so that the norm of the ratio $\tan\phi / \cos\xi$ is minimized. Therefore, the two optimal values for the azimuth angle are $\xi = 0$ and $\xi = \pi$ corresponding to positioning the lamp either to the right ($\xi = 0$) or to the left ($\xi = \pi$) of the camera (see figure 8). Regarding the elevation angle ϕ, it would be beneficial to make $\tan\phi$ as small as possible. However, making ϕ arbitrarily small is not practically possible. Indeed, setting $\phi = 0$ would constrain the light source to lie on the plane Π_h which would then drastically reduce the effective coverage of the scene due to large amount of self-shadows cast on the scenery. A reasonable tradeoff for ϕ is roughly between 60 and 70 degrees. Regarding the camera position, it would also be beneficial to make $\sin\theta$ as large as possible (ideally equal to one). However, it is very often not practical to make θ arbitrarily close to $\pi/2$. Indeed, having $\theta = \pi/2$ brings the reference plane Π_h parallel to the image plane. Then, standard visual camera calibration algorithms are known to fail (due to lack of depth perspective in the image). In most experiments, we set θ to roughly $\pi/4$.

Once again, for validation purposes, we may use

equation 10 to estimate the reconstruction error of the scans performed in experiment 3 (figure 12). From a set of 10 images, we first estimate the average image brightness noise ($\sigma_I = 2$), and the shadow edge sharpness ($\|\overline{\nabla} I\| \approx 50$). After camera and light source calibration, the following set of parameters is recovered: $f_c = 428$ pixels, $d_h = 22$ cm, $\theta = 39.60$ degrees, $h_S = 53.53$ cm, $\xi = -4.91$ degrees and $\phi = 78.39$ degrees. Equation 10 returns then an estimate of the reconstruction error ($\sigma_{Z_c} \approx 0.2$ mm) very close to the actual error experimentally measured on the final reconstructed surface (between 0.1 mm and 0.2 mm). The first expression given in equation 9 may also be used for obtaining a more accurate estimate of σ_{Z_c} specific to every point in the scene.

2.8 Merging scans

The range data can only be retrieved at pixels corresponding to regions in the scene illuminated by the light source and seen by the camera. In order to obtain better coverage of the scene, one may take multiple scans of the same scene having the light source at different locations each time, while keeping the camera position fixed. Consider the case of two scans with the lamp first on the right, and then on the left of the camera (see figure 9). Assume that at a given pixel $\overline{x}_c$ on the image, two shadow planes are available from the two scans: Π_c^L and Π_c^R. Denote by $\overline{\omega}_c^L$ and $\overline{\omega}_c^R$ their respective coordinate vectors. Then, two estimates Z_c^L and Z_c^R of the corresponding depth at $\overline{x}_c$ are available (from equation 8):

$$\left\{ \begin{array}{rcl} Z_c^L & = & 1 / \left\langle \overline{\omega}_c^L, \overline{x}_c \right\rangle \\ Z_c^R & = & 1 / \left\langle \overline{\omega}_c^R, \overline{x}_c \right\rangle \end{array} \right. \tag{11}$$

One may then calculate the depth estimate at $\overline{x}_c$ by taking a weighted average of Z_c^L and Z_c^R:

$$Z_c \doteq \alpha_L \, Z_c^L + \alpha_R \, Z_c^R \tag{12}$$

where the weights α_L and α_R are computed based on the respective reliabilities of the two depth estimates. Assuming that Z_c^L and Z_c^R are random variables with independent noise terms, they are optimally averaged (in the minimum variance sense) using the inverse of the variances as weights [30]:

$$\frac{\alpha_L}{\alpha_R} = \frac{\sigma_R^2}{\sigma_L^2} \implies \left\{ \begin{array}{l} \alpha_L = \sigma_R^2 / (\sigma_R^2 + \sigma_L^2) \\ \alpha_R = \sigma_L^2 / (\sigma_R^2 + \sigma_L^2) \end{array} \right. \tag{13}$$

where σ_L^2 and σ_R^2 are the variances of the error attached to Z_c^L and Z_c^R respectively.

A sensitivity analysis of the method described in Appendix C provides expressions for those variances (given in equation 9). This technique was used in experiment 1 for merging two scans (see figure 9).

2.9 Real-time implementation

We implemented a real-time version of our 3D scanning algorithm in collaboration with Silvio Savarese of the university of Naples, Italy. In that implementation the process is divided into two main steps. In the first step, the minimum and maximum images $I_{\min}(x, y)$ and $I_{\max}(x, y)$ (eq. 1) are computed through a first fast shadow sweep over the scene (with no shadow edge detection). That step allows to pre-compute the threshold image $I_{\text{shadow}}(x, y)$ (eq. 2) which is useful to compute in real-time the difference image $\Delta I(x, y, t)$ (eq. 3) during the next stage: the scanning procedure itself. During scanning, temporal and spatial shadow edge detections are performed as described in section 2.4: As a new image $I(x, y, t_0)$ is acquired at time $t = t_0$, the corresponding difference image $\Delta I(x, y, t_0)$ is first computed. Then, a given pixel (x_c, y_c) is selected as a pixel lying on the edge of the shadow if $\Delta I(x_c, y_c, t)$ crosses zero between times $t = t_0 - 1$ and $t = t_0$. In order to make that decision, and then compute its corresponding sub-frame shadow time $t_s(x_c, y_c)$, only the previous image difference $\Delta I(x, y, t_0 - 1)$ needs to be stored in memory. Once a pixel (x_c, y_c) is activated and its sub-frame shadow time $t_s(x_c, y_c)$ computed, one may directly identify its corresponding shadow plane Π by linear interpolation between the current shadow plane $\Pi(t_0)$ and the previous one $\Pi(t_0 - 1)$ (see sec. 2.5). Therefore, the 3D coordinates of the point may be directly computed by triangulation (see sec. 2.6). As a result, in that implementation, neither the shadow times $t_s(x, y)$, nor the entire list of shadow planes $\Pi(t)$ need to be stored in memory, only the previous difference image $\Delta I(x, y, t_0 - 1)$ and the previous shadow plane $\Pi(t_0 - 1)$. In addition, scene depth map (or range data) is computed in real-time. The final implementation that we designed also takes advantage of possible multiple passes of the shadow edge over a given pixel in the image by integrating all the successive depth measurements together based on their relative reliabilities (equations 11, 12 and 13 in section 2.8). Details of the implementation may be found in [34].

The real-time program was developed under Visual C++ and works at 30 frames a second on images of size 320×240 on a Pentium 300MHz machine: it takes approximately 30 seconds to scan a scene with a single shadow pass (i.e. $30 \times 30 = 900$ frames), and between one and two minutes for a refined scan using multiple shadow passes. The system uses the PCI frame grabber PXC200 from Imagenation, a NTSC black and white SONY XC-73/L camera (1/3 inch CCD) with a 6mm COSMICAR lens (leading to a 45^o horizontal field of view). Source code (`matlab` for calibration and `C` for scanning) and complete hardware references and specifications are available online at `http://www.vision.caltech.edu/bouguetj/ICCV98`. At the same location, a short demonstration movie of

the working system is also available.

3 Experimental Results

3.1 Calibration accuracy

Camera calibration. For a given setup, we acquired 5 images of the checkerboard pattern (see figure 3-right), and performed independent calibrations on them. The checkerboard, placed at different positions in each image, consisted of 187 visible corners on a 16×10 grid. We computed both mean values and standard deviations of all the parameters independently: the focal length f_c, radial distortion factor k_c and ground plane position Π_h. Regarding the ground plane position, it is convenient to look at its distance d_h to the camera origin O_c and its normal vector $\overline{n}_h$ expressed in the camera reference frame (recall: $\overline{\omega}_h = \overline{n}_h / d_h$). The following table summarizes the calibration results:

Parameters	Estimates	Relative errors
f_c (pixels)	426.8 ± 0.8	0.2%
k_c	-0.233 ± 0.002	1%
d_h (cm)	112.1 ± 0.1	0.1%
$\overline{n}_h$	$\begin{pmatrix} -0.0529 \pm 0.0003 \\ 0.7322 \pm 0.0003 \\ 0.6790 \pm 0.0003 \end{pmatrix}$	0.05%
$\overline{\omega}_h$ (m^{-1})	$\begin{pmatrix} -0.0472 \pm 0.0003 \\ 0.653 \pm 0.006 \\ 0.606 \pm 0.006 \end{pmatrix}$	0.1%

Lamp calibration. Similarly, we collected 10 images of the pencil shadow (like figure 4-top-right) and performed calibration of the light source on them. See section 2.3. Notice that the points $\overline{b}$ and t_s were manually extracted from the images. Define $\overline{X}_S$ as the coordinate vector of the light source in the camera reference frame. The following table summarizes the calibration results obtained for the setup shown in figure 4 (refer to figure 8 for notation):

Parameters	Estimates	Relative errors
$\overline{X}_S$ (cm)	$\begin{pmatrix} -13.7 \pm 0.1 \\ -17.2 \pm 0.3 \\ -2.9 \pm 0.1 \end{pmatrix}$	$\approx 2\%$
h_S (cm)	34.04 ± 0.15	0.5%
ξ (degrees)	146.0 ± 0.8	0.2%
ϕ (degrees)	64.6 ± 0.2	0.06%

The estimated lamp height agrees with the manual measure (with a ruler) of 34 ± 0.5 cm.

This accuracy is sufficient for not inducing any significant global distortion onto the final recovered shape, as we discuss in the next section.

3.2 Scene reconstructions

Experiment 1 - Indoor scene: We took two scans of the same scene with the desk lamp first on the right side and then on the left side of the camera. The two resulting meshes are shown on the top row on figure 9. The meshes were then merged together following the technique described in section 2.8. The bottom figure shows the resulting mesh composed of 66,579 triangles. We estimated the surface error (σ_{Z_c}) to approximately .7 mm in standard deviation over 50 cm large objects, leading to a relative reconstruction error of 0.15%. The white holes in the mesh images correspond to either occluded regions (not observed from the camera, or not illuminated) or very low albedo areas (such as the black squares on the horizontal plane). There was no significant global deformation in the final structured surface: after fitting a quadratic model through sets of points on the two planes, we only noticed a decrease of approximately 5% in standard deviation of the surface error. One may therefore conclude that the calibration procedure returns sufficiently accurate estimates. The original input sequences were respectively 665 and 501 frames long, each image being 320×240 pixels large, captured with a grayscale camera.

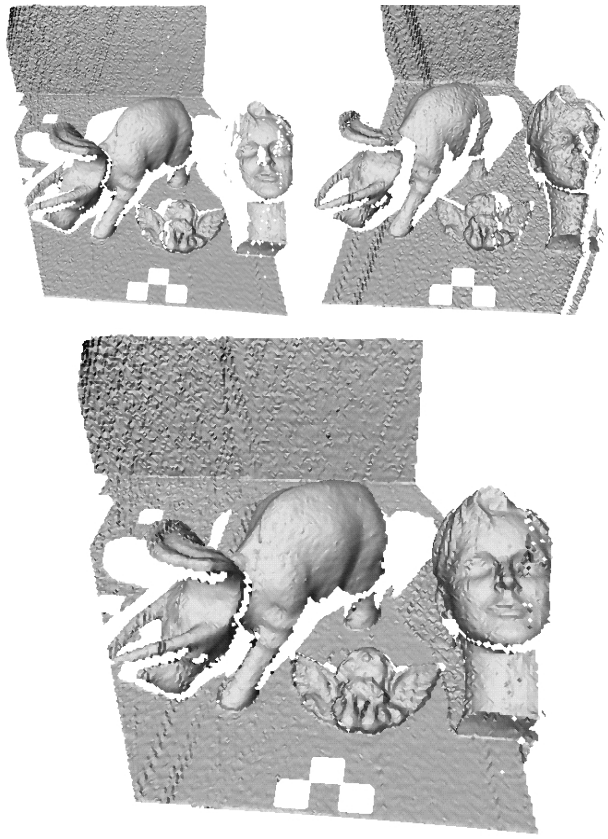
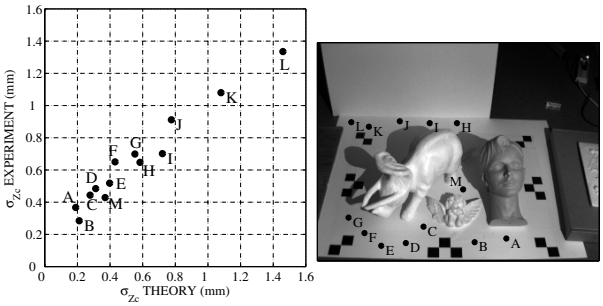


Figure 9: **Experiment 1 - Indoor scene**

Figure 10 reports a comparison test between the theoretical depth variances obtained from expression (9) and that computed from the reconstructed surface. This test was done on the first scan of the scene shown on figure 9-top-left. In that test, we experimentally compute the standard deviation σ_{Z_c} of the error on the depth estimate Z_c at 13 points $p = (A, B, \ldots, M)$



p	$\overline{\nabla} I$	$[\omega_x \ \omega_y]^T$	Z_c	σ_{Z_c} th.	σ_{Z_c} exp.
A	71.5 18.0	1.6591 0.2669	1332.4	0.19	0.37
B	69.0 12.0	1.7755 0.3762	1317.2	0.21	0.28
C	61.0 11.0	1.9639 0.3576	1355.6	0.28	0.44
D	52.0 12.0	2.0788 0.3071	1300.0	0.31	0.48
E	40.5 14.0	2.2454 0.2170	1286.2	0.40	0.52
F	42.0 12.0	2.3455 0.1606	1318.6	0.43	0.65
G	37.5 10.0	2.5048 0.1101	1363.4	0.55	0.70
H	46.5 9.0	1.7752 0.3776	1800.8	0.58	0.65
I	38.5 9.5	1.8700 0.3608	1789.6	0.72	0.70
J	38.0 9.5	2.0038 0.3491	1786.1	0.78	0.91
K	28.0 7.5	2.1815 0.2523	1749.7	1.08	1.08
L	21.5 7.0	2.2834 0.1953	1769.0	1.46	1.34
M	51.0 10.0	1.7905 0.3765	1495.2	0.37	0.43

Figure 10: **Comparison of measured and predicted reconstruction error σ_{Z_c}:** The standard deviation σ_{Z_c} of the depth estimate error are experimentally calculated at 13 points $p = (A, B, \ldots, M)$ picked randomly on the horizontal plane Π_h and computed theoretically using equation 9. The experimental estimates are reported in the last column of the table (in mm) and the second last column reports the corresponding theoretical estimates. The terms involved in equation 9 are also given: $\overline{\nabla} I$ (in units of brightness per pixel), $[\omega_x \ \omega_y]^T$ (in m^{-1}) and Z_c (in mm). The image noise was experimentally estimated to $\sigma_I = 2$ brightness values, and the focal value used was $f_c = 426$ pixels. The top-left figure shows a plot is the theoretical standard deviations versus the experimental ones. Observe that the theoretical error model captures quite faithfully the actual variations in accuracy of reconstruction within the entire scene: as the point of interest moves from the left to the right part of the scenery, accuracy increases due to sharper edges, and a smaller shadow plane vector $\overline{\omega}_c$; in addition, deeper areas in the scene are more noisy mainly because of larger absolute depths Z_c and shallower shadow edges (smaller $\|\overline{\nabla} I\|$). We conclude from that experiment that equation 9 returns an accurate estimate for σ_{Z_c}.

picked randomly on the horizontal plane Π_h of the scan data shown on figure 9-top-left. Figure 10-top-right shows the positions of those points in the scene. The standard deviation σ_{Z_c} at a given point p in the image is experimentally calculated by first taking the 9×9 pixel neighborhood around p resulting into a set of 81 points in space that should lie on Π_h. We then fit a plane across those 81 points (in the least squares sense) and set σ_{Z_c} as the standard deviation of the residual algebraic distances of the entire set of points to this best fit plane. The experimental estimates for σ_{Z_c} are reported in the last column of the table (in mm). The second last column reports the corresponding theoretical estimates of σ_{Z_c} (in mm) computed using equation 9. The terms involved in that equation are also given: $\overline{\nabla} I$ (in units of brightness per pixel), $[\omega_x \ \omega_y]^T$ (in m^{-1}) and Z_c (in mm). The image noise was experimentally estimated to $\sigma_I = 2$ brightness values (calculation based on 100 acquired images of the same scene), and the focal value used was $f_c = 426$ pixels. See sec. 2.7 for a complete description of those quantities. The top-left figure shows a plot of the theoretical standard deviations versus the experimental ones. Observe that the theoretical error model captures quite faithfully the actual variations in accuracy of reconstruction within the entire scene: as the point of interest moves from the left to the right part of the scenery, accuracy increases due to sharper edges, and a smaller shadow plane vector $\overline{\omega}_c$; in addition, deeper areas in the scene are more noisy mainly because of larger absolute depths Z_c and shallower shadow edges (smaller $\|\overline{\nabla} I\|$). We conclude from that experiment that equation 9 returns a valid estimate for σ_{Z_c}.

Experiment 2 - Scanning of a textured skull: We took one scan of a small painted skull, using a single reference plane Π_h, with known light source position (pre-calibrated). Two images of the sequence are shown on the top row of figure 11. The recovered shape is presented on the second row (33,533 triangles), and the last row shows three views of the mesh textured by the top left image. Notice that the textured regions of the object are nicely reconstructed (although these regions have smaller contrast I_{contrast}). Small artifacts observable at some places on the top of the skull are due to the saturation of the pixel values to zero during shadow passage. This effect induces a positive bias on the threshold I_{shadow} (since $I_{\min}$ is not as small as it should be). Consequently, those pixels take on slightly too small shadow times t_s and are triangulated with shadow planes that are shifted to the left. In effect, their final 3D location is slightly off the surface of the object. One possible solution to that problem consists of taking multiple scans of the object with different camera apertures, and retain each time the range results for the pixels that do not suffer from saturation. The overall

reconstruction error was estimated to approximately 0.1 mm over a 10 cm large object leading to a relative error of approximately 0.1%. In order to check for global distortion, we measured the distances between three characteristic points on the object: the tip of the two horns, and the top medium corner of the mouth. The values obtained from physical measurements on the object and the ones from the retrieved model agreed within the error of measurement (on the order of 0.5mm over distances of approximately 12 to 13cm). The sequence of images was 670 frames long, each image being 320×240 pixels large (acquired with a grayscale camera).

Experiment 3 - Textured and colored fruits: Figure 12 shows the reconstruction results on two textured and colored fruits. The second row shows the reconstructed shapes. The two meshes with the pixel images textured on them are shown on the third row. Similar reconstruction errors to the previous experiment (Experiment 2) were estimated on that data set. Notice that both textured and colored regions of the objects were well reconstructed: the local surface errors was estimated between 0.1 mm and 0.2 mm, leading to relative errors of approximately 0.1%.

Experiment 4 - Outdoor scene: In this experiment, the sun was the light source. See figure 13. The final mesh is shown on the bottom figure (106,982 triangles). The reconstruction error was estimated to 1mm in standard deviation, leading to a relative error of approximately 0.2%. The larger reconstruction error is possibly due to the fact that the sun is not well approximated by a point light source (as discussed in Appendix C). Once again, there was no noticeable global deformation induced by calibration. After fitting a quadratic model to sets of points on the planes, we only witnessed a decrease of approximately 5% on the standard deviation of the residual error. The original sequence was 790 images long acquired with a consumer electronics color camcorder (at 30 Hz). After digitization, and de-interlacing, each image was 640×240 pixel large. The different digitalization technique may also explain the larger reconstruction error.

Experiment 5 - Outdoor scanning of a car: Figure 14 shows the reconstruction results on scanning a car with the sun. The two planes (ground floor and back wall) approach was used to infer the shadow plane (without requiring the sun position). The initial sequence was 636 frames long acquired with a consumer electronics color video-camera (approximately 20 seconds long). Similarly to Experiment 4, the sequence was digitized resulting to 640×240 pixel large non-interlaced images. Two images of the sequence are presented on the top row, as well as two views of the reconstructed 3D mesh after scanning. The reconstruction errors were estimated to approximately 1 cm, or 0.5% of the size of the car (approximately 3 meters).

4 Conclusion and future work

We have presented a simple, low cost system for 3D scanning. The system requires very little equipment (a light source, and a straight edge to cast the shadow) and is very simple and intuitive to use and to calibrate. This technique scales well to large objects and may be used in brightly lit scenes where most active lighting methods are impractical (expect synchronized scanning systems [33]). In outdoor scenarios, the sun is used as light source and is allowed to move during a scan. The method requires very little processing and image storage and has been implemented in real time (30 Hz) on a Pentium 300MHz machine. The accuracies that we obtained on the final reconstructions are reasonable (error at most 0.5% of the size of the scene). In addition, the final outcome is a dense and conveniently organized coverage of the surface (one point in space for each pixel in the image), allowing direct triangular meshing and texture mapping. We also showed that using dual-space geometry enables us to keep the mathematical formalism simple and compact throughout the successive steps of the method. An error analysis was presented together with a description of a simple technique for merging multiple 3D scans in order to obtain a better coverage of the scene, and reduce the estimation error. The overall calibration procedure, even in the case of multiple scans, is intuitive, simple, and accurate.

Our method may be used to construct complete 3D object models. One may take multiple scans of the object at different locations in space, and then align the sets of range images. For that purpose, a number of algorithms have been explored and shown to yield excellent results [3, 21, 40]. The final step consists of constructing the final object surface from the aligned views [1, 17, 40].

It is part of future work to incorporate a geometrical model of extended light source to the shadow edge detection process, in addition to developing an uncalibrated (projective) version of the method. One step towards an uncalibrated system may be found in [9]. In this paper, we study the case of 3D reconstruction from a set of planar shadows when there is no calibrated background plane in the scene.

A Dual-space formalism

Let $(E) = \mathbb{R}^3$ be the 3D Euclidean space. A plane Π in (E) is uniquely represented by the 3-vector $\overline{\omega} = [\omega_x \quad \omega_y \quad \omega_z]^T$ such that any point P of coordinate vector $\overline{X}_c = [X_c \quad Y_c \quad Z_c]^T$ (expressed in the camera reference frame) lies on Π if and only if $\langle \overline{\omega}, \overline{X}_c \rangle = 1$ ($\langle ., . \rangle$ is the standard scalar product operator). Notice that $\overline{\omega} \doteq \overline{n}/d$ where $\overline{n}$ is the unitary normal vector of the plane and $d \neq 0$ the plane's distance to the origin. Let $(\Omega) = \mathbb{R}^3$. Since every point $\overline{\omega} \in (\Omega)$

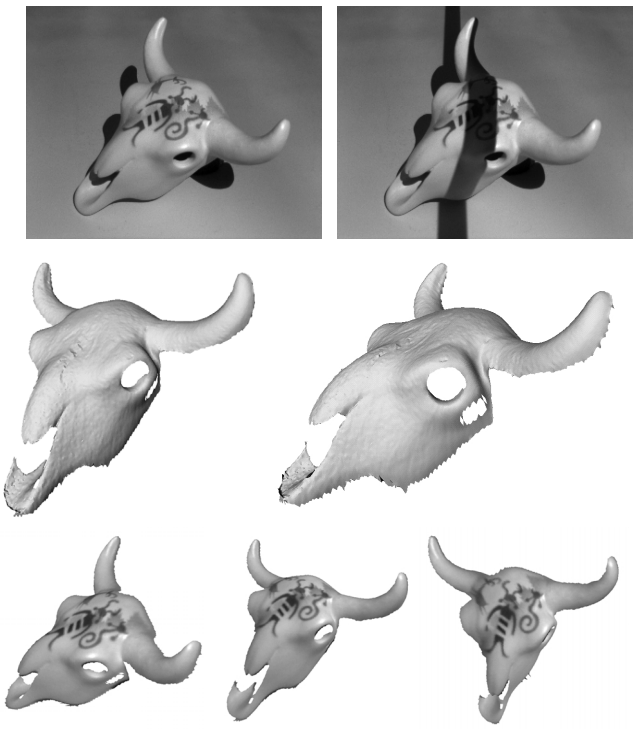


Figure 11: **Experiment 2 - Scanning of a textured skull**

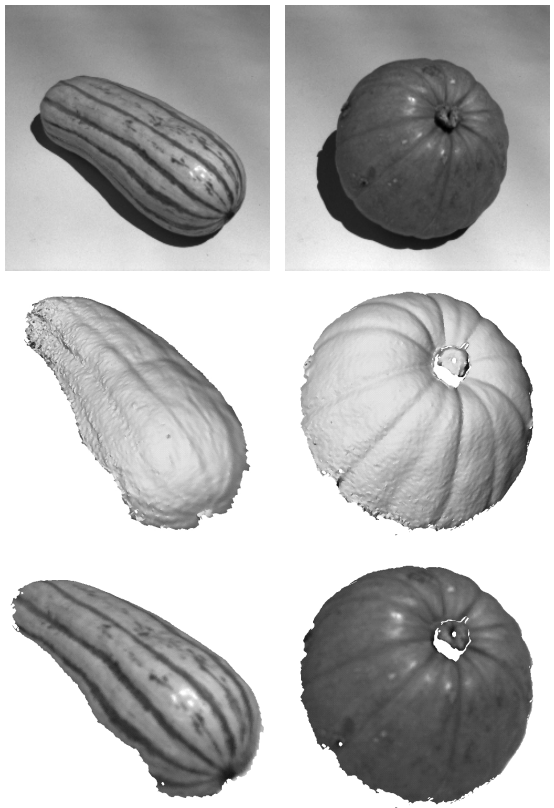


Figure 12: **Experiment 3 - Textured and colored fruits**

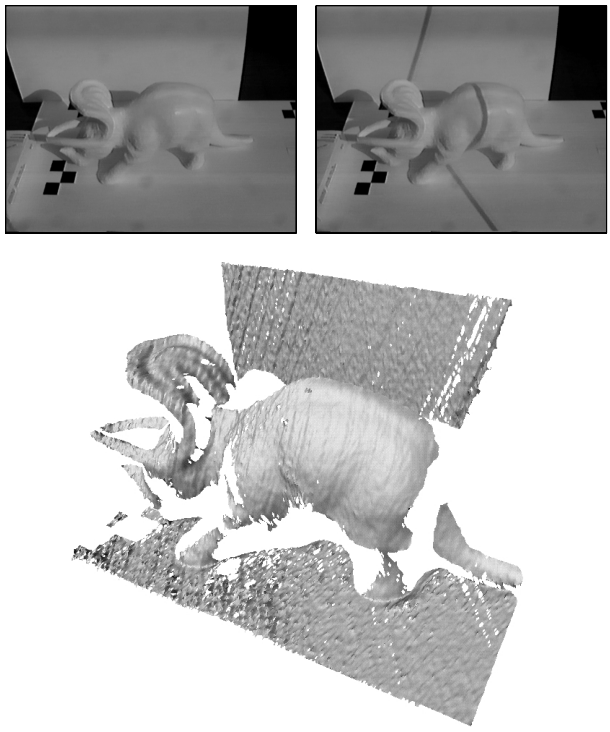


Figure 13: **Experiment 4 - Outdoor scanning of an object**

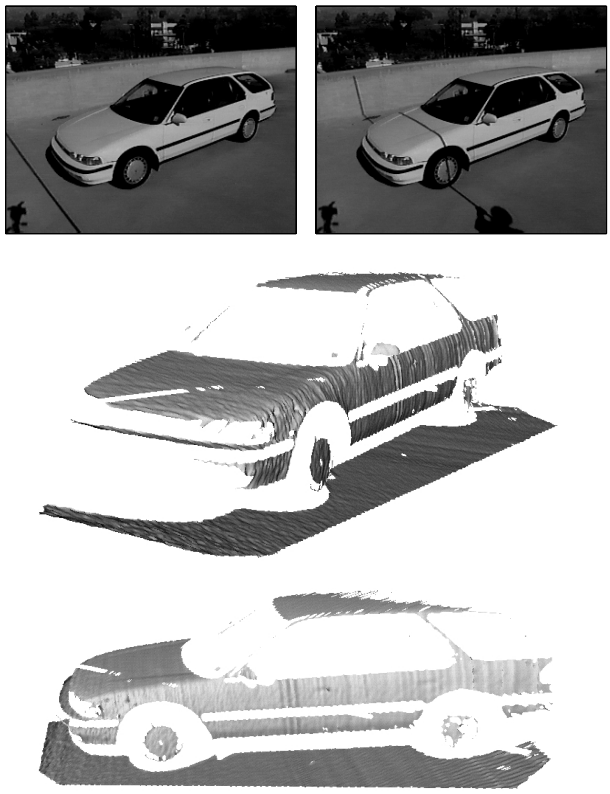


Figure 14: **Experiment 5 - Outdoor scanning of a car**

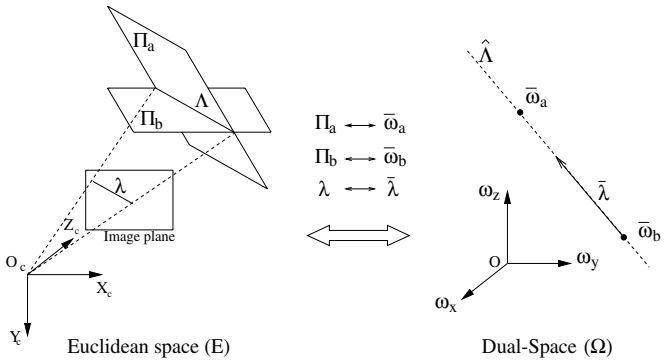


Figure 15: **Proposition 1:** The direction of the line connecting two planes vectors $\overline{\omega}_a$ and $\overline{\omega}_b$ in dual-space (Ω) is precisely $\overline{\lambda}$, the coordinate vector of the perspective projection λ of the line of intersection Λ between the two planes Π_a and Π_b in Euclidean space (E).

corresponds to a unique plane Π in (E), we refer to (Ω) as the 'dual-space'. Conversely, every plane Π that does not contain the origin has a valid coordinate vector $\overline{\omega}$ in (Ω). Notice that the set of plane crossing the origin cannot be parameterized in (Ω) space, since the $\overline{\omega}$ diverges to infinity as d gets closer to zero.

Similarly, a line λ on the image plane is represented by the 3-vector $\overline{\lambda}$ (up to scale) such that any point p of coordinates $\overline{x}_c = [x_c \;\; y_c \;\; 1]^T$ lies on this line if and only if $\langle \overline{\lambda}, \overline{x}_c \rangle = 0$. See [20, 24, 35].

Originally, the dual-space of a given vector space (E) is defined as the set of linear forms on (E) (linear functions of (E) into the reals $\mathbb{R}$). See [4]. In the case where (E) is the three dimensional Euclidean space, each linear form may be interpreted as a plane Π in space that is typically parameterized by a homogeneous 4-vector $\overline{\pi} = [\pi_1 \;\; \pi_2 \;\; \pi_3 \;\; \pi_4]^T$. A point P of homogeneous coordinates $\overline{X} = [X \;\; Y \;\; Z \;\; 1]^T$ lies on a generic plane Π of coordinates $\overline{\pi}$ if and only if $\langle \overline{\pi}, \overline{X} \rangle = 0$ (see [12]). Our $\overline{\omega}-$parameterization differs from the conventional parameterization in that it does not allow to represent planes crossing the origin (the correspondence between the two parameterizations is $\overline{\omega} = -[\pi_1 \;\; \pi_2 \;\; \pi_3]^T/\pi_4$, therefore $\pi_4 \neq 0$). However, that does not constitute a limitation in our application since none of the planes we need to parameterize are allowed to cross the origin (as discussed in sections 2.2 and 2.6). Furthermore, this new representation exhibits useful properties allowing to naturally relate objects in 3D (planes, lines and points) to their perspective projections on the image plane (lines and points) in addition to providing very compact analytical results in error sensitivity analysis.

The following proposition constitutes the major property associated to our choice of parameterization:

Proposition 1: Consider two planes Π_a and Π_b in space, with respective coordinate vectors $\overline{\omega}_a$ and $\overline{\omega}_b$ ($\overline{\omega}_a \neq \overline{\omega}_b$), and let $\Lambda = \Pi_a \cap \Pi_b$ be the line of intersection between them. Let λ be the perspective projection of Λ on the image plane, and $\overline{\lambda}$ its representative vector. Then $\overline{\lambda}$ is parallel to $\overline{\omega}_a - \overline{\omega}_b$ (see figure 15). In other words, $\overline{\omega}_a - \overline{\omega}_b$ is a valid coordinate vector of the line λ.

Proof: Let $P \in \Lambda$ and let p be the projection of P on the image plane. Call $\overline{X} = [X \;\; Y \;\; Z]^T$ and $\overline{x} = \frac{1}{Z}\overline{X}$ the respective coordinates of P and p. We successively have:

$$
\begin{aligned}
P \in \Lambda \quad &\Longleftrightarrow \quad \begin{cases} P \in \Pi_a \\ P \in \Pi_b \end{cases} \\
&\Longleftrightarrow \quad \begin{cases} \langle \overline{\omega}_a, \overline{X} \rangle = 1 \\ \langle \overline{\omega}_b, \overline{X} \rangle = 1 \end{cases} \\
&\Longrightarrow \quad \langle \overline{\omega}_a - \overline{\omega}_b, \overline{x} \rangle = 0.
\end{aligned}
$$

Therefore $(\overline{\omega}_a - \overline{\omega}_b)$ *is* a representative vector of λ and must be parallel to $\overline{\lambda}$. ■

Consequently, the coordinate vector $\overline{\omega}$ of any plane Π containing the line Λ will lie on the line connecting $\overline{\omega}_a$ and $\overline{\omega}_b$ in dual-space (Ω). We denote that line by $\hat{\Lambda}$ and call it the *dual image* of Λ. The following definition generalizes that concept of dual image:

Definition: Let $\mathcal{A}$ be a submanifold of (E) (e.g. a point, line, plane, surface or curve). The *dual image* $\hat{\mathcal{A}}$ of $\mathcal{A}$ is defined as the set coordinates vectors $\overline{\omega}$ in dual-space (Ω) representing the tangent planes to $\mathcal{A}$. Following that standard definition (see [12]), the dual images of points, lines and planes in (E) may be shown to be respectively planes, lines and points in dual-space (Ω), as illustrated in figure 16. Further properties regarding non-linear sub-manifolds may be observed, such as for quadric surfaces in [15].

B Proof of $h_S/d_h = 1 - \langle \overline{\omega}_h, \overline{X}_S \rangle$

Since $\overline{\omega}_h$ is the coordinate vector of the plane Π_h, the vector $\overline{n}_h = d_h\,\overline{\omega}_h$ is the normal vector of the plane Π_h in the camera reference frame (see figure 8). Let P be a point in Euclidean space (E) of coordinate vector $\overline{X}$. The quantity $d_h - \langle \overline{n}_h, \overline{X} \rangle$ is then the (algebraic) orthogonal distance of P to Π_h (positive quantity if the P is on the side of the camera, negative otherwise). In particular, if P lies on Π_h, then $\langle \overline{n}_h, \overline{X} \rangle = d_h$, which is equivalent to $\langle \overline{\omega}_h, \overline{X} \rangle = 1$. The orthogonal distance of the light source S to Π_h is denoted h_S on figure 8. Therefore $h_S = d_h - \langle \overline{n}_h, \overline{X} \rangle$, or equivalently $1 - \langle \overline{\omega}_h, \overline{X}_S \rangle = h_S/d_h$. ■

C Sensitivity Analysis

This appendix presents a complete error analysis for the whole reconstruction scheme. As first mentioned in section 2, the method proposes to associate to every pixel $\overline{x}_c$ the time instant $t_s(\overline{x}_c)$ at which the shadow crosses that particular pixel. That given time corresponds to the shadow plane $\Pi(t_s(\overline{x}_c))$ in space (of coordinate vector $\overline{\omega}_c$), used at the triangulation step

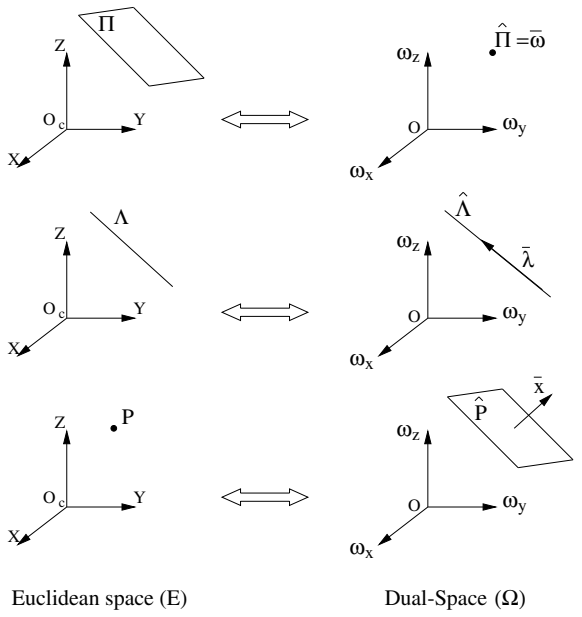


Figure 16: **Duality principle:** The dual images of a plane Π, a line Λ and a point P. Notice that the perspective projection $\overline{\lambda}$ of the line Λ is directly observable in dual-space as the direction vector of its dual image $\hat{\Lambda}$. Similarly, the coordinate vector $\overline{x}$ of the projection of P is precisely the normal vector the plane $\hat{P}$ (dual image of P).

to retrieve the coordinates of the point P in space (see figure 2). In addition, at every time instant t, a shadow plane $\Pi(t)$ is estimated based on two line segments $\lambda_h(t)$ and $\lambda_v(t)$ extracted from the image plane (see section 2.4).

Therefore, one clearly identifies two possible sources of error affecting the overall reconstruction: errors in localizing the two edges $\lambda_h(t)$ and $\lambda_v(t)$ leading to error in estimating the shadow plane $\Pi(t)$ (or error on the vector $\overline{\omega}(t)$), and errors in finding the shadow time $t_s(\overline{x}_c)$ (at every pixel $\overline{x}_c$) leading to an error in shadow plane assignment.

Experimentally, we found that the error coming from spatial processing (shadow plane localization) was much smaller than the one coming from temporal processing (shadow time computation). In other words, in all the experiments we carried out, the shadow planes were localized to such a degree of accuracy that the errors induced by the noise on $\overline{\omega}_c$ were negligible compared to the errors induced by the noise on $t_s(\overline{x}_c)$. This experimental observation is reasonable because the shadow edges $\lambda_h(t)$ and $\lambda_v(t)$ are recovered by fitting lines through many points on the image plane (an order of 50 points per line) while shadow time $t_s(\overline{x}_c)$ is estimated on a basis of a single pixel. Notice that this is experiment dependent, and may very well not be true if fewer points were used to extract the shadow edges, or if the image were more noisy, or more distorted. In those cases, both error terms should be retained. In the present analysis, we propose to derive an expression of the variance of the error in depth estimation $\sigma^2_{Z_c}$ assuming that the main source of noise comes from temporal processing. In the experimental section, we verify that the final variance expression agrees numerically with accuracies achieved on real scan data.

C.1 Derivation of the depth variance $\sigma^2_{Z_c}$

Every pixel $\overline{x}_c$ on the image sees the shadow passing at time a $t_s(\overline{x}_c)$, called the shadow time, that is estimated through temporal processing (see section 2.4). This estimation is naturally subject to errors, leading to inaccuracies in the final 3D reconstruction. The purpose of that analysis is to study how damaging those errors truly are on the final structure, and quantify them. Assume that for a given pixel $\overline{x}_c$, an additive temporal error $\delta t_s(\overline{x}_c)$ is made on its shadow time estimate: $\tilde{t}_s(\overline{x}_c) = t_s(\overline{x}_c) + \delta t_s(\overline{x}_c)$. This typically leads the algorithm to assign to the pixel $\overline{x}_c$ the "wrong" shadow plane $\Pi(t_s(\overline{x}_c) + \delta t_s(\overline{x}_c))$ for the geometrical triangulation step. Equivalently, one can think that the plane $\Pi(t_s(\overline{x}_c) + \delta t_s)$ has been associated with the "wrong" pixel $\overline{x}_c$ in the image. Although it does not change anything to the problem, that way of centering the reasoning onto the shadow plane instead of the pixel actually significantly simplifies the whole analysis. Indeed, as we will show in the following, if we assign the noise to the pixel location itself, the time variable can then be omitted.

To be more precise, let us first define $\overline{v}(\overline{x}_c) = [v_x(\overline{x}_c) \quad v_y(\overline{x}_c)]^T$ to be the velocity vector of the shadow at the pixel $\overline{x}_c$ that is orthogonal to the shadow edge. Then, the closest point to $\overline{x}_c$ that has truly been lit by the shadow plane $\Pi(t_s(\overline{x}_c) + \delta t_s(\overline{x}_c))$ is $\overline{x}_c + \delta t_s(\overline{x}_c)\,\overline{v}(\overline{x}_c)$. Therefore, by picking $\overline{x}_c$ instead, we introduce an additive pixel error $\delta\overline{x}_c \doteq -\delta t_s(\overline{x}_c)\,\overline{v}(\overline{x}_c)$. This is the equivalent noise that can be attributed to the pixel location $\overline{x}_c$ before triangulation.

One can then see that this equivalent image coordinate noise is naturally related to the speed of the shadow. Indeed, even if we assume that the time estimation error δt_s is identical for every pixel in the image, the corresponding pixel error $\delta\overline{x}_c$ is generally not uniform, neither in direction, nor in magnitude. Typically, fast moving shadow regions will be subject to larger errors than slow moving shadow regions. Variations in apparent shadow speed can be caused by a change in the actual speed at which the stick is moved, a change in local surface orientation of the scene, or both.

Before triangulation, the pixel coordinates have to be normalized by the intrinsic parameters of the camera. Let us assume, for simplicity in the notation, that $\overline{x}_c = [x_c \quad y_c \quad 1]^T$ is directly the normalized, homogeneous coordinate vector associated to the pixel. The two coordinates x_c and y_c are affected by the

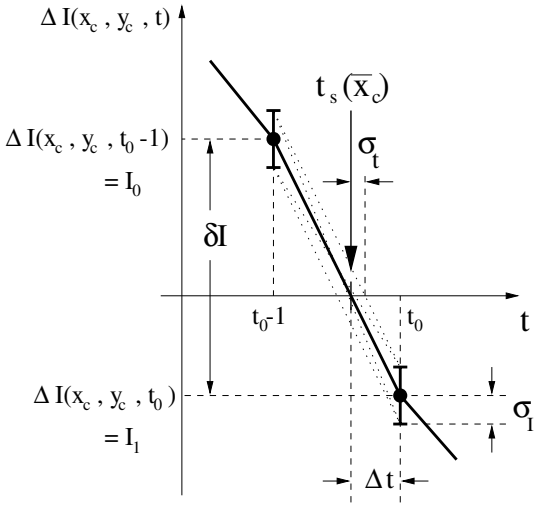


Figure 17: **Estimation error on the shadow time:** The shadow time $t_s(\overline{x}_c)$ is estimated by linearly interpolating the difference temporal brightness function $\Delta I(x_c, y_c, t)$ between times $t_0 - 1$ and t_0. The pixel noise (of standard deviation σ_I) on $I_0 \doteq \Delta I(x_c, y_c, t_0 - 1)$ and $I_1 \doteq \Delta I(x_c, y_c, t_0)$ induces errors on the estimation of Δt, or equivalently $t_s(\overline{x}_c)$. This error has variance σ_t^2.

error vector $\delta\overline{x}_c$ whose variance-covariance matrix is denoted $\Sigma_{\overline{x}_c}$ (a 2×2 matrix). Let us derive an expression for that matrix as a function of the image brightness noise.

Lemma: Let σ_I be the standard deviation of the image brightness noise (estimated experimentally). We can write $\Sigma_{\overline{x}_c}$ as a function of the image gradient $\overline{\nabla} I(\overline{x}_c)$ at pixel $\overline{x}_c$ at time $t = t_s(\overline{x}_c)$:

$$\Sigma_{\overline{x}_c} = \frac{\sigma_I^2}{f_c^2 \, \|\overline{\nabla} I(\overline{x}_c)\|^2} \begin{bmatrix} \cos^2\varphi & \cos\varphi \sin\varphi \\ \cos\varphi \sin\varphi & \sin^2\varphi \end{bmatrix} \tag{14}$$

where f_c is the focal length of the camera (in pixels), $\overline{\nabla} I(\overline{x}_c)$ is the gradient vector of the image brightness at the shadow, and φ the orientation angle of that vector (orientation of the shadow edge at pixel $\overline{x}_c$):

$$\overline{\nabla} I(\overline{x}_c) = \begin{bmatrix} I_x(\overline{x}_c) \\ I_y(\overline{x}_c) \end{bmatrix} = \|\overline{\nabla} I(\overline{x}_c)\| \begin{bmatrix} \cos\varphi \\ \sin\varphi \end{bmatrix}$$

where:

$$I_x(\overline{x}_c) \doteq \left. \frac{\partial I(\overline{x}, t)}{\partial x} \right|_{\overline{x}=\overline{x}_c, t=t_s(\overline{x}_c)}$$

$$I_y(\overline{x}_c) \doteq \left. \frac{\partial I(\overline{x}, t)}{\partial y} \right|_{\overline{x}=\overline{x}_c, t=t_s(\overline{x}_c)}$$

Proof of lemma (eq. 14): Figure 17 shows the principle of computing the shadow time $t_s(\overline{x}_c)$ from the difference image ΔI (refer to section 2.5). For clarity in the notation, define $I_0 \doteq \Delta I(x_c, y_c, t_0 - 1)$ and $I_1 \doteq \Delta I(x_c, y_c, t_0)$. Then, the shadow time $t_s(\overline{x}_c)$ is given by:

$$t_s(\overline{x}_c) = t_0 - \Delta t$$

where:

$$\Delta t \doteq \frac{I_1}{I_1 - I_0}$$

Let σ_t^2 be the variance of the error $\delta t_s(\overline{x}_c)$ attached to the shadow time $t_s(\overline{x}_c)$. In normal sampling conditions (if the temporal brightness is sufficiently sampled within the shadow transition area), the same error is on the variable Δt, and therefore σ_t may be directly expressed as a function of σ_I, the variance of pixel noise on I_0 and I_1:

$$\sigma_t^2 = \left(\left(\frac{\partial \Delta t}{\partial I_0} \right)^2 + \left(\frac{\partial \Delta t}{\partial I_1} \right)^2 \right) \sigma_I^2$$

$$\sigma_t^2 = \frac{I_0^2 + I_1^2}{\delta I^4} \, \sigma_I^2 \tag{15}$$

where $\delta I \doteq I_1 - I_0$ is the temporal brightness variation at the zero crossing (or equivalently at the shadow time). One may notice from equation 15 that, as the brightness difference δI increases, the error in shadow time decreases. That is a very intuitive behavior given that higher shadow contrasts should give rise to better accuracies. Notice however that the variance σ_t^2 is not only a function of δI but also of the absolute brightness values I_0 and I_1. One may then consider the maximum value of σ_t^2 for a fixed δI over all I_0 and I_1, subject to the constraint $I_1 = I_0 + \delta I$:

$$\sigma_t^2 = \max_{0 < I_0 < -\delta I} \left\{ \frac{2\, I_0^2 + 2\, I_0\, \delta I + \delta I^2}{\delta I^4} \right\} \sigma_I^2$$

leading to the following simplified expression for σ_t^2:

$$\sigma_t^2 = \frac{\sigma_I^2}{\delta I^2} \tag{16}$$

To motivate that simplification, one may notice that the minimum and maximum values of σ_t^2 over all values I_0 and I_1 are quite similar anyway: $\sigma_I^2/(2\,\delta I^2)$ (minimum) and $\sigma_I^2/\delta I^2$ (maximum). The maximum may be thought as an upper bound on the error. Notice that δI is nothing but the first temporal derivative of the image brightness at the pixel $\overline{x}_c$, at the shadow time:

$$\delta I = \left. \frac{\partial I(\overline{x}, t)}{\partial t} \right|_{\overline{x}=\overline{x}_c, t=t_s(\overline{x}_c)}$$

This temporal derivative may also be expressed as a function of the image gradient vector $\overline{\nabla} I(\overline{x}_c) =$

$[I_x(\overline{x}_c) \quad I_y(\overline{x}_c)]^T$ and the shadow edge velocity vector $\overline{v}(\overline{x}_c) = [v_x(\overline{x}_c) \quad v_y(\overline{x}_c)]^T$:

$$\delta I = -\overline{\nabla} I(\overline{x}_c)^T \, \overline{v}(\overline{x}_c) = -I_x(\overline{x}_c)\, v_x(\overline{x}_c) - I_y(\overline{x}_c)\, v_y(\overline{x}_c)$$

By definition, the edge velocity vector $\overline{v}(\overline{x}_c)$ is orthogonal to the shadow edge. Therefore it may be also written as a direct function of the gradient vector $\overline{\nabla} I(\overline{x}_c)$:

$$\overline{v}(\overline{x}_c) = s \, \|\overline{v}(\overline{x}_c)\| \, \frac{\overline{\nabla} I(\overline{x}_c)}{\|\overline{\nabla} I(\overline{x}_c)\|} = s \, \|\overline{v}(\overline{x}_c)\| \begin{bmatrix} \cos\varphi \\ \sin\varphi \end{bmatrix}$$

where s is either $+1$ or -1 depending on the direction of motion of the edge. Therefore,

$$\delta I = (-s) \, \frac{\overline{\nabla} I(\overline{x}_c)^T \, \overline{\nabla} I(\overline{x}_c)}{\|\overline{\nabla} I(\overline{x}_c)\|} \, \|\overline{v}(\overline{x}_c)\|$$

$$\delta I = (-s) \, \|\overline{\nabla} I(\overline{x}_c)\| \, \|\overline{v}(\overline{x}_c)\| \qquad (17)$$

Consequently, by substituting (17) into (16), we obtain a new expression for the temporal variance σ_t^2:

$$\sigma_t^2 = \frac{\sigma_I^2}{\|\overline{\nabla} I(\overline{x}_c)\|^2 \, \|\overline{v}(\overline{x}_c)\|^2}$$

Then, the error vector $\delta\overline{x}_c$ transfered on the image plane is also related to the shadow edge velocity $\overline{v}(\overline{x}_c)$ and the temporal error $\delta t_s(\overline{x}_c)$:

$$\delta\overline{x}_c = -\delta t_s(\overline{x}_c) \, \overline{v}(\overline{x}_c)$$

$$\delta\overline{x}_c = (-s) \, \|\overline{v}(\overline{x}_c)\| \, \delta t_s(\overline{x}_c) \begin{bmatrix} \cos\varphi \\ \sin\varphi \end{bmatrix}$$

Then, the variance-covariance matrix of the noise $\delta\overline{x}_c$ is (recall that $s^2 = 1$):

$$\Sigma_{\overline{x}_c} = \|\overline{v}(\overline{x}_c)\|^2 \, \sigma_t^2 \begin{bmatrix} \cos^2\varphi & \cos\varphi\sin\varphi \\ \cos\varphi\sin\varphi & \sin^2\varphi \end{bmatrix}$$

$$\Sigma_{\overline{x}_c} = \frac{\sigma_I^2}{\|\overline{\nabla} I(\overline{x}_c)\|^2} \begin{bmatrix} \cos^2\varphi & \cos\varphi\sin\varphi \\ \cos\varphi\sin\varphi & \sin^2\varphi \end{bmatrix}$$

Finally, note that this relation is valid if x_c is expressed in pixel coordinates. After normalization, this variance must be scaled by the square of the inverse of focal length f_c:

$$\Sigma_{\overline{x}_c} = \frac{\sigma_I^2}{f_c^2 \, \|\overline{\nabla} I(\overline{x}_c)\|^2} \begin{bmatrix} \cos^2\varphi & \cos\varphi\sin\varphi \\ \cos\varphi\sin\varphi & \sin^2\varphi \end{bmatrix}$$

which ends the proof of the lemma (eq. 14). ■

Notice that if the shadow edge is roughly vertical on the image, one may assume $\varphi = 0$, and therefore simplify quite significantly the variance expression:

$$\Sigma_{\overline{x}_c} = \frac{\sigma_I^2}{f_c^2 \, I_x^2(\overline{x}_c)} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

In that case, we reach the very intuitive result that only the first coordinate of $\overline{x}_c$ is affected by noise.

Since $\Sigma_{\overline{x}_c}$ in inversely proportional to the image gradient, accuracy improves with shadow edge sharpness. In addition, observe that $\Sigma_{\overline{x}_c}$ does not directly depend upon the local shadow speed. Therefore, decreasing the scanning speed would not increase accuracy. However, for the analysis leading to equation 14 to remain valid, the temporal pixel profile must be sufficiently sampled within the transition area of the shadow edge (the penumbra). Therefore, if the shadow edge were sharper, the scanning should also be slower so that the temporal profile at every pixel would be properly sampled. Further discussions may be found in section 2.7. Another consequence of equation 14 is that one may experimentally compute the variance $\Sigma_{\overline{x}_c}$ of the transfered error directly from the original input sequence: $\overline{\nabla} I(\overline{x}_c)$ is the image gradient at the shadow edge and σ_I is the pixel noise on the image. In addition, assuming that the sharpness of the shadow is approximately uniform over the entire image, then $\Sigma_{\overline{x}_c}$ may also be assumed to be uniform to a first approximation. That constitutes an additional simplification that does not have to be retained in practice.

The final expression of the variance $\sigma_{Z_c}^2$ of the error attached to the depth estimate Z_c may be written as follows:

$$\sigma_{Z_c}^2 = \left(\frac{\partial Z_c}{\partial \overline{x}_c}\right) \Sigma_{\overline{x}_c} \left(\frac{\partial Z_c}{\partial \overline{x}_c}\right)^T$$

One may derive the expression for the Jacobian matrix $\left(\frac{\partial Z_c}{\partial \overline{x}_c}\right)$ from the triangulation equation 8:

$$Z_c = \frac{1}{\langle \overline{\omega}_c, \overline{x}_c \rangle} \implies \frac{\partial Z_c}{\partial \overline{x}_c} = Z_c^2 \begin{bmatrix} \omega_x & \omega_y \end{bmatrix}$$

where ω_x and ω_y are the two first coordinates of the shadow plane vector $\overline{\omega}_c$. This allows to expand the expression of $\sigma_{Z_c}^2$:

$$\sigma_{Z_c}^2 = Z_c^4 \left(\frac{\omega_x \cos\varphi + \omega_y \sin\varphi}{f_c \, \|\overline{\nabla} I(\overline{x}_c)\|}\right)^2 \sigma_I^2 \qquad (18)$$

This expression is directly computable from the original input sequence, and used for scan merging (refer to section 2.8). Several observations regarding that expression may be found in section 2.7.

C.2 System Design Issues

Let us consider the scanning setup as it is presented on figure 8 where the scan is done roughly vertically. In that case, $\varphi \approx 0$, and $I_y^2(\overline{x}_c) \ll I_x^2(\overline{x}_c)$ (see figure 10). Then, the depth variance expression (18) may be further simplified to:

$$\sigma_{Z_c}^2 \approx \frac{Z_c^4\,\omega_x^2}{f_c^2\,I_x^2(\overline{x}_c)}\,\sigma_I^2 \tag{19}$$

It appears then that the first coordinate ω_x of the shadow plane vector $\overline{\omega}_c$ carries most of the variations in accuracy of reconstruction within a given scan. When designing the scanning system, an important issue is to choose the spatial configurations of the camera and the light source that maximize the overall quality of reconstruction, or equivalently minimize $|\omega_x|$. In order to address this issue, it is necessary to further expand the term ω_x, and study its dependence upon the geometrical variables characterizing the system. Since the light source position is of interest here, let us consider the case where a single plane Π_h is used for scanning. In that case, the shadow plane vector $\overline{\omega}_c$ appears as a function of the light source position vector $\overline{X}_S$, as stated by equation 6. Assume that $\overline{\lambda}_h = [\lambda_x \;\; \lambda_y \;\; \lambda_z]^T$ is normalized such that $\lambda_x = 1$. In addition, assume that the (O_c, X_c) axis of the camera is approximately parallel to the plane Π_h (as suggested in figure 8). This implies that the first coordinate of $\overline{\omega}_h$ is zero. Then, the first coordinate ω_x of $\overline{\omega}_c$ reduces to:

$$\omega_x = \frac{1 - \langle \overline{\omega}_h, \overline{X}_S \rangle}{\langle \overline{\lambda}_h, \overline{X}_S \rangle} = \frac{h_S/d_h}{\langle \overline{\lambda}_h, \overline{X}_S \rangle} \tag{20}$$

where d_h and h_S are the respective orthogonal distances of the camera center O_c and the light source S to the plane Π_h.

For simplification purposes, let us assume that the shadow edge λ_h appears vertically on the image plane, and let x be its horizontal position (on the image). As the shadow moves from left to right, x varies from negative values to positive values, crossing zero when the shadow is at the center of the image. In that specific scenario, the shadow edge vector reduces to: $\overline{\lambda}_h = [1 \;\; 0 \;\; -x]^T$ simplifying equation 20:

$$\frac{1}{\omega_x} = \frac{d_h}{h_S}(X_S - x\,Z_S) \tag{21}$$

The problem of maximizing the reconstruction quality corresponds then to maximizing $|1/\omega_x|$. Since that quantity is function of the shadow edge location x, we may observe that the accuracy of reconstruction is not uniform throughout the scene for a given scan (unless the depth of the light source in the camera reference frame is zero: $Z_S = 0$). A better understanding of that relation may be achieved by expressing the light source coordinate vector $\overline{X}_S$ as a function of the angular coordinates θ, ϕ, and ξ defining the mutual positions of the camera and the light source with respect to the plane Π_h (see figure 8):

$$\overline{X}_S = \begin{bmatrix} X_S \\ Y_S \\ Z_S \end{bmatrix} = \begin{bmatrix} h_S \frac{\cos\xi}{\tan\phi} \\ -h_S \frac{\sin\theta\sin\xi}{\tan\phi} + (d_d - h_S)\cos\theta \\ h_S \frac{\cos\theta\sin\xi}{\tan\phi} + (d_d - h_S)\sin\theta \end{bmatrix}$$

Following this notation, the inverse of ω_x may be written as follows:

$$\frac{1}{\omega_x} = d_h \left(\frac{\cos\xi}{\tan\phi} - x \left(\frac{\cos\theta\sin\xi}{\tan\phi} + \frac{d_h - h_S}{h_S}\sin\theta \right) \right)$$

Since during scanning, the shadow edge coordinate x spans a range of values going from negative to positive values, we may consider that taking $x = 0$ gives us an indication of the "average" reconstruction quality:

$$\left.\frac{1}{\omega_x}\right|_{\text{average}} \approx \left.\frac{1}{\omega_x}\right|_{x=0} = d_h\,\frac{\cos\xi}{\tan\phi}$$

Equation 19 may then be used to infer an expression for the "average" depth variance:

$$\left.\sigma_{Z_c}^2\right|_{\text{average}} \approx \frac{Z_c^4}{d_h^2}\,\frac{\tan^2\phi}{\cos^2\xi}\,\frac{\sigma_I^2}{f_c^2\,I_x^2(\overline{x}_c)}$$

A next simplification step may be applied, by observing that the average depth of the scene is approximately related to the height d_h and the tilt angle θ of the camera through the following expression:

$$\left.Z_c\right|_{\text{average}} \approx \frac{d_h}{\sin\theta}$$

That relation leads us to a new expression for the "average" σ_{Z_c}:

$$\left.\sigma_{Z_c}\right|_{\text{average}} \approx d_h\,\frac{\tan\phi}{\sin^2\theta\;|\cos\xi|}\,\frac{\sigma_I}{f_c\;|I_x(\overline{x}_c)|} \tag{22}$$

Notice that this quantity may be computed prior to scanning knowing the geometrical configuration of the system. From that expression, it is also possible to identify optimal configurations of the camera and the light source that maximize the overall quality of the reconstruction. See section 2.7.

Acknowledgments

This work is supported in part by the California Institute of Technology; an NSF National Young Investigator Award to

P.P.; a STC fund; the Center for Neuromorphic Systems Engineering funded by the National Science Foundation at the California Institute of Technology. We wish to thank all the colleagues that helped us throughout this work, especially Peter Schröder, Paul Debevec, Wolfgang Stürzlinger, Luis Goncalves, George Barbastathis and Mario Munich for very useful discussions. Very special thanks go to Silvio Savarese for his work on the real-time implementation of our algorithm.

References

[1] C.L. Bajaj, F. Bernardini, and G. Xu Xu, "Automatic reconstruction of surfaces and scalar fields from 3D scans", *In SIGGRAPH '95, Los Angeles, CA*, pages 109–118, August 1995.

[2] Paul Besl, *Advances in Machine Vision*, chapter 1 - Active optical range imaging sensors, pages 1–63, Springer-Verlag, 1989.

[3] P.J. Besl and N.D. McKay, "A method for registration of 3-d shapes", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[4] R.L. Bishop and S.I. Goldberg, *Tensor analysis on manifold*, Dove Publications, 1980.

[5] Sylvain Bougnoux, "From projective to euclidean space under any practical situation, a criticism of self-calibration", *Proc. 6^{th} Int. Conf. Computer Vision, Bombay, India*, pages 790–796, January 1998.

[6] Jean-Yves Bouguet, *Visual methods for three-dimensional modeling*, PhD thesis, California Institute of Technology, 1999. Available at: http://www.vision.caltech.edu/bouguetj/thesis/thesis.html.

[7] Jean-Yves Bouguet and Pietro Perona, "3D Photography on your Desk", Technical report, California Institute of Technology, 1997, available at: http://www.vision.caltech.edu/bouguetj/ICCV98.

[8] Jean-Yves Bouguet and Pietro Perona, "3D Photography on your Desk", *Proc. 6^{th} Int. Conf. Computer Vision, Bombay, India*, pages 43–50, January 1998.

[9] Jean-Yves Bouguet, Markus Weber, and Pietro Perona, "What do planar shadows tell us about scene geometry?", *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recogn.*, I:514–520, 1999.

[10] D. C. Brown, "Analytical calibration of close range cameras", *Proc. Symp. Close Range Photogrammetry, Melbourne, FL*, 1971.

[11] D. C. Brown, "Calibration of close range cameras", *Proc. 12^th Congress Int. Soc. Photogrammetry, Ottawa, Canada*, 1972.

[12] J.W. Bruce, "Lines, surfaces and duality", Technical report, Dept. of Pure Mathematics, University of Liverpool, 1992.

[13] J.F. Canny, "A computational approach to edge detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[14] B. Caprile and V. Torre, "Using vanishing points for camera calibration", *IJCV*, 4(2):127–140, March 1990.

[15] Geoffrey Cross and Andrew Zisserman, "Quadric Reconstruction from Dual-Space Geometry", *Proc. 6^{th} Int. Conf. Computer Vision, Bombay, India*, pages 25–31, 1998.

[16] Brian Curless and Marc Levoy, "Better optical triangulation through spacetime analysis", *Proc. 5^{th} Int. Conf. Computer Vision, Boston, USA*, pages 987–993, 1995.

[17] Brian Curless and Marc Levoy, "A volumetric method for building complex models from range images", *SIGGRAPH96, Computer Graphics Proceedings*, 1996.

[18] K. Daniilidis and J. Ernst, "Active intrinsic calibration using vanishing points", *PRL*, 17(11):1179–1189, September 1996.

[19] O.D. Faugeras, *Three dimensional vision, a geometric viewpoint*, MIT Press, 1993.

[20] Olivier Faugeras and Bernard Mourrain, "On the geometry and algebra of the point and line correspondence between n images", *Proc. 5^{th} Int. Conf. Computer Vision, Boston, USA*, pages 851–856, 1994.

[21] H. Gagnon, M. Soucy, R. Bergevin, and D. Laurendeau, "Registration of multiple range views for automatic 3-D model building", *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recogn.*, pages 581–586, June 1994.

[22] A.A. Goshtasby, S. Nambala, W.G. deRijk, and S.D. Campbell, "A System for Digital Reconstruction of Gypsum Dental Casts", *IEEE Transactions on Medical Imaging*, 16(5):664–674, October 1987.

[23] A. Gruss, S. Tada, and T. Kanade, "A VLSI Smart Sensor for Fast Range Imaging", In *DARPA93*, pages 977–986, 1993.

[24] Richard I. Hartley, "A linear method for reconstruction from lines and points", *Proc. 5^{th} Int. Conf. Computer Vision, Boston, USA*, pages 882–887, 1994.

[25] Janne Heikkila and Olli Silven, "A four-step camera calibration procedure with implicit image correction", *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recogn.*, pages 1106–1112, 1997.

[26] R. A. Jarvis, "A perspective on range-finding techniques for computer vision", *IEEE Trans. Pattern Analysis Mach. Intell.*, 5:122–139, March 1983.

[27] T. Kanade, A. Gruss, and L. Carley, "A Very Fast VLSI Rangefinder", In *IEEE International Conference on Robotics and Automation*, volume 39, pages 1322–1329, April 1991.

[28] Reinhard Koch, Marc Pollefeys, and Luc Van Gool, "Multi viewpoint stereo from uncalibrated video sequence", *Proc. 5^{th} European Conf. Computer Vision, Freiburg, Germany*, pages 55–71, June 1998.

[29] Jurgen R. Meyer-Arendt, "Radiometry and photometry: Units and conversion factors", *Applied Optics*, 7(10):2081–2084, October 1968.

[30] Athanasios Papoulis, *Probability, Random Variables and Stochastic Processes*, Mac Graw Hill, 1991, Third Edition, page 187.

[31] Marc Pollefeys, Reinhard Koch, and Luc Van Gol, "Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters", *Proc. 6^{th} Int. Conf. Computer Vision, Bombay, India*, pages 90–95, January 1998.

[32] Marc Pollefeys and Luc Van Gool, "A stratified approach to metric self-calibration", *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recogn.*, pages 407–412, 1997.

[33] Riou, "High resolution digital 3-d imaging of large structures", *SPIE Proceedings, 3-D Image Capture, San Jose*, 3023:109–118, February 1997.

[34] Silvio Savarese, "Scansione tridimensionale con metodi a luce debolmente strutturata", *Tesi di Laurea, Universita degli Studi di Napoli Federico II*, 1998.

[35] A. Shashua and M. Werman, "Trilinearity of three perspective views and its associated tensor", *Proc. 5^{th} Int. Conf. Computer Vision, Boston, USA*, pages 920–925, 1995.

[36] G.P. Stein, "Accurate Internal Camera Calibration Using Rotation, with Analysis of Sources of Error", In *Proc. 5^{th} Int. Conf. Computer Vision, Boston, USA*, pages 230–236, 1995.

[37] Peter F. Sturm and Stephen J. Maybank, "On plane-based camera calibration: A general algorithm, singularities, applications", *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recogn.*, I:432–437, 1999.

[38] Marjan Trobina, "Error model of a coded-light range sensor", Technical Report BIWI-TR-164, ETH-Zentrum, 1995.

[39] R.Y. Tsai, "A versatile camera calibration technique for high accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses", *IEEE J. Robotics Automat.*, RA-3(4):323–344, 1987.

[40] G. Turk and M. Levoy, "Zippered polygon meshes from range images", *In SIGGRAPH '94*, pages 311–318, July 1994.

[41] John W. T. Walsh, *Photometry*, Dover, NY, 1965.

[42] L.L. Wang and W.H. Tsai, "Computing camera parameters using vanishing-line information from a rectangular parallelepiped", *MVA*, 3(3):129–141, 1990.

[43] Y.F. Wang, "Characterizing three-dimensional surface structures from visual images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(1):52–60, 1991.