EUROGRAPHICS 2002 / I. Navazo Alvaro and Ph. Slusallek
(Guest Editors)

Short Presentations

Flexible bump map capture from video

James A. Paterson and Andrew W. Fitzgibbon

Department of Engineering Science, The University of Oxford, UK.

Abstract

We extend recent techniques for the automated capture of surface normal maps or bump maps from real-world material samples. In contrast to recent systems, which have required cumbersome laboratory setups with careful calibration, we propose a moving-camera system which requires only one light source, rigidly attached to the camera. We describe a simple but accurate image-based calibration technique which allows the construction of an approximation to the standard photometric stereo setup. We show how to optimally estimate the system geometry during capture, and demonstrate that the accuracy of the new system is sufficient to allow normals to be computed.

Categories and Subject Descriptors (according to ACM CCS): I.2.10 [Artificial Intelligence]: Vision and Scene Understanding—modeling and recovery of physical attributes. I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—color, shading, shadowing, and texture.

1. Introduction

More and more, computer graphics depends on capture of real-world data for the accurate rendering of real-world objects. Examples include laser scanning, lighting capture[1] and the subject of this paper—the recovery of material properties expressed as BRDF samples[2, 3, 4] or *bump maps*[5, 6]. However, in many applications such capture is difficult because the required equipment is specialized, bulky, or must remain carefully calibrated and thus cannot be moved. This paper shows how one such system can be redesigned using computer vision techniques to permit more flexible bump map capture using readily available hardware.

Bump mapping is a generic term grouping techniques which use 2D texture to apply 3D detail to a surface. The core idea is that shapes and materials may be geometrically approximated relatively coarsely (for example, by large polygons in 3D), with 2D texel maps applying intricate surface detail. Currently, most renderers using bump mapping can be grouped into one of two categories:

- Displacement mapping: Here each texel specifies a (signed) micro distance from the 3D object to which it is applied. Renderers can store this information more compactly and/or render it more quickly than is possible with a dense polygon mesh.
- Normal mapping: Each texel defines the normal to the surface of the object at that point, to be combined with the direction of the general surface normal at that point to give an overall local normal. Performing lighting calculations using this modified normal creates the impression of surface texture or roughness without explicit geometric modelling of surface detail.

Rendering of 3D objects with bump mapping can be a computationally expensive operation—few modern graphics pipelines can can directly support bump mapping within hardware, which requires per-pixel lighting calculation rather than traditional per-vertex operations. OpenGL itself requires the use of language extensions to support per-pixel shading, this combined with the computational load of multipass software rendering has led to halfway houses, such as ‘light mapping’, where precomputed lighting effects are applied to objects statically, for given lighting conditions[7].

Bump maps are traditionally generated procedurally, using strategies which are designed to emulate the generic behaviour of certain classes of surface variation. Modelling from the real world can be achieved by obtaining high resolution 2D surface detail from a physical object, but this is an involved process. An alternative strategy, is to use the real world as a source of models—sampling surface shape, or more accurately, surface normal estimates by observing the effect of lighting variation[6, 8].

Photometric stereo

By considering the standard Lambertian lighting model with no ambient light, we observe that the normal to a surface at a given location can be found by observation of the intensity of the point on the image over different lighting conditions. By recording the intensity of the image of the object illuminated with a single light at three different known positions, solution of a system of linear equations yields the normal at that point. Below L_o represents a constant—light intensity multiplied by lambertian reflectivity, $l_{[1,2,3],[1,2,3]}$ denotes three sets of light directions, $n_{[1,2,3]}$ the normal vector, and $i_{[1,2,3]}$ the observed intensities at the points on the surface up to scale α.

$$L_o \begin{bmatrix} l_{1,1} & l_{1,2} & l_{1,3} \\ l_{2,1} & l_{2,2} & l_{2,3} \\ l_{3,1} & l_{3,2} & l_{3,3} \end{bmatrix} \begin{bmatrix} n_1 \\ n_2 \\ n_3 \end{bmatrix} = \begin{bmatrix} \alpha i_1 \\ \alpha i_2 \\ \alpha i_3 \end{bmatrix}$$

If relative light and object positions are known, and the camera is fixed relative to the object, these equations are readily solved. In the case where the Lambertian approximation is poor, or shadowing must be accounted for, extensions to the model allow accurate reconstruction of the surface normals. Our goal in this paper is to satisfy the geometric conditions: in testing we use only the simplest lighting model. Although more complex models would be expected to provide more accurate normals, the simple model is adequate to test our geometry.

Existing techniques for bump map capture

A typical experimental set up, as exemplified by Rushmeier et al[6], uses a fixed camera and object with a set of identical lights at known relative positions to the object to provide the required illumination. Figure 1 sketches such a setup. The accuracy of this model is of course limited by the ability of the Lambertian lighting model to predict actual lighting effects. Three common properties of physical objects which can produce non-Lambertian conditions are specular highlights, reflection and self-shadowing. Highlights and shadows are avoided by discarding particularly low or high intensity values. For accurate results, images must be captured under several lighting configurations, requiring a complex rig with limited portability. Rushmeier et al use 5 lights, and suggest that at least this number are required for accurate capture. In their extension of the process to a moving rig[9], the rig retains 5 lights, rigidly attached to the camera, and is still somewhat cumbersome.

Flexible bump map capture

We propose an alternative capture rig, requiring only a single light and camera, which is moved relative to the sample (or vice versa). Its limitation is that it requires that the sample to be analysed will be modelled as a single planar surface. However, many samples of interest satisfy this approximation, particularly as the intent of bump map capture is to allow coarse planar approximations to be combined with dense normal maps. Figure 2 illustrates two equivalent embodiments of such a system. For our experiments, we use a fixed light and camera, with the sample being moved; but for more flexible capture, rigidly attaching a light to the camera would allow an extremely portable capture system. A summary of the system follows. The rest of the paper describes these steps in more detail.

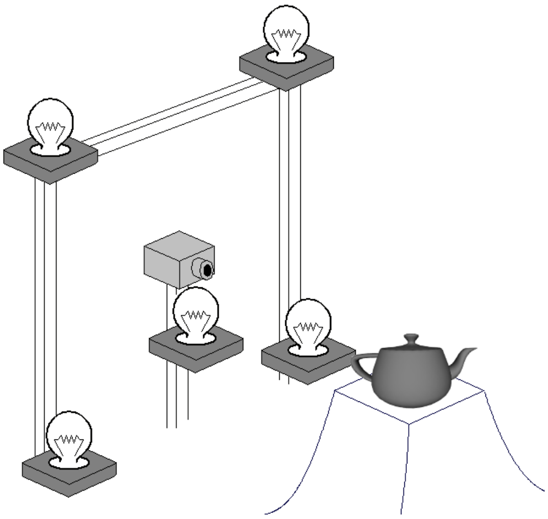

Figure 1: ***A conventional rig for bump map capture****. The positions of the camera and several light sources must be accurately calibrated, and the work area is limited in size.*

In operation, a (video) sequence of images of the object is taken, with the relative position of the camera and light fixed. Four calibration markers are placed on or near the sample, and by application of computer vision techniques the relative position of the light/camera rig is calculated, and an inverse perspective transformation applied to re-render the image of the object as fronto-parallel (see Figure 3). The resulting image sequence is analogous to the situation of a static camera and object with several lights[6]. We observe that, in terms of the relative geometry of the capture rig the two situations: static object, moving rig; and static rig, moving object are equivalent. Furthermore, the relative positions of camera and light are fixed, so a single calibration step suffices for capture of a large number of light-direction samples.

2. Details

The following subsections detail the design and construction of a prototype of our system. We divide the discussion into two parts. In the first, we do not refer to light source position, and describe only the recovery of camera position relative to the object's markers. This will allow us, in §2.3, to describe an easy technique for calibrating the relative position of the camera and light source. The guiding principle

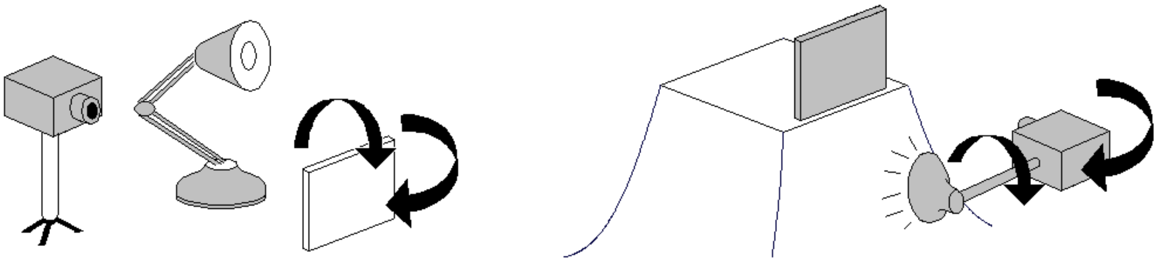

Figure 2: ***Two equivalent single-light-and-camera rigs.*** *The rig on the left has fixed camera and light, with the sample moving on a planar backplate. On the right, the light is attached to the camera, and the combined light/camera rig is moved. In this case, the light is shown on an extension to ensure sufficient baseline between its centre and the optical centre.*

.

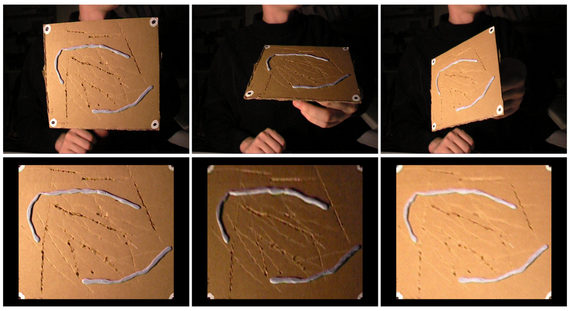

Figure 3: ***Inverse perspective mapping.*** *The markers placed on the sample are tracked from frame to frame, and the object re-rendered to appear fronto-parallel. The resulting images approximate the appearance of a fixed object with moving light source, so photometric stereo may be applied to compute normals.*

of these steps is that any measurements or construction required of the user should be as insensitive to human error as possible. For example, we assume it is straightforward to accurately place markers on a measured rectangle, but not to measure the distance from the light source centre to the (poorly defined) optical centre of the camera. For concreteness we shall describe the system from the point of view of figure 2a, with a fixed camera and light, and moving sample on a planar backing plate.

2.1. Estimating camera position

Reference to figure 3 will reveal four high-contrast markers placed on the sample's planar backing plate. The positions of these markers is assumed known (although assuming them approximately known is an easy extension). This section describes how the camera-to-sample transformation is computed, which in combination with the calibration information of the next section, gives the sample-to-light vector.

Several techniques are possible for the recovery of camera position from known 3D points, and the book by Hartley and Zisserman[10] provides a review. We use the technique of Simon et al[11], which is fast and accurate for coplanar markers.

We may temporarily choose world coordinates in the sample's plane coordinate system, which we arbitrarily define to be the xy plane. Denote the four marker positions in that plane by $\mathbf{p}_{1..4}$ where $\mathbf{p}_i = (p_i, q_i, 0, 1)^\top$ is the representation in homogeneous coordinates. We note also that we may write these as homogeneous 2D points in the plane by omitting the third component, giving $\bar{\mathbf{p}}_i = (p_i, q_i, 1)^\top$.

We are given a video sequence of images of the object in various positions, and observe the 2D image coordinates $\bar{\mathbf{x}}_{1..4}$ of the projections of the 3D points. The camera position is defined by the 3×4 projection matrix

$$\mathtt{P} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$$

where $\mathtt{R}$ is a 3×3 rotation matrix, and $\mathbf{t}$ is the translation of the camera. The matrix $\mathtt{K}$ represents the internal calibration parameters of the camera:

$$\mathtt{K} = \begin{bmatrix} f & s & u_0 \\ 0 & af & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad \begin{cases} f \text{ is focal length;} \\ (u_0, v_0) \text{ is principal point;} \\ a \text{ is aspect ratio; } s \text{ is skew} \end{cases}$$

We assume that $\mathtt{K}$ is approximately known—for the DV camcorder used in our experiments, it was sufficient to guess a focal length of 1000, and assume square pixels ($a = 1, s = 0$), with principal point at the image centre. We will need to work with the columns of $\mathtt{R}$, which we denote by $\mathbf{r}_{1..3}$, so $\mathtt{R} = [\mathbf{r}_1 \mid \mathbf{r}_2 \mid \mathbf{r}_3]$.

Observing the action of $\mathtt{P}$ on the markers $\mathbf{p}_{1..4}$, we may write

$$\bar{\mathbf{x}}_i = \mathtt{K}[\mathbf{r}_1 \mathbf{r}_2 \mathbf{r}_3 \mathbf{t}] \begin{pmatrix} p_i \\ q_i \\ 0 \\ 1 \end{pmatrix}.$$

Noting that the third column of $\mathtt{P}$ is always multiplied by the (zero) z coordinate of $\mathbf{p}$, the transformation becomes a 2D homography

$$\begin{aligned}\bar{\mathbf{x}}^i &= \mathtt{K}[\mathbf{r}_1\ \mathbf{r}_2\ \mathbf{t}]\bar{\mathbf{p}}_i \\ &= \mathtt{H}\bar{\mathbf{p}}_i\end{aligned}$$

Where the 3×3 matrix $\mathtt{H}$ parametrizes the homography. Efficient linear methods exist[10] to compute $\mathtt{H}$ from the four point correspondences $\bar{\mathbf{x}}_i \leftrightarrow \bar{\mathbf{p}}_i$.

On computing $\mathtt{H}$, the remaining column of $\mathtt{R}$ is then computed as follows. In the absence of noise, the matrix $\mathtt{M} = \mathtt{K}^{-1}\mathtt{H}$ has columns $[\mathbf{r}_1\ \mathbf{r}_2\ \mathbf{t}]$ so $\mathbf{r}_3$ may be computed as the cross product $\mathbf{r}_1\times\mathbf{r}_2$. In practice, due to inaccuracies in the measurements of the image positions $\mathbf{x}_i$, the columns of $\mathtt{M}$, now denoted $[\mathbf{m}_1\ \mathbf{m}_2\ \mathbf{m}_3]$ will not satisfy the orthonormality constraints on the columns of a rotation matrix: $\|\mathbf{m}_1\| = \|\mathbf{m}_2\| = 1$ and $\mathbf{m}_1\cdot\mathbf{m}_2 = 0$. A more accurate estimate of the rotation may be obtained by first truncating $\mathtt{M}$ to the special form where these constraints are satisfied. Figure 4 illustrates the process, i.e. construction of an orthonormal basis set from a pair of 3D directions. As the homography is defined only up to scale (as it is a mapping between projective spaces), the matrix may be assumed scaled so that $(\|\mathbf{m}_1\|+\|\mathbf{m}_2\|) = 2$. Then the following steps yield a matrix which satisfies the appropriate constraints.

1. Normalize $\mathbf{m}_1$, $\mathbf{m}_2$ i.e. $\mathbf{m}_i = \mathbf{m}_i/\|\mathbf{m}_i\|$.
2. $\mathbf{r}_3 = \mathbf{m_1}\times\mathbf{m_2}$. Normalize $\mathbf{r}_3$.
3. $\mathbf{tmpa} = \mathbf{m_1}+\mathbf{m_2}$. Normalize $\mathbf{tmpa}$.
4. $\mathbf{tmpb} = \mathbf{r_3}\times\mathbf{tmpa}$. Normalize $\mathbf{tmpb}$.
5. Set $\mathbf{r}_1 = \mathbf{tmpb}+\mathbf{tmpa}$. Normalize $\mathbf{r}_1$.
6. Set $\mathbf{r}_2 = \mathbf{tmpb}-\mathbf{tmpa}$. Normalize $\mathbf{r}_2$.

Assembling the columns $\mathbf{r}_{1,2,3}$ into $\mathtt{R}$ completes the process.

2.2. Nonlinear refinement of reprojection error

The preceding paragraphs describe the estimation of camera position from measured 2D point positions, and provide reliable solutions via closed-form (or provably convergent) algorithms such as eigenvalue computation or the singular value decomposition[12]. However, the estimates can be significantly improved at small computational cost by a maximum likelihood estimation of the parameters[10]. This is a process of nonlinear minimization of the forward model which generates the 2D point tracks. This process optimizes the error in the system not by least-squares approximations of matrices as above, but at its source, namely the image plane. The free parameters of the system are the six parameters of camera pose, and one for focal length. Given the known marker positions and image points, written $\vec{x}_i$ in nonhomogeneous coordinates, the objective function to be minimized is

$$\varepsilon(\mathtt{R},\mathbf{t},f) = \sum_{i=1}^{4}\|\vec{x}_i - \pi(\mathtt{K}[\mathtt{R}\ \ \mathbf{t}]\,\mathbf{p}_i)\|^2$$

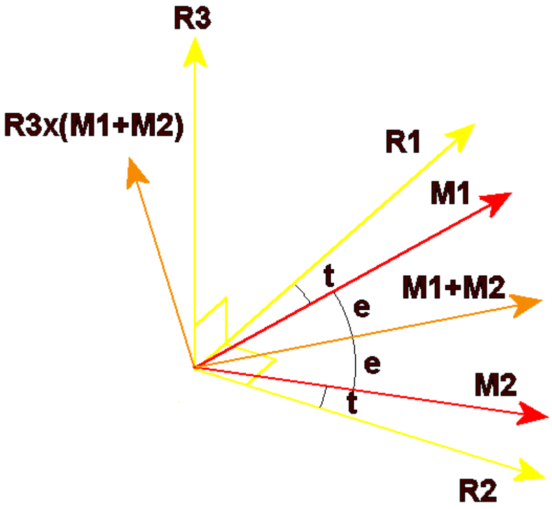

Figure 4: ***Constructing an orthonormal basis.*** *Given a pair of 3D directions $(M1,M2)$, the closest orthonormal pair $(R1,R2)$ and their cross product $R3$ are computed.*

Where the projection function $\pi([x,y,z]^\top) = (x/z,y/z)$. Parametrization of the rotation $\mathtt{R}$ avoids the singularities of Euler angles or equivalent three-parameter forms by using a quaternion. The gauge freedom introduced by this overparametrization is handled via a publically available implementation of the Levenberg-Marquardt algorithm[13]. This minimization of the *reprojection error* directly reduces the image-plane difference between the projected 3D points and their associated 2D markers, computing the optimal transformation in terms of registering the images.

Typical performance is exemplified by the images in figure 3, where the RMS distance of markers to the reprojected points is typically around 25 pixels before the nonlinear optimization, reducing to around 5 in total afterwards.

2.3. Estimating light position

The system as described to this point covers the estimation of camera position and *rectification* of the image into an approximation to the fronto-parallel view. The remaining task is to compute the position of the light source for each rectified image, which is the focus of this section. The primary constraint on light source position is that the relative positions of light source and camera are fixed during the acquisition—only the relative position of the light and camera system changes. Therefore, if we know the position of the light source in camera coordinates before acquisition, we can transform this position to sample coordinates to obtain a light-source direction for each of the registered fronto-parallel views.

Given accurate measuring equipment, and accurate

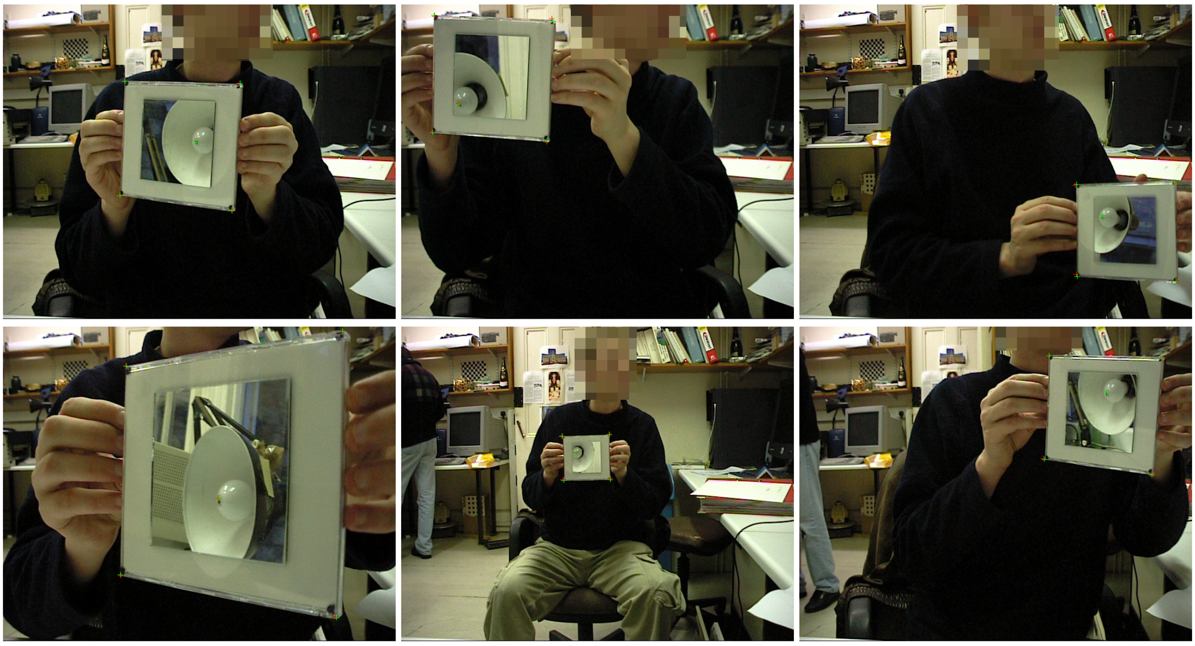

Figure 5: ***Calibration of light-source positions***. *Green crosses show the detected points, red the reprojections from the model. The model is estimated by minimizing the reprojection error of all five points in all images simultaneously.*

knowledge of the 3D position of the effective optical centre of the camera, it would be possible to physically measure the light's position in camera coordinates. However, such measurements are inconvenient, time-consuming to obtain, and extremely difficult to do accurately. Fortunately a more accurate solution can be obtained with less effort by adopting an image-based calibration procedure. By placing a mirror on the sample backing plate, the plate can be moved in front of the camera so that the reflection of the light source is visible. Figure 5 shows several images of this procedure. In each image, the position of (the reflection of) the light source is manually specified using the mouse.

From one such image, the relative position of the sample plane and the camera coordinate system can be computed as described in §2.1, and it is a matter of straightforward geometry to go from the 2D coordinates of the image of the light centre to determine the 3D line (in camera coordinates) on which the light source must lie. From two such images, two such 3D lines are obtained, in camera coordinates, both of which include the light source—Figure 6 shows a typical arrangement. In exact geometry, the intersection of these lines provides the light-source position, and calibration is complete. We observe that the calibration remains valid as long as the parameters of the light-camera system (most importantly their relative position) remains constant. This means that calibration can be carried out 'off-line' for portable capture rigs, such as the second system in Figure 2.

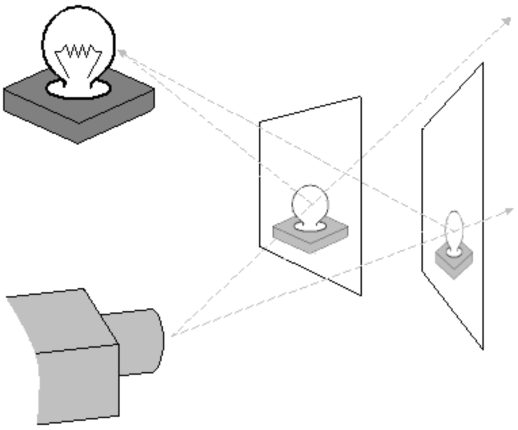

Figure 6: ***Geometry of calibration of light position***. *At least two mirror positions are required to calculate the light position. Note that in this picture, we consider the camera and light to be fixed while the mirror moves. The two mirrors indicate mirror position at two different times. Combining the information at more than two mirror positions allows a more accurate least-squares estimate to be obtained.*

In practice, of course, the lines are not accurately computed, due to inaccuracies in the localization of the backing plane corners and the manual indication of the light-source centre. By using more than the minimum of two images, a least-squares estimate[14] of the intersection point can be computed, which will improve accuracy. However, the estimate will be biased unless the errors in the 3D lines are accurately modelled. The optimal estimate is obtained[10] using an extension of the nonlinear estimator described in §2.2.

Suppose V views are available. The parameters of the system are: V camera positions $(\mathtt{R}_i, \mathbf{t}_i)_{i=1}^{V}$, the camera focal length f, and the light position $\mathbf{L}$. These parameters determine the light position with optimal camera transformations. Let $\vec{x}_{[k]i}$ be the image of $\mathbf{p}_i$ in the kth image. The reflection of the light through the xy plane may be writted as a constant transformation matrix $\mathtt{F}$ premultiplying $\mathbf{L}$, and the 2D image coordinates of the center of the light in image k are denoted $\vec{x}_{[k]5}$. The objective function to be minimised is then:

$$\varepsilon(\mathtt{R}_1 \ldots \mathtt{R}_V, \mathbf{t}_1 \ldots \mathbf{t}_V, f, \mathbf{L}) = \sum_{k=1}^{V} \varepsilon_k$$

where ε_k, the error in image k, is

$$\varepsilon_k = \left(\sum_{i=1}^{4} \underbrace{\|\vec{x}_{[k]i} - \pi(\mathtt{K}\,[\mathtt{R}_k \;\; \mathbf{t}_k]\,\mathbf{p}_{[k]i})\|^2}_{\text{Reprojection error of point } i \text{ in view } k} \right) + \\ + \underbrace{\|\vec{x}_{[k]5} - \pi(\mathtt{K}\,[\mathtt{R}_k \;\; \mathbf{t}_k]\,\mathtt{F}\mathbf{L})\|^2}_{\text{Reprojection error of light in view } k}$$

This minimization has the effect of optimally distributing the 2D localization error of the 3D components. The 3D rays are guaranteed to intersect, as they all emerge from the light source $\mathbf{L}$. This means that *ad hoc* schemes for 3D error propagation are not needed.

To demonstrate the accuracy of light position estimation, $V = 6$ different images of the mirror were used. Figure 5 is used to demonstrate the effectiveness of the non-linear optimisation. In each calibration image, green crosses indicate $\vec{x}_{[k]i}$, and red crosses $\pi(\mathtt{K}\,[\mathtt{R} \;\; \mathbf{t}]\,\mathbf{p}_{[k]i})$. Again the reduction in average reprojection error is from 7 pixels at the initial estimate to around 1 to 2 pixels at convergence. Execution time for the calibration stage is a few seconds on a modern desktop PC.

2.4. Tracking the markers

Whilst digital still cameras offer much better resolution than a DV camcorder, one major advantage of video over stills is that it is easy to capture large quantities of frames in very little time. By automating the detection of the position of the high-contrast markers in each sample image, we can easily capture data for hundreds of camera positions. A simple ‘brute force’ search for the marker points has proved relatively effective so far. This works by storing the images and positions of the markers in the first image (the positions of the markers having been manually defined for the first frame). The search then examines each possible location for the markers in the following images and the locations which give minimal root-sum-squared pixel difference from the new image are returned. The computational overhead of this method can be large, and is reduced by limiting the search to within a specfied radius of pixels from the position of the center of the marker in the previous image. One possible improvement to this method is to use an adaptive search, i.e. to update the stored images of the markers after finding them in the new image.

2.5. The capture process

Our experimental setup followed the first arrangement suggested in figure 2, i.e. a static light and camera with moving sample. Equipment comprised a Canon DV camcorder, tripod, a standard office angle-poise lamp, along with a roughly 140mm by 110mm rectangular sample. A handle was attached to the back of the sample so it could be manipulated within the camera view manually.

Calibration was carried out under room lighting. For capture, we needed the capture sequence of the sample to be as close to zero ambient light as possible, so only the desk lamp illuminated the scene.

Identification of the corners of the mirror in the image is facilitated by the having the observed size of the mirror in the image as large as possible. A reasonably sized mirror proved rather ungainly to manipulate, and so instead a small much lighter mirror was attached to the reverse of a larger plastic case.

Once the light position relative to the camera $\mathbf{L}$ and focal length f has been computed, the system is calibrated. We are now able to calculate the direction of the light for any point $\mathbf{Q}$ on the object, given the camera position. (We observe that $\mathbf{Q}$ will be of the form $\mathbf{Q} = (x, y, 0, 1)^\top$):

$$\mathit{Direction} = \begin{bmatrix} \mathtt{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix}^{-1} \mathbf{L} - \mathbf{Q}$$

Figure 7 shows the set of inverse perspective mappings from Figure 3, overlaying a 2D projection of the calculated direction to the light from the center point of the sample.

The final stage is to collate the inverse perspective mappings and use them to create normals as described in §1. The density of the field is in real terms limited by the resolution of the camera—the quality of the reconstructed fronto-parallel view is reduced as the angle between the object and the camera increases. Even so, Figure 3 shows that a reasonably good re-rendering is possible despite fairly severe foreshortening in the image.

For the purposes of demonstration, a 20 by 18 grid of normals was constructed from 3 images of the sample. These

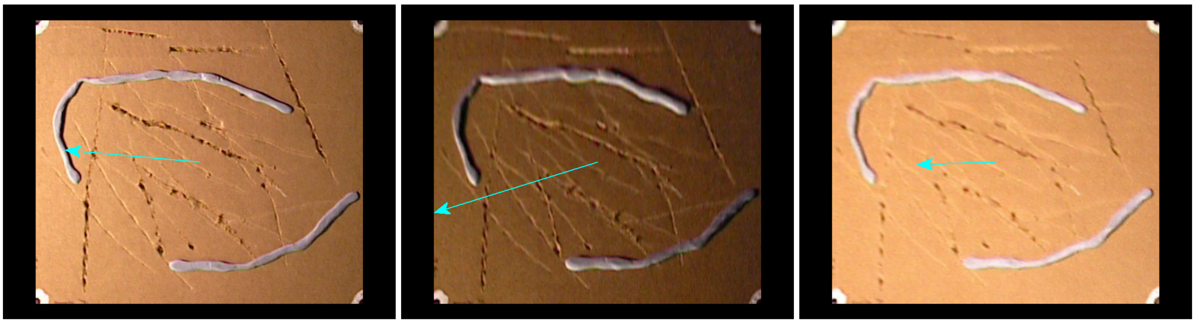

Figure 7: ***Calculated light direction.*** *The light direction, projected into the fronto-parallel view, is superimposed on the rectified images. This is the input to a photometric stereo algorithm which recovers surface normals. Comparing the light source direction to the shadow directions indicates that the direction is qualitatively well estimated.*

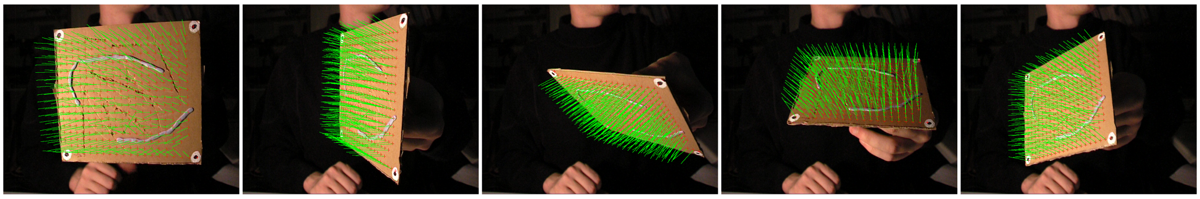

Figure 8: ***Calculated normal map.*** *Computed surface normals are superimposed onto the five captured images of the sample used to generate the map. Because of the deviation from Lambertian imaging conditions, there is a systematic error in the normals, but their local consistency gives us confidence that flexible capture is as accurate as traditional, laboratory-bound systems.*

were then overlaid on the input sample image for inspection. As noted by Rushmeier et al[6], highlights and shadowing mean that some of the calculated intensity values must be rejected, leaving some positions on the object with fewer than the required three values for normal computation. Inclusion of additional images of the sample provides the necessary data—a topic for further investigation is how to optimise the choice of which fronto-paralllel views are used in the normal calculation. Even a short captured sequence will contain many hundreds of images—we would intend to maximise the range of object orientations used during the computation of the surface normals, whilst avoiding images containing extremely large object-camera angles due to the suspected loss in re-rendering quality. Typical processing time on a desktop PC is around 5 seconds per image, leading to a total time of approximately 2 hours for a 60 second sequence.

3. Conclusions

We see in Figure 8 a set of well formed calculated normals, with a generally smooth continuity of direction over the board. A more effective way to examine the results is to re-render the sample as a normal mapped quadrilateral, which can be observed in Figure 9. Examination of these images reveals a relatively good representation of the sample—the sample has been successfully approximately as discussed in §1. As with all bump mapping techniques, secondary surface effects such as self-shadowing are not modelled. This characteristic is shared by our computed bump maps—the rendered surface is less convincing at acute angles to the camera, such in as the second image in Figure 9.

Although non-Lambertian effects on the surface mean there is a systematic error in normal direction at the extremities, we are confident that this error is common to both fully calibrated[6] and flexible bump map capture. Ongoing work includes the implementation of more sophisticated techniques to estimate the normal map in order to accurately quantify the system's efficacy.

An interesting extension is to the case where the marker positions on the planar surface are known only approximately. This might happen if only archive footage is available, and four arbitrary high-contrast points are tracked. Then approximate world coordinates for the markers can be guessed, but must be refined in order to obtain an accurate normal map. Given enough images, we can in fact include the positions of the markers in the nonlinear optimization, noting that $\mathbf{p}_1$ and $\mathbf{p}_2$ may be arbitrarily assigned the origin and x-axis direction, but the remaining pair must be added to the parametrization of the error function ε.

This paper has shown that recent systems for bump map

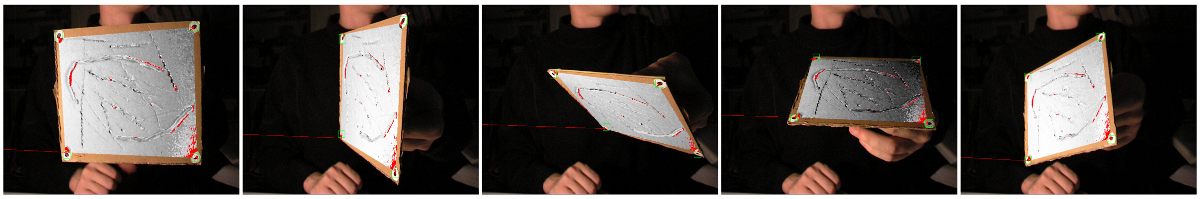

Figure 9: ***Rendered normal map.*** *By applying the Lambertian lighting model (computing the angle between the light and surface normal for each texel in the normal map) we can render the normal map and overlay it in the corresponding image in the input sequence. Red pixels indicate there was insufficient data to compute a normal at that point, showing the need for greater than 5 input images.*

capture using calibrated light sources and static cameras may be made more flexible by allowing the movement of a light/camera rig, if the surface to be sampled is of low curvature. We have introduced a novel technique for calibration of such a rig and demonstrated the computation of surface normals from the system's output. We expect that future enhancements will allow the extension of the system to highly curved samples.

References

1. P. Debevec, T. Hawkins, C. Tchou, H.-P. Duiker, W. Sarokin, and M. Sagar. Acquiring the reflectance field of a human face. In *Proceedings, SIGGRAPH*, pages 145–156, 2000. 1

2. S. Marschner, S. Westin, E. Lafortune, K. Torrance, and D. Greenberg. Image-based BRDF measurement including human skin. In *Proceedings, Eurographics Workshop on Rendering*, pages 131–144, 1999. 1

3. K. J. Dana, B. van Ginneken, S. K. Nayar, and J. J. Koenderink. Reflectance and texture of real-world surfaces. *ACM Transactions on Graphics*, 18(1):1–34, 1999. 1

4. H. Lensch, J. Kautz, M. Goesele, W. Heidrich, , and H.-P. Seidel. Image-based reconstruction of spatially varying materials. In *Proceedings, Eurographics Rendering Workshop*, 2001. 1

5. J. Blinn. Models of light reflection for computer synthesized pictures. In *Proceedings, SIGGRAPH*, pages 192–198, 1977. 1

6. H. Rushmeier, G. Taubin, and A. Guéziec. Applying shape from lighting variation to bump map capture. In *Proceedings, Eurographics Workshop on Rendering*, pages 35–44, 1997. 1, 2, 7

7. Rocchini C. Tarini M., Cignoni P. and Scopigno R. Real time, accuracte,multi-featured rendering of bump mapped surfaces. In *Proceedings, Eurographics*, 2000. 1

8. R.J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):139–144, 1980. 1

9. H. Rushmeier, F. Bernardini, J. Mittleman, and G. Taubin. Acquiring input for rendering at appropriate levels of detail: Digitizing a Pietà. In *Proceedings, Eurographics Workshop on Rendering*, 1998. 2

10. R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000. 3, 4, 6

11. G. Simon, A. Fitzgibbon, and A. Zisserman. Markerless tracking using planar structures in the scene. In *Proceedings, International Symposium on Augmented Reality (ISAR)*, 2000. 3

12. G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, MD, second edition, 1989. 4

13. B. S. Garbow, K. E. Hillstrom, and J J. Moré. Documentation for MINPACK subroutine LMDIF. Technical report, Argonne National Laboratory, 1980. http://www.math.utah.edu/software/minpack/minpack/lmdif.html. 4

14. K. Kanatani. *Statistical Optimization for Geometric Computation: Theory and Practice*. Elsevier Science, Amsterdam, 1996. 6