Vision, Video, and Graphics (2003)
P. Hall, P. Willis (Editors)

An Eigenvector Method for Surface Recovery

Antonio Robles-Kelly[1] and Edwin R. Hancock[2]

Department of Computer Science
The University of York
York, Y01 5DD, UK
[1]arobkell@cs.york.ac.uk [2]erh@cs.york.ac.uk

Abstract

In this paper we explore how spectral methods for graph seriation can be used to develop a new shape-from-shading algorithm. We characterise the field of surface normals using a transition matrix whose elements are computed from the sectional curvature between different image locations. We use a graph seriation method to define a curvature minimising surface integration path for the purposes of height reconstruction. To smooth the reconstructed surface, we fit quadric patches to the height data. The smoothed surface normal directions are updated ensuring compliance with Lambert's law. The processes of height recovery and surface normal adjustment are interleaved and iterated until a stable surface is obtained. We provide results on synthetic and real-world imagery.

Categories and Subject Descriptors (according to ACM CCS): I.4.8 [Image Processing and Computer Vision]: Spectral-graph methods, shape-from-shading

1. Introduction

Graph spectral methods have recently found widespread use in computer vision. Concrete examples include the use of normalised cuts for image segmentation [1], the use of matrix factorisation methods for correspondence matching [2] and spectral encodings for shock-graph indexation and retrieval [3]. Most of these methods use eigenvectors for the purposes of clustering. However, graph-spectral methods can also be used to solve path-based problems. The most familiar example here is the minimum linear arrangement problem, which involves placing the nodes in a graph in an order which minimises the total edge length. A related task is the graph seriation problem [4], which involves ordering the set of nodes in a graph in a sequence such that strongly correlated elements are placed next to one another. There are many problems in computer vision that could potentially be posed as graph-seriation. In low and intermediate level vision, contour grouping can be posed as the task of finding a curve of minimum length and curvature across of pixel lattice. At higher levels of abstraction path planning tasks such as the travelling salesman problem, can be posed as finding a minimum length path on a graph.

The seriation problem can be approached in a number of ways. Clearly the problem of searching for a serial ordering of the nodes, which maximally preserves the edge ordering is one of exponential complexity. As a result approximate solution methods have been employed. These involve casting the problem in an optimisation setting. However, recently a graph-spectral solution has been found to the problem. Atkins, Boman and Hendrikson [4] have shown how to use an eigenvector of the Laplacian matrix to sequence relational data. The method has been successfully applied to the consecutive ones problem and a number of DNA sequencing tasks.

In this paper our aim is to exploit graph-spectral seriation for a problem from intermediate level vision. The problem chosen is surface reconstruction using shape-from-shading. The original work on shape-from-shading adopted a variational approach and aimed to recover surface normal directions by applying Euler's method to a regularised energy function [5]. There have since been a multitude of papers attempting to improve the method and render it useful for practical shape-analysis tasks (for a good recent review, see the comprehensive comparative study of Zhang et al [6]). However, recently there has a consolidated effort in the literature aimed at overcoming the well documented problems with shape-from-shading. One of the most important of these is the method of Dupuis and Oliensis [7] which does not use regularisation and has been proved to reconstruct height information correctly from intensity data. The method propa-

gates surface normals in the steepest downhill direction from singular points on the surface. Bichsel and Pentland [8] have developed a fast variant of the method. Worthington and Hancock have developed a new framework for shape-from-shading [9] in which the image irradiance equation is treated as a hard constraint. They have also shown how curvature consistency constraints can be used to recover meaningful topographic surface structure. According to this geometric framework, the surface normals are constrained to fall on an irradiance cone whose axis is in the light source direction and whose apex angle is proportional to the inverse cosine of the measured image brightness. The azimuthal angle of the surface normal on the irradiance cone is determined by local smoothness contraints. However, the method only delivers fields of surface normals and height recovery requires a surface integration algorithm to be applied as a postprocessing step.

We use the graph-spectral seriation method to develop a shape-from-shading algorithm. We commence with the surface normals on their irradiance cones pointing in the direction of the image gradient. The vector field is characterised as a weighted graph using an affinity matrix whose elements are determined using the sectional curvature between locations in the field of surface normals. We recover a curvature minimising path through the field of surface normals by performing an eigenvector analysis on the affinity matrix. By traversing this path we perform height reconstruction by applying the trapezoid rule to the surface normal directions. Surface smoothing is effected by fitting quadric patches to the height data. The patch-fits also allow refined estimates of the surface normals to be made. The image irradiance equation may be imposed by projecting these surface normals onto the irradiance cones. The algorithm may hence be iterated until convergence is reached.

2. Lambertian Reflectance

In the case of Lambertian reflectance from a matte surface of constant albedo illuminated with a single point light-source, the observed intensity is independent of the viewing direction. The observed intensity depends only on the quantity of absorbed light, and this in turn is proportional to the cosine of the incidence angle. Suppose that $\vec{L}$ is the unit-vector in the direction of the light source and that $\vec{N}_i$ is the unit-vector in the surface normal direction at the pixel i. According to Lambert's law, the observed image intensity at the pixel indexed i is $E_i = \vec{N}_i \cdot \vec{L}$.

Lambert's equation provides insufficient information to uniquely determine the surface normal direction. However, as recently observed by Worthington and Hancock [9], the equation does have a simple geometric interpretation which can be used to constrain the direction of the surface normal. The equation specifies that the surface normal must fall on the surface of a right-cone whose axis is aligned in the light-source direction $\vec{L}$ and whose apex angle is $\arccos(E)$. We can use this constaint to acquire intial surface normal directions. The surface normals are placed on the position on the irradiance cone where their projections onto the image plane are aligned in the direction of the local (Canny) image gradient.

3. Integration Path from Graph Spectral Seriation

We provide a graph-spectral characterisation of the field of surface normals by computing an affinity matrix whose elements are given by $W_{i,j} = \exp[-\beta\kappa_{i,j}^2 l_{i,j}]$ where $\kappa_{i,j}$ is the sectional curvature between points i and j and $l_{i,j}$ is the length of the connecting path. The curvature can be approaximated using the surface normal directions $\vec{N}_i$ and $\vec{N}_j$ at the locations i and j. Assuming that the underlying surface is approximately circular we find that

$$\kappa_{i,j}^2 = \frac{2(1-\vec{N}_i.\vec{N}_j)}{l_{i,j}^2} \tag{1}$$

Our aim is to find a curvature minimising path through the field of surface normals by performing an eigenvector analysis of the matrix W, and to use this path for surface height recovery. To commence, we pose the problem in a graph-based setting. The set of pixel sites can be viewed as a weighted graph $G = (V,E,W)$ with index-set V, edge-set $E = \{(i,j)|(i,j) \in V \times V, i \neq j\}$ and weight function $W : E \rightarrow [0,1]$. Let the curvature minimising path commence at the node j_1 and proceed via the sequence of edge-connected nodes $\Gamma = \{j_1, j_2, j_3, ...\}$ where $(j_i, j_{i-1}) \in E$. Further, we suppose that the transition weight matrix $W(j_i, j_{i+1})$ associated with the move between the nodes j_i and j_{i+1} can be regarded as a pairwise similarity measure. With these ingredients, the problem of finding the path that minimises the curvature between adjacent pixel-sites can be viewed as one of seriation, subject to edge connectivity constraints.

As noted by Atkins, Boman and Hendrikson [4], many applied computational problems, such as sparse matrix envelope reduction, graph partitioning and genomic sequencing, involve ordering a set according to a permutation $\pi = \{\pi(j_1), \pi(j_2), \ldots, \pi(j_{|V|})\}$ so that strongly related tokens are placed next to one another. The seriation problem is that of finding the permutation π that satisfies the condition $\pi(j_i) < \pi(j_k) < \pi(j_l) \Rightarrow \{W(i,k) \geq W(i,l) \wedge W(k,l) \geq W(i,l)\}$ This task has been posed as a combinatorial optmisisation problem which involves minimising the penalty function $g(\pi) = \sum_{i=1}^{|V|} \sum_{k=1}^{|V|} W(i,k)\big(\pi(j_i) - \pi(j_k)\big)^2$ for a set of N elements and a symmetric, real transition weight matrix W.

Unfortunately, the penalty function $g(\pi)$, as given above, does not impose edge connectivity constraints on the ordering computed during the minimisation process. Furthermore, it implies no directionality in the transition from the node indexed j_i to the one indexed j_{i+1}. To overcome these shortcomings, we turn our attention instead to the penalty func-

tion

$$g(\pi) = \sum_{i=1}^{|V|-1} W(i,i+1)\big(\pi(j_i) - \pi(j_{i+1})\big)^2 \qquad (2)$$

where the nodes indexed j_i and j_{i+1} are edge connected. After some algebra, it is straightforward to show that

$$g(\pi) = \sum_{i=1}^{|V|-1} W(i,i+1)(\pi(j_i)^2 + \pi(j_{i+1})^2) - 2\sum_{i=1}^{|V|-1} W(i,i+1)\pi(j_i)\pi(j_{i+1}) \qquad (3)$$

It is important to note that $g(\pi)$ does not have a unique minimiser. The reason for this is that its value remains unchanged if we add a constant amount to each of the co-efficients of π. We also note that it is desirable that the minimiser of $g(\pi)$ is defined up to a constant λ whose solutions are polynomials in the elements of W. Therefore, we subject the minimisation problem to the constraints

$$\lambda\pi(j_i)^2 = \sum_{k=1}^{|V|} W(k,i)\pi(j_k)^2 \text{ and } \sum_{k=1}^{|V|} \pi(j_k)^2 \neq 0 \qquad (4)$$

Since the co-efficients $\pi(j_{i+1})$ are inversely proportional to $\lambda - W(i+1,i)$, the co-efficient $\pi(j_{i+1})^2$ increase with decreasing sectional curvature (i.e. the similarity tends to one). The effect of this is to enforce edge connectivity while favouring paths of small local curvature, and also to minimise the overall cost of the path.

Combining the constraint conditions given in Equation 4 with the definition of the penalty function given in Equation 3, it is straightforward to show that the permutation π satisfies the condition $\sum_{k=1}^{|V|}\sum_{i=1}^{|V|-1}\big(W(k,i) + W(k,i+1)\big)\pi(j_k)^2 = \lambda\sum_{i=1}^{|V|-1}(\pi(j_i)^2 + \pi(j_{i+1})^2)$. Using matrix notation, we can write the above equation in the more compact form

$$\Omega W\phi = \lambda\Omega\phi \qquad (5)$$

where $\phi = \{\pi(j_1)^2, \pi(j_2)^2, \ldots, \pi(j_{|V|})^2\}^T$ and Ω is the $(N-1)\times N$ matrix

$$\Omega = \begin{bmatrix} 1 & 1 & 0 & \ldots & 0 \\ 0 & 1 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & 1 & 1 \end{bmatrix} \qquad (6)$$

Hence it is clear that locating the permuation π that minimises $g(\pi)$ can be posed as an eigenvalue problem, and that ϕ is an eigenvector of W. This follows from the fact that Equation 13 can be obtained by multiplying both sides of the eigenvector equation $W\phi = \lambda\phi$ by Ω. Furthermore, due to the norm condition of the eigenvector, the constraint $\sum_{k=1}^{|V|}\pi(j_k)^2 \neq 0$ is always satisfied. Taking this analysis one step further, we can premultiply both sides of Equation 5 by

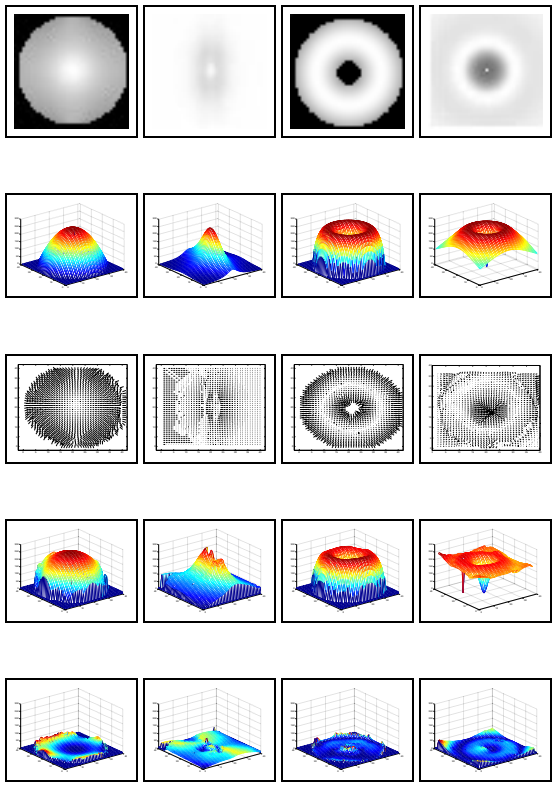

Figure 1: *Results on synthetic images.*

ϕ^T to get the matrix equation $\phi^T\Omega W\phi = \lambda\phi^T\Omega\phi$. As a result, it follows that

$$\lambda = \frac{\phi^T\Omega W\phi}{\phi^T\Omega\phi} \qquad (7)$$

This expression is reminiscent of the Rayleigh quotient. It also suggests the plausibility of using the mathematical techniques commonly employed to study the asymptotic behaviour of non-homogeneous Markov chains [10] to take our analysis further.

We note that the elements of the permutation π are required to be real. Consequently, the co-efficients of the eigenvector ϕ are always non-negative. Since the elements of the matrices Ω and W are also positive, it follows that the quantities $\phi^T\Omega W\phi$ and $\phi^T\Omega\phi$ are positive. Hence, the set of solutions reduces itself to those that are determined up to a constant $\lambda > 0$. As a result, the co-efficients of the eigenvector ϕ are linearly independent of the all-ones vector $\mathbf{e} = (1,1....,1)^T$.

With these observations in mind, we focus on proving the existence of a permutation that minimises $g(\pi)$ subject to the constraints in Equation 4, and demonstrating that this per-

mutation is unique. To this end we use the Perron-Frobenius theorem [11]. This concerns the proof of existence regarding the eigenvalue $\lambda_* = \max_{i=1,2,...,|V|}\{\lambda_i\}$ of a primitive, real, non-negative, symmetric matrix W, and the uniqueness of the corresponding eigenvector ϕ_*. The Perron-Frobenius theorem states that the eigenvalue $\lambda_* > 0$ has multiplicity one. Moreover, the co-efficients of the corresponding eigenvector ϕ_* are all positive and the eigenvector is unique. As a result the remaining eigenvectors of W have at least one negative co-efficient and one positive co-efficient. If W is substochastic, ϕ_* is also known to be linearly independent of the all-ones vector $\mathbf{e}$. As a result, the leading eigenvector of W is the minimiser of $g(\pi)$.

The elements of the leading eigenvector ϕ_* can be used to construct an integration path. As noted earlier the components of ϕ_* decrease with increasing curvature of the seriation path. We commence from the node associated with the largest component of ϕ_*. We then sort the elements of the leading eigenvector such that they are both in the decreasing magnitude order of the co-efficients of the eigenvector, and satisfy neighbourhood connectivity constraints on the pixel lattice. The procedure is a recursive one that procedes as follows. At each iteration, we maintain a list of sites visited. At iteration k let the list of sites be denoted by $\mathcal{L}_k$. Initially, $\mathcal{L}_0 = j_o$ where $j_0 = \arg\max_j \phi_*(j)$, i.e. j_0 is the component of ϕ_* with the largest magnitude. Next, we search through the set of 8-neighbours of j_o to find the pixel associated with the largest remaining component of ϕ_*. If $\mathcal{N}_{j_0}$ is the set of 8-neighbours of j_0, the second element in the list is $j_1 = \arg\max_{l\in\mathcal{N}_{j_0}} \phi_*(l)$. The pixel index j_1 is appended to the list of sites visited and the result is $\mathcal{L}_1$. In the kth (general) step of the algorithm we are at the pixel site indexed j_k and the list of sites visited by the path so far is $\mathcal{L}_k$. We search through those 8-neighbours of j_k that have not already been traversed by the path. The set of pixel sites is $C_k = \{l | l \in \mathcal{N}_{j_k} \wedge l \notin \mathcal{L}_k\}$. The next site to be appended to the path list is therefore $j_{k+1} = \arg\max_{l\in C_k} \phi_*(l)$. This process is repeated until no further moves can be made. This occurs when $C_k = \emptyset$ and we denote the index of the termination of the path by T. The integration path Γ_S is given by the list of pixel sites $\mathcal{L}_T$.

4. Extracting Patches

In practice the surface under study may have a patch structure. The patches may be identified by finding the blocks of the transition matrix induced under a permutation of the nodes. We commence by constructing the thresholded transition matrix A whose elements are defined as follows

$$A(i,j) = \begin{cases} 0 & \text{if } P(i,j) << 1 \\ P(i,j) & \text{otherwise} \end{cases} \tag{8}$$

Suppose that there are m distinct surface patches, each associated with an adjacency matrix $B^{(i)}$. If C represents a noise matrix, then the relationship between the observed transition matrix A and the underlying block-structured transition matrix is $A = B + C$ where $B = \mathcal{P}B_D\mathcal{P}^T$, $\mathcal{P}$ is a permutation matrix and $B_D = diag(B^{(1)}, B^{(2)}, \ldots, B^{(i)}...)$ is a block diagonal matrix in which $B^{(i)}$ is the sub-block corresponding to the patch indexed i. To recover the matrix B_D, we turn to the eigenvector expansion of the matrix A and write

$$A = \vec{b}_*\vec{b}_*^T + \sum_{i=2}^{|V|} \lambda_i \vec{b}_i \vec{b}_i^T \tag{9}$$

where the leading eigenvalue is unity i.e. $\lambda_1 = 1$, $\vec{b}_*$ is the leading eigenvector and the eigenvectors are normalised to be of unit length, i.e. $|\vec{b}_i| = 1$. To identify the patches, we use the following iterative procedure. We initialise the algorithm by letting $A^{(1)} = A$. Further suppose that $\vec{b}_*^{(1)}$ is the leading eigenvector of $A^{(1)}$. The matrix $B^{(1)} = \vec{b}_*^{(1)}\vec{b}_*^{(1)T}$ represents the first block of A. The nodes with non-zero entries belong to the patch. These nodes may be identified and removed from further consideration. To do this we compute the residual transition matrix $A^{(2)} = A^{(1)} - B^{(1)}$ in which the elements of the first patch are nulled. The leading eigenvector $\vec{b}_*^{(2)}$ of the residual transition matrix $A^{(2)}$ is used to compute the second block $B^{(2)} = \vec{b}_*^{(2)}\vec{b}_*^{(2)T}$. The process is repeated iteratively to identify all of the principal blocks of A. At iteration n, $\vec{b}_*^{(n)}$ is the leading eigenvector of the residual transition matrix $A^{(n)}$, and the n^{th} block is $B^{(n)} = \vec{b}_*^{(n)}\vec{b}_*^{(n)T}$. The index set of the patch indexed n is the set of nodes for which the components of the leading eigenvector $\vec{b}_*^{(n)}$ are non-zero. Hence, the index-set for the i^{th} patch is $S_i = \{k | \vec{b}_*^{(i)}(k) \neq 0\}$. It is important to stress that the patches are non-overlapping, i.e. the inner product of the block eigenvectors for different patches is zero $\vec{b}_*^{(k)}.\vec{b}_*^{(l)} = 0$, where $k \neq l$.

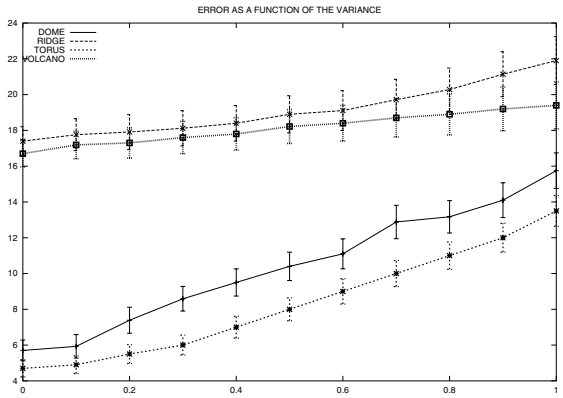

Figure 2: *Plot of the error percentage as a function of the variance for the four synthetic basic shapes.*

5. Height Recovery

Our surface height recovery algorithm proceeds along the sequence of pixel sites defined by the order of the co-efficients of the leading eigenvector associated with the separate patches. For the k^{th} patch, the path is $\Gamma_k = (j_k^1, j_k^2, j_k^3, ...)$. As we move from pixel-site to pixel-site defined by this path, we increment the surface height-function. In this section, we describe the trigonometry of the height incrementation process.

At step n of the algorithm, we make a transition from the pixel with path-index j_{n-1} to the pixel with path-index j_n. The distance between the pixel-centres associated with this transition is

$$d_n = \sqrt{(x_{j_n}^2 - x_{j_{n-1}})^2 + (y_{j_n} - y_{j_{n-1}})^2} \qquad (10)$$

This distance, together with the surface normals $\vec{N}_{j_n} = (N_{j_n}(x), N_{j_n}(y), N_{j_n}(z))$ and $\vec{N}_{j_{n-1}} = (N_{j_{n-1}}(x), N_{j_{n-1}}(y), N_{j_{n-1}}(z))$ at the two pixel-sites may be used to compute the change in surface height associated with the transition. The height increment is given by

$$h_n = \frac{d_n}{2}\left\{\frac{N_{j_n}(x)}{N_{j_n}(y)} + \frac{N_{j_{n-1}}(x)}{N_{j_{n-1}}(y)}\right\} \qquad (11)$$

If the height-function is initialised by setting $z_{j_0} = 0$, then the centre-height for the pixel with path-index j_n is $z_{j_{n+1}} = z_{j_n} + h_n$.

Once the surface normals that belong to the individual patches have been integrated together, then we merge them together to form a global surface. Suppose that S_k is the integrated surface for the k^{th} patch. We compute the mean height for the pixels belonging to this boundary. We merge the patches together by ensuring that abutting patches have the same mean boundary height.

6. Region Quadric Fitting

Once the height values are availble for each pixel site in a patch, then we perform smoothing. We do this by fitting a local quadric to the height data for the patch sites. To do

this we employ a simple least-squares fitting method. Our aim is to adjust the surface normal directions so that they are consistent with the gradient of the fitted quadric and also remain on their respective irradiance cone. The opening angle of the corresponding cone will be determined by the gray-level E_i at pixel-site indexed i and the direction of the surface normal on the image plane will be determined by the gradient of the fitted quadric. As a result, we can parameterise the surface normal directions at the kth iteration of the recovery process using two angles. The first of these is the angle $\theta_i^{(k)}$ between the surface normal and the light source direction. This is simply the opening angle of the irradiance cone, and this angle must be kept fixed in order to ensure that the recovered surface normal satisfies Lambert's law. The second

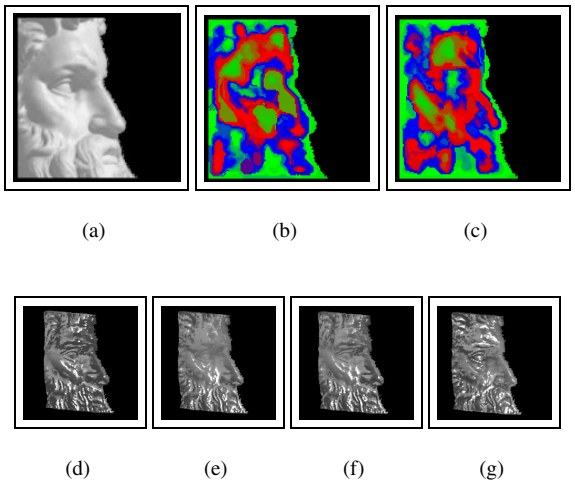

Figure 3: *Results of the algorithm on the image of Michelangelo's Moses*

is the azimuthal angle $\varphi_i^{(k)}$, which measures the position of the surface normal on the irradiance cone.

The angles $\varphi_i^{(k)}$ and $\theta_i^{(k)}$ can be defined in terms of the gradient of the local quadric patch $Q_p^{(k)}(x_i, y_i)$ fitted at iteration k projected onto the image plane. Using some simple geometry, we can write

$$\begin{aligned}
\varphi_i^{(k)} &= \begin{cases} 2\pi - \arccos\dfrac{\frac{\partial Q_p^{(k)}(x_i,y_i)}{\partial x_i}}{R_i^{(k)}} & \text{if } \frac{\partial Q_p^{(k)}(x_i,y_i)}{\partial y_i} < 0 \\ \arccos\dfrac{\frac{\partial Q_p^{(k)}(x_i,y_i)}{\partial x_i}}{R_i^{(n)}} & \text{otherwise} \end{cases} \\
\theta_i^{(k)} &= \arccos(E_i)
\end{aligned} \qquad (12)$$

where $R_i^{(k)}$ is the magnitude of the component of the gradient on the image plane, which is given by

$$R_i^{(k)} = \sqrt{\left(\frac{\partial Q_p^{(k)}(x_i,y_i)}{\partial x_i}\right)^2 + \left(\frac{\partial Q_p^{(k)}(x_i,y_i)}{\partial y_i}\right)^2} \qquad (13)$$

Once the azimuth and the tilt angles have been computed, then the surface normals may be obtained using the equations

$$\begin{aligned}
N_i(x) &= \begin{cases} \sin(\theta_i^{(k)})\cos(\frac{\varphi_i^{(k)} + \varphi_i^{(k-1)}}{2}) & \text{if } k > 1 \\ \sin(\theta_i^{(k)})\cos(\varphi_i^{(k)}) & \text{otherwise} \end{cases} \\
N_i(y) &= \begin{cases} \sin(\theta_i^{(n)})\sin(\frac{\varphi_i^{(n)} + \varphi_i^{(n-1)}}{2}) & \text{if } n > 1 \\ \sin(\theta_i^{(n)})\sin(\varphi_i^{(n)}) & \text{otherwise} \end{cases} \\
N_i(y) &= \cos(\theta_i^{(n)})
\end{aligned} \qquad (14)$$

7. Algorithm Description

In this section, we summarise a shape-from-shading algorithm based on the height recovery method described above. The sequence of processing steps is as follows:

- Step 0: The surface normals are placed in their initial positions on the irradiance cones. To do this we align them in the directions of the image gradient. The gradient is computed by first smoothing the grey-scale image by fitting local quadric patches to the raw image intensity. The smoothed image gradient is found from the derivatives of the fitted patch.
- Step 1: From the initial field of surface normals, we compute the sectional curvatures and hence the transition matrix. The blocks of the matrix are surface patches. The leading eigenvector of the transition matrix for each block is used to compute the integration path. Using the patch integration paths, we recover estimates of the surface height.
- Step 2: For the sites in each patch, we fit a quadric patch to the available height estimates. The fitted surface patches are used to compute an estimate of the surface gradient. At each location, the gradient estimate is used to adjust the position of the surface normals on their respective irradiance cones.

Steps 1 and 2 are iterated until a stable set of surface height estimates are located.

8. Experiments

Our experiments with the new graph-spectral method for shape-form-shading are divided into two parts. We commence with a study based on synthetic data which is aimed at establishing the noise sensitivity and failure modes of the method. In the second part of the study we focus on the behaviour of the method when confronted with real world data. Here we study images of classical statues.

8.1. Synthetic Data

In this section we provide some experiments on synthetic data. The aim here is to determine the accuracy of the surface reconstruction method. To this end we have generated synthetic surface height data. From the surfaces, we have computed the field of surface normal directions. A light source direction is then selected and the surfaces have been rendered using the Lambertian reflectance process outlined in Section 2. We have then applied the graph-spectral shape-from-shading method to the resulting synthetic images. We compare the resulting height estimates with the height data for the original surfaces.

In Figure 1 we show the results obtained for a series of different surfaces. In the first, second and third rows we show the Lambertian shading, height data and surface normals for the synthetic surfaces. In the fourth row of the figure we show the surface reconstructed by applying our shape-form-shading method to the images in the top row. The bottom row of the figure shows the absolute error between the ground-truth and reconstructed surface height. From left-to-right the surfaces studied are a dome, a sharp ridge, a torus and a volcano. In all four cases the surface reconstructions are qualitatively good. For the dome the height errors are greater at the edges of the surface where the slope is largest. In the case of the ridge, there are errors at the crest. For the volcano, there are some problems with the recovery of the correct depth of the "caldera", i.e. the depression in the centre. For the reconstructed surfaces, the mean-squared errors are 5.6% for the dome, 10.8% for the ridge, 7.8% for the torus and 4.7% for the volcano. Hence, the method seems to have greater difficulty for surfaces containing sharp creases.

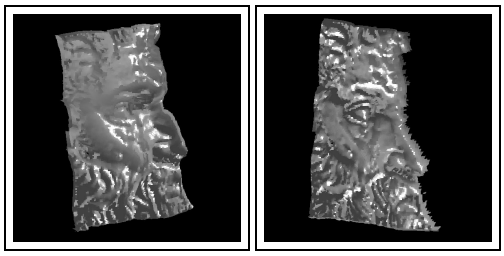

Figure 4: *Views of the reconstructed surface of Michelangelo's Moses*

We have repeated these experiments under conditions of controlled noise. To do this we have added random measurement errors to the raw image brightness. The measurement errors have been sampled from a Gaussian distribution with zero mean and known variance.

To investigate the effect of noise further, in Figure 2 we plot the mean-squared error for the reconstructed surface height as a function of the standard deviation of the added Gaussian noise. The different curves are for the different surfaces shown in Figures 1. It is clear that the mean-squared error grows slowly with increasing noise standard deviation. The torus and the volcano give the poorest errors, while the ridge and the dome give the smallest errors. This is s reflection of the fact that the torus and the volcano are the more structured surfaces.

8.2. Real World Data

We have experimented with a variety of real world images of predominantly Lambertian objects with small local specular highlights and albedo variations. In principle we can overcome both of these problems. In a recent paper, we have described a probabilistic method for specularity removal which uses the Torrance and Sparrow model to perform Lambertian reflectance correction for shiny objects [12]. Local albedo

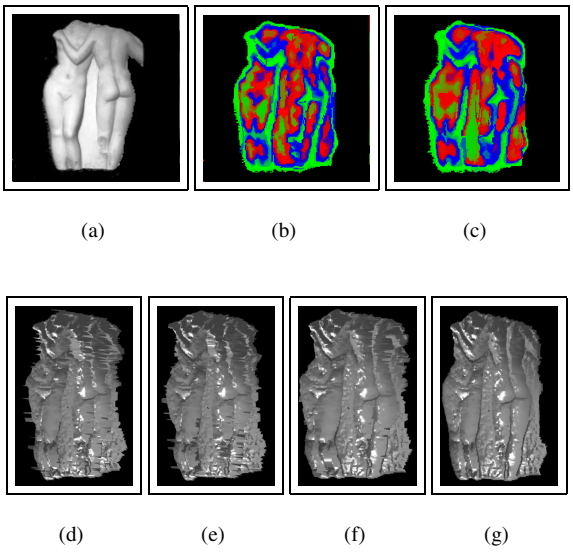

Figure 5: *Results of the algorithm on the image of the Three Graces Relief fragment*

changes can be accommodated using brightness normalisation or histogram equalisation. The objects studied are a detail of Michaelangelo's Moses, a fragment of "The Three Graces" relief and an image of a bust of Beethoven from the University of Central Florida shape-from-shading data-base [6].

In Figure 3 we show our first sequence of results. The panels of the figure are organised as follows. In Figure 3a we show the original image used as input to the shape-from-shading process. This is a side view of the head of the statue "Moses". Figures 3b and c show the arrangements of quadric patches after one and four iterations of the algorithm. In these images, the different quadric patches are coded in different colours. Finally, Figures 3 d to g show the reconstructed surface after successive iterations of the algorithm. Initially, the set of surface patches is fragmented, lack coherence and do not reproduce the surface detail well. However, after four iterations the reconstructed surface is more continuous and the fine detail of the object is well reproduced.

In Figure 4 we show the reconstructed surface viewed from two different directions. There are a number of features that are worth noting from the panels of the figure. First, the organisation of the surface normals and the arrangement of patches both improve as the algorithm iterates. Second, from the different surface views it is clear that the surface structure is well reconstructed. For instance the shape of the nose, particularly in the proximity of the nostrils, is well reproduced. Moreover, the fine structure of the beard, the detail in the eye-sockets and the shape of the cheek bones are all well reconstructed.

In Figure 5 we repeat the sequence of panels and in Figure 6 we show different views of the reconstructed surface for a section of the relief "The Three Graces". The same iterative improvement in the quality of the surface normals and the patch arrangement is clear. In this case the algorithm converged after 3 iterations. Morover, the different views of the surface reveal that the detail of the relief is well reproduced. The legs, buttocks and indentation in the back of the left-hand figure are all well reconstructed.

Finally, Figures 7 and 8 show analogous results for an image of a bust of Beethoven. Here the path structure is particularly clear and corresponds well to the topographic structure of the surface. For instance, the eye sockets correspond to distinct patches. Both the patch structure and the structure of the needle maps are improved as the algorithm iterates. Initially, little of the surface structure is evident. However, after the algorithm has converged the structure of the hair and the boundary of the cheeks have become well defined. These features are all clear in the different views of the reconstructed surface.

9. Conclusions

In this paper, we have described a graph-spectral algorithm for shape-from-shading. We constrain the surface normals at each image location to fall on an irradiance cone whose axis is the light source direction and whose apex angle is determined by the measured image brightness. The method uses the leading eigenvector of the adjacency matrix to identify a curvature minimising path for surface integration and perform height recovery. By fitting quadric surfaces to the height data, we perform surface smoothing. We update the surface normal directions by rotating them so that they point in the direction of the fitted surface gradient. The surface integration and surface normal adjustment steps are iterated until stable height estimates are recovered. The method proves effective for reconstructing surfaces from single views of 3D objects, and gives subjectively better results than a number of alternative shape-from-shading methods.

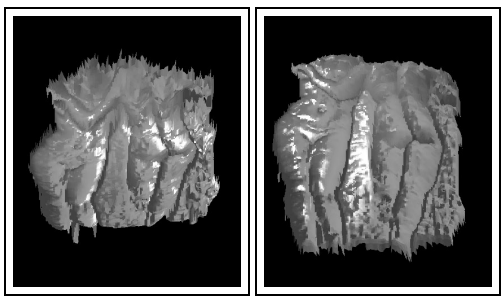

Figure 6: *Views of the surface of the Three Graces Relief fragment*

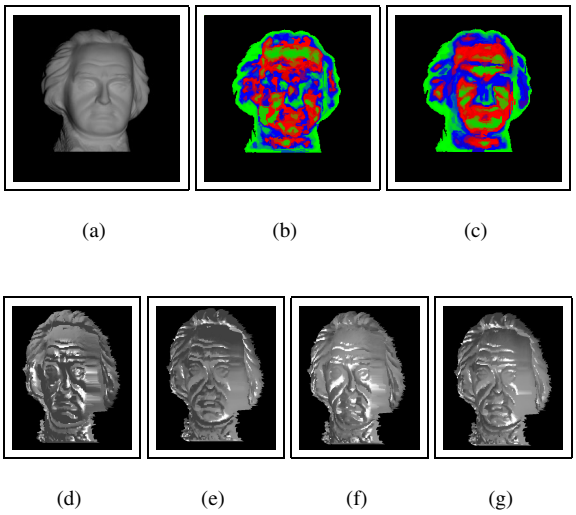

Figure 7: *Results of the algorithm on the image of the Beethoven bust*

References

1. J. Shi and J. Malik. Normalized cuts and image segmentations. In *Proc. IEEE CVPR*, pages 731–737, 1997.
2. L. S. Shapiro and J. M. Brady. A modal approach to feature-based correspondence. In *British Machine Vision Conference*, pages 78–85, 1991.
3. W. Dickson. Feature grouping in a hierarchical probabilistic network. *Image and Vision Computing*, 9(1):51–57, 1991.
4. J. E. Atkins, E. G. Roman, and B. Hendrickson. A spectral algorithm for seriation and the consecutive ones problem. *SIAM Journal on Computing*, 28(1):297–310, 1998.
5. B. K. P. Horn and M. J. Brooks. The variational approach to shape from shading. *CVGIP*, 33(2):174–208, 1986.
6. R. Zhang, Ping-Sing Tsai, J. E. Cryer, and M. Shah. Shape from shading: A survery. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(8):690–706, 1999.
7. P. Dupuis and J. Oliensis. Direct method for reconstructing shape from shading. In *CVPR92*, pages 453–458, 1992.
8. M. Bichsel and A. P. Pentland. A simple algorithm for shape from shading. In *CVPR92*, pages 459–465, 1992.
9. P. L. Worthington and E. R. Hancock. New constraints on data-closeness and needle map consistency for shape-from-shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(12):1250–1267, 1999.
10. P. Bremaud. *Markov Chains, Gibbs Fields, Monte Carlo Simulation and Queues*. Springer, 2001.
11. R. S. Varga. *Matrix Iterative Analysis*. Springer, second edition, 2000.
12. H. Ragheb and E. R. Hancock. Separating lambertian and specular reflectance components using iterated conditional modes. In *Proc. of the British Machine Vision Conference*, pages 541–552, 2001.

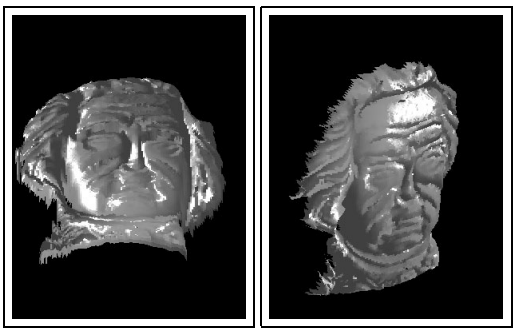

Figure 8: *Views of the surface of the Beethoven bust*