Knowledge-Based Shape from Shading [*]

Nick Barnes and Zhi-Qiang Liu
Computer Vision and Machine Intelligence Lab (CVMIL)
Department of Computer Science
The University of Melbourne,
221 Bouverie Street, Carlton, Victoria, Australia, 3053
email: nmb@cs.mu.oz.au,
zliu@cs.mu.oz.au

In this paper, we study the problem of recovering the approximate three-dimensional shape of an object when knowledge about the object is available. The application of knowledge-based methods to image processing tasks will help overcome problems which arise from processing images using a pixel-based approach. We show that by applying domain specific knowledge, we can fully automate the derivation of the approximate shape of an object. Further, this approach can yield specific advantages over existing approaches, both in terms of computation and processing results. This is a powerful paradigm that will have applications in object recognition, robotic navigation, and domain specific scene understanding.

Keywords: Shape-from-shading, knowledge-based system, image processing, object recognition, frames, knowledge representation, mobile robot navigation, computer vision

1. Introduction

The classical computer vision approach[23] attempts to build a model of a three-dimensional scene from a single image. This process is expected to de-

[*] This work is supported in part by an Australian Research Council (ARC) large grant.

rive a description without any prior knowledge of what appears in the scene. However, the importance of applying domain knowledge to artificial intelligence (AI) programs has been a fundamental observation arising from research in AI.[8] This observation has not yet been extensively applied in image processing. Most methods act only on local regions of pixels; however, even the entire image does not carry sufficient contextual information. Image processing methods still regularly fail to extract features which are obvious to the human eye. This paper presents a new and powerful paradigm for image processing based tasks: a knowledge-based approach. We demonstrate this method for the classic shape-from-shading problem, showing that it can give advantages both in terms of what problems can be solved, and of computational time.

Most shape-from-shading research takes the classical computer vision approach, see Ref. 16 for examples. However, it is well known that the shape-from-shading problem is under-determined and difficult to solve. Certainly, simplifications are necessary, for instance, the light source and observer are assumed to be far from the object, and the underlying surface is assumed to be lambertian. All available methods impose some constraints. For example, Brooks and Horn[3] require that the underlying surface be smooth, albedo be uniform across the scene, and the occluding boundaries be known with the surface smoothly curving at that point (a limb boundary). Variational methods, such as this, require *boundary conditions*, which are points where the surface normal can be determined, in order to find a unique solution, if at all possible. The assumption of an occluding limb boundary is common also to Refs. 4, 18, 21, 22 and others. Many variational methods also use singular points, which are points where the brightness is maximal, as boundary conditions. Singular points indicate that the surface normal is in the direction of the light source.

Ikeuchi and Horn[18] show that in some cases, self-shadow boundaries can be used as singular points. Horn[14] derives a non-variational method which assumes the existence of singular points. Local methods have to make other assumptions, and do not generally produce exact estimates of shape. Pentland[26], and Lee and Rosenfeld[20] assume that the surface is locally spherical. Ferrie and Levine[7] make less strict assumptions, but still require local constraints on the surface geometry. Pentland[27] takes a linear approximation to the reflectance model which allows a closed form solution. However, this approximation is only correct when the variation of surface normals in a region is small, or when the light source is oblique to the viewer. The linear approximation proposed in Ref. 30 does not handle noise well.

To solve the shape-from-shading problem, Horn[15] suggests that it is necessary that the surface reflectance be locally uniform. However, the aim of reconstructing the shape of an arbitrary scene with arbitrary lighting purely from image shading is probably not achievable even given this. In this paper, we apply shape-from-shading to object recognition, with a particular aim for robot visual guidance around man-made objects. Existing variational methods are inadequate as man-made objects may consist entirely of planar faces, and thus have no occluding limb boundaries, and are unlikely in general to have points of singular brightness. The linear surface reflectance assumption is problematic for robot vision because the light-source may be mounted on the robot, making the light source and viewing direction close, and noise is often a problem with real cameras. Local methods require assumptions about local surface geometry. We cannot tell, *apriori*, what the local surface shape will be, and we would like not to be restricted to handling objects with a particular local surface geometry. Finally, the traditional three-dimensional reconstruction methods discussed above generally implicitly rely on user intervention. Variational methods require boundary conditions to be set manually, while local methods require the selection of an appropriate model for local shape.

Recent research has seen the application of generic shape primitives[2] to find quantitative and qualitative shape.[6] However, this is only applicable to objects composed of distinct volumetric parts with no fine structural detail.[6] Also, purposive motion has been applied for global surface reconstruction.[19] However, this work is unable to reconstruct the entire surface of objects with concavities.

We wish to use a shape-from-shading method in conjunction with edge-based matching in a model-based object recognition system. In deriving the surface shape of an object, the system may have access to some prior knowledge from the model. We apply shape from shading after edge-based matching, so the object is pre-segmented into surfaces along orientation discontinuities and there are no discontinuities within the surfaces. This would often be the case if we use object recognition for robot navigation such as in Ref. 1. Here, the task is for a robot to move to a particular location relative to an object. To do this, it recognises an object view and determines its relative pose and position based on object to model correspondences. This data is used to plan the robot's path around the object.

We assume that our model provides the angles between surfaces along the discontinuities, and the reflectance function of the object. In this paper, we describe a general formulation, but particularly examine the case where surfaces are approximately lambertian, with approximately constant albedo. Other papers discuss the use of different lighting and reflectance models .[17,25,29]

In order to apply knowledge to shape from shading, we employ a knowledge-based architecture based on frame constructs.[24] Note, however, that this is not a traditional application of frames, shape-from-shading operates on numeric rather than symbolic data. Our knowledge here is a mixture of symbolic and numeric data.

In the following sections, we describe a knowledge-based shape-from-shading technique, where knowledge from an object model and edge matching is applied to assist solving the shape-from-shading problem. In many domains where shape-from-shading could be used on man-made and natural objects, we may have knowledge about object structure. We present results for simulated images with noise and real camera images. We also consider human faces, demonstrating that the use of domain specific knowledge can reduce computation time.

2. Background

This paper uses materials from two distinct fields: shape-from-shading; and knowledge-based architectures.

2.1. Shape from Shading

Brooks and Horn[3] derived a method for finding surface shape and light source direction for a smooth, lambertian surface where boundary conditions are set. They use a global scheme based on the image irradiance equation with a smoothing component. The surface is found iteratively by minimising the error in the surface fit for a functional composed of the image irradiance equation, and a regularising term.

$$I(x,y) = \int\int_{\Omega} ((E - \mathbf{n}\cdot\mathbf{s})^2 + \lambda(\parallel \mathbf{n}_x \parallel^2 + \parallel \mathbf{n}_y \parallel^2) + \mu(x,y)(\parallel \mathbf{n} \parallel^2 - 1))dxdy, \tag{1}$$

where E is the image irradiance, $\mu(x, y)$ is a lagrange multiplier function to impose the constraint that $\mathbf{n}$(x,y) is a unit vector, $\mathbf{s}$ points to the light source, $\mathbf{n}_x$ and $\mathbf{n}_y$ are the partial derivatives of n, and λ rates the relative importance of the regularisation term. Minimising I is solved as a problem of variational calculus. They derive an iterative solution using the Euler equations.

Malik and Maydan[22] extend Brooks and Horn's technique, combining shape from shading with three-dimensional line drawing interpretation to find the shape from a single image with piecewise smooth surfaces. They assume that the image has been pre-segmented, and that a sparse labelling of the image is available. The sparse labelling is used to constrain the possible shape of the object. At a labelled vertex, with three or more visible surfaces, they are typically able to constrain the surface normal at the points around the vertex to a unique value.

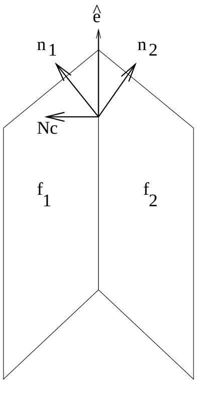


Figure 1: The tangent vector, and surface normals at a non-occluding edge.

They extend Brooks and Horn's iterative scheme to propagate the shape across discontinuities. Let $\hat{\mathbf{e}}$ be the unit tangent vector to the edge at a point on the roof, and let $\mathbf{n}_1$ and $\mathbf{n}_2$ be the unit surface normals to the tangent planes to the two faces f_1 and f_2 at that point, respectively. Let $\hat{\mathbf{e}}$ be oriented such that when one walks on the edge in the direction of $\hat{\mathbf{e}}$, the face f_1 is to the left (see Fig. 1). Now $\hat{\mathbf{e}}$ is perpendicular to both $\mathbf{n}_1$ and $\mathbf{n}_2$, and thus parallel to

$\mathbf{n}_1 \times \mathbf{n}_2$, provided that the variation in depth of the edge is small compared to the distance between the surface and the camera. Also, let $\hat{\mathbf{e}}_{proj}$ be the unit vector of the orthographic projection of $\hat{\mathbf{e}}$ onto the image plane.

If we let $\mathbf{N}_c$ be a unit vector parallel to the image plane and perpendicular to $\hat{\mathbf{e}}_{proj}$, then:

$$(\mathbf{n}_1 \times \mathbf{n}_2)_{proj} \cdot \mathbf{N}_c = 0 \tag{2}$$

We have four unknowns for the unit normals $\mathbf{n}_1$ and $\mathbf{n}_2$, of the two faces, and have only three equations: Eq. (2), and the image irradiance equation for the points on both surfaces. Thus, a unique solution is not possible. However, Malik and Maydan add a smoothness constraint and form a composite functional which allows shape to propagate across orientation discontinuities. Here the equation is formed for a general reflectance map $\mathbf{R}$ and is taken over the surface s:

$$\begin{aligned} I(x,y) = \int_s [\lambda_1[(E_1 - R(\mathbf{n_1}))^2 + ((E_2 - R(\mathbf{n_2}))^2] + \ \lambda_2[(\mathbf{n}_1 \times \mathbf{n}_2) \cdot N_c]^2 \\ + \lambda_3(\| \mathbf{n}_1' \|^2 + \| \mathbf{n}_2' \|^2) + \ \mu_1(\| \mathbf{n}_1 \|^2 - 1) + \ \mu_2(\| \mathbf{n}_2 \|^2 - 1)]ds \end{aligned} \tag{3}$$

Again, the Euler equation of the composite functional is used to derive an iterative scheme for estimating the surface normal along a non-occluding edge.

The scheme operates by first constraining the surface normals at limbs, or vertexes of three or more surfaces where all surfaces showing. With these boundary conditions the scheme uses the method of Ref. 3 to propagate shape across surfaces, and the iterative method formed from Eq. (3) to propagate shape across orientation discontinuities.

Both these techniques, and other variational schemes require limb occluding boundaries, or singular points to provide boundary conditions and hence to find a unique solution. For smoothly curving surfaces, it is reasonable to assume that the use of these constraints will provide adequate boundary value conditions to calculate unique solutions for a large proportion of images. However, man-made objects often consist largely of flat surfaces. In such cases there may not be a singular point where the surface normal is parallel to the light source. Also, there may not be any limbs: all occluding edges may occur at orientation discontinuities. Further, the scheme of Malik and Maydan[22] relies on an *apriori* sparse labelled edge diagram being derived from edges extracted from a raw image. Edge labelling is not reliable with edges extracted from noisy image data. Further, they suggest no method for representing this knowledge.

2.2. Knowledge Based Representation of Objects

Generally, knowledge-based approaches emphasise the role of purely symbolic inferencing for reasoning and decision making.[9] However, much of commonsense knowledge about problems is in the form of procedures or processes for achieving particular goals.[10] One important distinction between knowledge-based systems that reason about processes and those that reason about facts, is that processes may succeed or fail. For systems which interact in some way with an environment success or failure cannot be predicted. The result depends on the environment. It is often not adequate to take the standard logic programming approach of negation-by-failure, as failure of a process may be due to lack of evidence. Also, shape from shading is a numerical process, which in our case consists of a number of independent processes. As such it requires a somewhat different approach from the use of a simple inference engine.[5]

To tackle the problem of shape from shading, we need to represent knowledge about the visual appearance of the object as well as some knowledge about properties of the object itself. Also, we need some correspondences between the geometric structures of the object and perceived structures in the image. Further known facts are light source direction and knowledge derived from the image. This must be combined with rules which govern the processing in the system, and the numerical processes which collectively form a shape solution for the object in the image. In specific domains we may also have knowledge about types of objects being examined.

Frames[24] allow packaging of declarative knowledge with procedural knowledge. Also, they allow the ability to group facts, rules and processes into associative clusters.[11] These features are useful for problems in which much of the knowledge is in the form of procedural know-how or numerical processes, and where the facts, rules and processes can be partitioned in a way that is useful for the problem domain.

A *frame* is a data structure for representing a stereotyped situation. Information includes how to use the frame, what to expect next, and what to do if the expected does not occur. This forms a network of nodes and relations. Upper levels are fixed, representing things that are always true for a possible scenario. Lower levels consist of terminals - or slots - to be filled by instances of specific data. The terminals can specify conditions which assignments must meet, and these assignments are usually smaller subframes. Once a frame is proposed for a situation, a matching process attempts to assign values to the terminals. Matching is controlled both by the frame, and the current system goals.

A frame system is a collection of related frames linked together. Transformations between frames relate to important transitions in what is being represented. Links may be formed so that if a proposed frame cannot make correct terminal assignments it can suggest other frames which may be appropriate for the given situation.

Frame terminals are normally filled with default assignments. These may be used for representing most likely situations and general information. Defaults can be displaced by new items that better fit the current situation.

Requicha[28] discusses computer-based, symbolic representation of three-dimensional solids. Boundary representations are used in this paper, because much of the perceptual data for recognising objects occurs at boundaries and on surfaces. In boundary representations, objects are represented as collections of "faces" or "patches", which are described by bounding edges and vertices. Planar faces can be entirely represented by edge information, while non-planar surfaces require representation of the surface shape. In typical boundary schemes the concept of a face may not correspond to what appears perceptually to be a face. Our purposes are better served by representations that correspond to the object as perceived.

3. Using Object Model Knowledge for Shape From Shading

In model-based computer vision, it is reasonable to assume that the object can be modelled as piecewise smooth, i.e. there are no discontinuities within faces, although there may be discontinuities between faces. Many man-made objects are composed of smooth faces separated by discontinuities. We consider a situation where edges of the image have already been hypothetically matched to the model, and we know the basic model.

This particular scenario is one possible application of a knowledge-based approach to image processing tasks, which illustrates the efficacy of the approach. The object model knowledge we apply consists of the following aspects:

- A wire frame model of the object;

- The direction of faces of the object (what is inside and outside the object);
- The angle between adjoining object faces;
- A hypothesis as to which faces are visible in the image; and
- The correspondence between the edges in the model and those in the actual image.

4. Shape from Shading

Our mathematical approach is based on that of Brooks and Horn,[3] and expands the ideas of Malik and Maydan[22] of using a segmented image. The method proposed in this paper is able to find a unique solution whenever part of an orientation discontinuity between two adjoining surfaces and the surfaces near the edge are visible. It also handles the limb and the singular point case. This is achieved by deriving additional constraints from the knowledge we have. In addition, this method is also able to find a unique solution for the cases described in Refs. 3, 22 without user intervention.

4.1. Constraints

If the angle between two surfaces along an edge is ϕ, then at a pair of pixels along an edge common to the two surfaces, the surface normals will be at an angle of $\theta = \phi - \pi$. We denote the surface normal on the first surface as $\mathbf{n}_1$ and the surface normal on the second surface as $\mathbf{n}_2$. Now from vector geometry[13] we have:

$$\mathbf{n}_1 \cdot \mathbf{n}_2 = \parallel \mathbf{n_1} \parallel \cdot \parallel \mathbf{n_2} \parallel \cdot \cos\theta. \tag{4}$$

This can be simplified for this problem as we restrict ourselves to unit vectors.

$$\mathbf{n}_1 \cdot \mathbf{n}_2 = \cos\theta. \tag{5}$$

This provides an additional constraint on the orientation of the surface normals for the pixels along an edge.

We combine this equation with those used by Malik and Maydan: the image irradiance equations for each pixel; and the triple product of the normals and the direction of the edge. We now have four equations and four unknowns, and so can find a solution. We are able to prune most ambiguous solutions because we know if each edge is concave or convex, and that surface normals for all visible points must be in the direction of the camera. Combining pruning with knowledge-based initialisation of surface normals generally leads to correct solutions. Direct iterative solutions such as Newton's method are too sensitive to noise, so we use variation. This method is similar to that in Ref. 3, described earlier.

The image irradiance equations for the two surfaces are:

$$E_1 - R_1(\mathbf{n}_1) = 0, E_2 - R_2(\mathbf{n}_2) = 0, \tag{6}$$

where E_1 and E_2 are the image intensities of pixels, and R_1 and R_2 are the reflectance maps, of the first and second surfaces respectively.

We can safely assume the edges are smooth, as any discontinuities should be modelled. As smoothing is not required to produce a unique solution, but only to suppress noise, we may smooth the resulting surface normals rather than adding a smoothing component to the variation. We also add a constraint that the surface normals are unit normals.

$$I(x,y) = \int\int_\Omega [(E_1 - R_1(\mathbf{n}_1))^2 + \ (E_2 - R_2(\mathbf{n}_2))^2 + \ \lambda_1(\mathbf{n}_1 \cdot \mathbf{n}_2 - \cos\theta)^2$$

$$+\lambda_2(\mathbf{n}_1\mathbf{n}_2\mathbf{Nc})^2 + \ \mu_1(x,y)(\| \ \mathbf{n}_1 \ \|) + \ \mu_2(x,y)(\| \ \mathbf{n}_2 \ \|)]dxdy. \qquad (7)$$

To solve Eq. (7) we form two components, and update $\mathbf{n}_1$ and $\mathbf{n}_1$ in turn.

$$f(E_1, \mathbf{n}_1, \mathbf{n}_2, \theta, \mathbf{Nc}, \lambda_1, \lambda_2) = \int\int_\Omega [(E_1 - R_1(\mathbf{n}_1))^2 \ + (E_2 - R_2(\mathbf{n}_2))^2 + \lambda_1(\mathbf{n}_1 \cdot \mathbf{n}_2 - \cos\theta)^2 + \ \lambda_2(\mathbf{n}_1\mathbf{n}_2\mathbf{Nc})^2 - \mu_1(x,y)(\| \ \mathbf{n}_1 \ \|)]dxdy, \qquad (8)$$

$$f(E_1, \mathbf{n}_1, \mathbf{n}_2, \theta, \mathbf{Nc}, \lambda_1, \lambda_2) = \int\int_\Omega [(E_1 - R_1(\mathbf{n}_1))^2 \ + (E_2 - R_2(\mathbf{n}_2))^2 + \lambda_1(\mathbf{n}_1 \cdot \mathbf{n}_2 - \cos\theta)^2 + \ \lambda_2(\mathbf{n}_1\mathbf{n}_2\mathbf{Nc})^2 - \mu_2(x,y)(\| \ \mathbf{n}_2 \ \|)]dxdy. \qquad (9)$$

(8) and (9) can be solved by taking the Euler equations. The functional

$$\int\int_\Omega F(x,y,\mathbf{n})dxdy \qquad (10)$$

has the Euler equation

$$\mathbf{F_n} = 0. \qquad (11)$$

Thus, assuming the reflectance map is differentiable, the Euler equations for

(8)and (9) are:

$$2(E_1 - R_1(\mathbf{n}_1))R_1'(\mathbf{n}_1) + 2(E_2 - R_2(\mathbf{n}_2))R_2'(\mathbf{n}_2)$$

$$+\lambda_1(\mathbf{n}_2(\mathbf{n}_1 \cdot \mathbf{n}_2 - \cos(\theta)) + \ \lambda_2(\mathbf{n}_1\mathbf{n}_2\mathbf{Nc})\mathbf{n}_{2_{i,j}} \times \mathbf{Nc} - \mu_1\mathbf{n}_1 = \mathbf{0}, \qquad (12)$$

$$2(E_1 - R_1(\mathbf{n}_1))R_1'(\mathbf{n}_1) + 2(E_2 - R_2(\mathbf{n}_2))R_2'(\mathbf{n}_2)$$

$$+\lambda_1(\mathbf{n}_1(\mathbf{n}_1 \cdot \mathbf{n}_2 - \cos(\theta)) + \ \lambda_2(\mathbf{n}_1\mathbf{n}_2\mathbf{Nc})\mathbf{n}_{1_{i,j}} \times \mathbf{Nc} - \mu_2\mathbf{n}_2 = \mathbf{0}, \qquad (13)$$

where $R_1'(\mathbf{n}_1), R_2'(\mathbf{n}_1)$ are derivatives with respect to $\mathbf{n}_1$, and $R_1'(\mathbf{n}_2)$, $R_2'(\mathbf{n}_2)$ are derivatives with respect to $\mathbf{n}_2$.

We may divide both by 2 and incorporate the factor into λ_1, λ_2 and the lagrange multipliers, and evaluate (12) at pixels using the discrete form:

$$(E_{1_{i,j}} - R_1(\mathbf{n}_{1_{i,j}}))R_1'(\mathbf{n}_{1_{i,j}}) \ + (E_{2_{i,j}} - R_2(\mathbf{n}_{2_{i,j}}))R_2(\mathbf{n}_{2_{i,j}})$$

$$+ \lambda_1\mathbf{n}_{2_{i,j}}(\mathbf{n}_{1_{i,j}} \cdot \mathbf{n}_{2_{i,j}} - \cos(\theta)) + \ \lambda_2(\mathbf{n}_{1_{i,j}}\mathbf{n}_{2_{i,j}}\mathbf{Nc})(\mathbf{n}_{2_{i,j}} \times \mathbf{Nc})$$

$$-\mu_{1_{i,j}}\mathbf{n}_{1_{i,j}} = \mathbf{0} \qquad (14)$$

We may rearrange this to form an iterative scheme,

$$\mathbf{n}_{1_{i,j}}^{(k+1)} = \tfrac{1}{\mu_{1_{i,j}}} \ [(E_{1_{i,j}} - R_1(\mathbf{n}_{1_{i,j}}^{(k)}))R_1'(\mathbf{n}_{1_{i,j}}^{(k)}) +$$

$$(E_{2_{i,j}} - R_2(\mathbf{n}_{2_{i,j}}^{(k)}))R_2'(\mathbf{n}_{2_{i,j}}^{(k)}) \ + \lambda_1\mathbf{n}_2^{(k)}(\mathbf{n}_1^{(k)} \cdot \mathbf{n}_2^{(k)} \ - \cos(\theta)) +$$

$$\lambda_2(\mathbf{n}_{2_{i,j}}^{(k)}\mathbf{n}_{1_{i,j}}^{(k)}\mathbf{Nc})(\mathbf{n}_{2_{i,j}}^{(k)} \times \mathbf{Nc})] \qquad (15)$$

Now, note that $\frac{1}{\mu_{1_{i,j}}}$ does not change the direction of the vector being computed. We may remove the Lagrange multiplier, and normalise the vector explicitly.

$$\begin{aligned}\mathbf{m}_{1_{i,j}}^{(k+1)} = & \ (E_{1_{i,j}} - R_1(\mathbf{n}_{1_{i,j}}^{(k)}))R_1'(\mathbf{n}_{1_{i,j}}^{(k)}) + (E_{2_{i,j}} - R_2(\mathbf{n}_{2_{i,j}}^{(k)}))R_2'(\mathbf{n}_{2_{i,j}}^{(k)}) \\ & +\lambda_1 \mathbf{n}_2^{(k)}(\mathbf{n}_1^{(k)} \cdot \mathbf{n}_2^{(k)} - \cos(\theta)) + \ \lambda_2(\mathbf{n}_{2_{i,j}}^{(k)}\mathbf{n}_{1_{i,j}}^{(k)}\mathbf{Nc})(\mathbf{n}_{2_{i,j}}^{(k)} \times \mathbf{Nc}),\end{aligned} \tag{16}$$

$$\mathbf{n}_{1_{i,j}}^{(k+1)} = \frac{\mathbf{m}_{1_{i,j}}^{(k+1)}}{\parallel \mathbf{m}_{1_{i,j}}^{(k+1)} \parallel}. \tag{17}$$

A similar derivation may be followed for $\mathbf{n}_{2_{i,j}}^{(k+1)}$.

In the case where we have a Lambertian reflectance model, we have:

$$R(\mathbf{n}) = \mathbf{n} \cdot s, R'(\mathbf{n}) = s. \tag{18}$$

Thus, Eq. (16) becomes:

$$\begin{aligned}\mathbf{m}_{1_{i,j}}^{(k+1)} = & \ (E_{1_{i,j}} - \mathbf{n}_{1_{i,j}}^{(k)} \cdot \mathbf{s})\mathbf{s} + (E_{2_{i,j}} - \mathbf{n}_{2_{i,j}}^{(k)} \cdot \mathbf{s})\mathbf{s} + \\ & \lambda_1 \mathbf{n}_2^{(k)}(\mathbf{n}_1^{(k)} \cdot \mathbf{n}_2^{(k)} - \cos(\theta)) + \ \lambda_2(\mathbf{n}_1^{(k)}\mathbf{n}_2^{(k)}\mathbf{Nc})(\mathbf{n}_{2_{i,j}}^{(k)} \times \mathbf{Nc}).\end{aligned} \tag{19}$$

Thus, we have boundary conditions, and so can find a unique solution in the following cases:

1. a discontinuity between two surfaces where the angle between the surfaces is known;
2. an occluding limb; and
3. a singular point.

In the case of occluding limbs, we may set the surface normal directly by calculating the normal to the tangent of the extracted edge. A singular point only has one possible solution to the irradiance equation, so the surface normal will be set directly. Note, we do not need to consider the three surface vertex case of Malik and Maydan, as this is a sub-case of (1).

Once we have determined boundary conditions we can propagate shape across surfaces, using the variational technique of Ref. 3.

4.2. Implementation

Noise in the image may distort shape results from the above solution. In a typical solution to the shape-from-shading problem the use of a mean filter will result in the distortion of discontinuities in the image. However, as we use a knowledge-based approach, and we have pre-processed the location of edges, we are able to safely perform filtering and regularisation on the image, provided we perform it separately for each edge and each surface. Edge preserving smoothing[12] would be another way of approaching this. However, edge-preserving smoothing techniques have known shortcomings, such as the loss of corners, and introducing artifacts in neighbourhood operations.

5. Knowledge-Based Implementation

This section describes a frame-based architecture which systematically applies a set of known facts, processes and rules to find a unique solution for the shape of an object. As a starting point for our method, we assume that segmentation and labelling of the image have been performed by a model-based edge matching process.

There are three main types of knowledge used in our system: facts, rules, and processes. The facts consist of correspondences between geometric structures of the object and perceived structures in the image, facts about properties of the object itself, and facts about the environment the image was taken in. There may also be domain specific facts about the type of object being viewed. The rules include knowledge of how to apply the facts and processes to solve the required problem, rules about how to proceed in cases of failure, and standard deductive rules to reach the goals of the system. The processes are the image processing tasks required to perform shape-from-shading, and the know-how of how these are performed, e.g. finding boundary conditions before attempting to solve surfaces.

5.1. Knowledge / Frame Topology

Our knowledge of an object is most naturally structured as it is perceived in the image, because the shape derivation process begins from a projected image. In object recognition, this type of representation is typically referred to as *view-centred*, where the object is described from different viewpoints. Our particular method is called *Canonical views.* Canonical-views and our reasons for choosing them are documented in Ref. 1. Briefly, canonical-views model a series of disjoint views of the object; the views are generic in the sense that a single view represents a finite proportion of the view space of the object. The views are two dimensional, as they would appear to the camera. Within a view we need to represent surfaces, edges, and information about each of these. By modelling each canonical-view using a frame, the transition of the viewer from one viewpoint to another can be modelled as the transformation between a frame representing each viewpoint. The viewpoint transition model is particularly useful for applications in vision for autonomous robots because viewer transitions correspond to *causal* moves by the robot. Viewpoint transitions are a method of indexing the views of an object, and the frames they correspond to, by the movement of the robot.

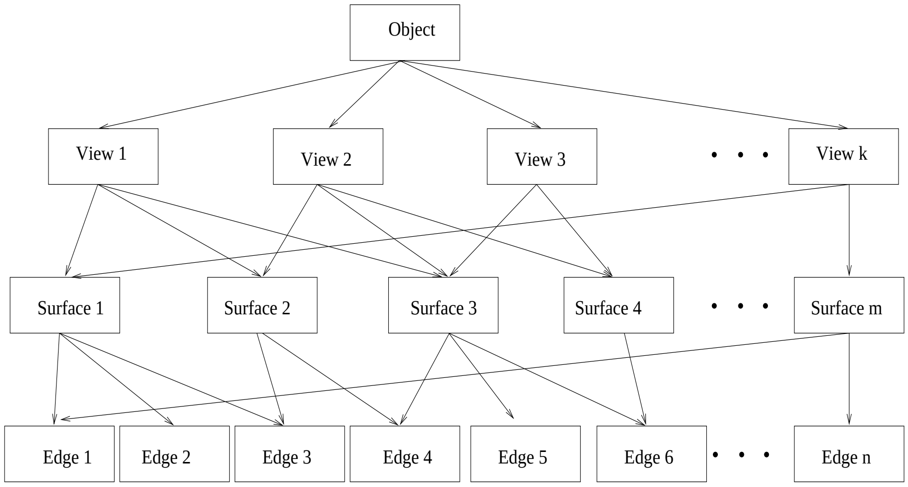


Figure 2: In our object representation, an object has several (possibly overlapping) views, each view has visible surfaces (overlapping views have common surfaces), and each surface will have some edges (surfaces will typically have some common edges).

Fig. 4 shows the object frames used in our system. Our frame system consists of Object, Surface, View, and Edge frames to represent objects. We also have a frame to handle environment information, in this case light source direction and albedo. This frame updates the lighting model every time the robot moves. In the case where the main light source is mounted on the robot this is a simple relation between the pan and tilt angle of the light source and camera. Complex lighting models could be represented by this structure to support the different models mentioned earlier. An intensity image frame abstracts the access to the image simplifying and safe-guarding the access required by the frames which perform the processing. Fig. 2 shows the hierarchy of representation in the object frame structures. An object frame represents a single object, and has terminals, or subframes for each of the represented views of the object. Each view frame has terminals, or subframes consisting of surfaces and edges. Fig. 3 shows an example of instantiated frames for a camera image of the guitar used in one of our experiments.

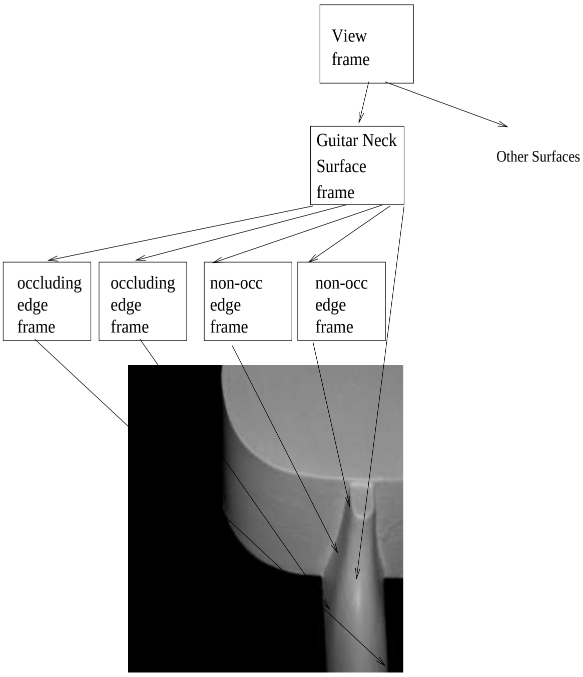


Figure 3: Instantiation of frames for the guitar image. There are five surfaces visible in this view. One of these is the guitar neck which has three occluding limb boundaries and two non-occluding edge boundary visible (other edges are modelled for this surface, but have no shape from shading calculations). The edge frames instantiate the edge pixel locations, and calculate shape at the corresponding image locations. The surface frame uses the instantiated edges to delimit the pixels it operates on.

5.2. Fact Knowledge

When an object is recognised, it is matched to one of the views of the object frame. Matching instantiates image features to all edges and surfaces within the view frame. If a frame fails to match sufficient slots, it is exchanged for a frame of a spatially neighbouring view, and matching is attempted again. View frames also store type descriptions for each edge. Surface frames contain a list of attached edges and their types for the current view which is passed to them by the instantiated view frame. They contain information about a reflectance model (this may be the same for all surfaces). They may also contain a description of expected surface shape if the system is employed for model matching. Finally, they contain a normal map for the surface, which is a product of the process. Our processing method treats occluding edges and orientation discontinuities differently, but a single edge subframe supports both types, as an edge may be occluding in one view, and an orientation discontinuity in another. The frames contain the pixel locations of the edges, which also delineate the surfaces in the image. For curved occluding edges, the tangent will be calculated at every point along the edge. For orientation discontinuities, the edge frame has a set of pixel locations for each surface, these are typically a few pixels away from the edge to avoid difficulties with narrow bands of high curvature. Orientation discontinuity edge frames also have a slot for the angle between the surface normals of the edges, in most cases this is a single value. However, for more complex surfaces this may be a function of position along the edge. If the edge is not curved and is occluding, only pixel location information is used.

5.3. Procedural Knowledge

Procedural knowledge consists of: the numeric processes that can be performed to find the shape of an object, the order and way in which they are per-

```
Frame Object {
        integer     NumViews
        Frame View  *Views
}

Frame View {
        integer                         NumSurfaces
        integer                         NumEdges
        Shape_description               *Surface_shape_list
        Frame Surface                   *Surfaces

        struct {
               Frame    Edge edge
               integer  Edge_type
        }                               *Edges
}

Frame Surface {
        Surface_reflectance_model       R
        /* Shape is used for model-based matching only */
        Surface_shape_description       Shape

        Vector                          *Derived_Surface_Normals

        struct {
                Frame   Edge edge
                integer Edge_type
        }                               *Edges

}

Frame Edge {
        integer  NumPixels
        struct {
                Vector  Unit_tangent /* Calculated at run time */
                integer x1
                integer y1
                /* Only used for non-occluding */
                integer x2
                integer y2
        }                               *Pixel_information

        /* non-occluding only */
        Angle_map                       Angle_between_surfaces
        /* occluding only */
        Boolean                         Edge_orientation
}
```

Figure 4: There are six frame types in our system. Object frames store all the information we require about objects, which consist of most our other frame types: views; surfaces; and edges. There is also an environment frame, and an image frame to mediate access to the intensity image.

formed, and the process of initialisation (setting terminal defaults) - which may be domain specific. The numerical processes are Brooks and Horn's method for propagating shape across surfaces given boundary conditions, our new method of calculating the surface normal along orientation discontinuity, and the method of setting the surface normal along limb boundaries to the tangent to the edge value. The required order of processing is to calculate normals for any solvable edges, then solve for the surfaces, and finally combine this for the view. Also, before any object recognition shape matching can occur for a surface, all numeric shape processing must be complete for that surface. Similarly, all surfaces must have completed processing and matching before matching can occur for the view. Finally, default values for surface terminals can generally be set using surface direction based on estimates of object pose extracted during model matching, and the assumption of a flat surface. However, far better initialisation can be achieved in specific domains. This is discussed further in the results.

5.4. Shape processing rulebase

The rulebase assumes as a starting point that an edge matching process has instantiated a view frame to an image. Within the view frame all surface and edge frames are instantiated to sets of pixel locations in the image. Our rulebase is partitioned by our frame topology, and is shown in Table 1. There may also be further domain specific rules.

6. Experimental Method and Results

These experiments demonstrate the efficacy of our knowledge-based approach to shape-from-shading. The output of these experiments is a surface normal map. As we are not interested in deriving the exact shape of the object in the image, 3D reconstruction is not justified. For object recognition

Antecedent	**Quantity**
Object	
Process: OR view match succeeds	Object match found for current view.
Process: OR view match fails	match next closest neighbouring view.
Process: OR view match fails no more views to match	Object match fails.
Process: OR no evidence from view match	match next closest neighbouring view. Do not reject unless another view match is found.
View	
Process: OR No surfaces can calculate shape	No evidence.
Process: surface normal map No surfaces can calculate shape	No surface normals can be found.
Process: OR Shape calculated for some surfaces and all of these gave a correct match.	Correct match.
Process: OR One or more surface shape match failed acceptable parameters	View match fails.
Process: surface normal map One or more surfaces produce surface normal map	Write normal map for extracted surfaces.
Surface	
One or more boundary conditions	Calculate surface normals Using Brooks and Horn's method
Process: OR Convergent surface solution	Derive surface properties and compare to expected ranges
Process: OR No boundary conditions available	no evidence for surface.
Process: surface normal map No boundary conditions available	no normal map for surface.
Process: OR Fails to converge	no evidence for surface.
Process: surface normal map Fails to converge	no normal map for surface.
Limb boundaries	
	Set surface normals at all points to the edge normal in the direction away from the surface.
Orientation discontinuities	
	Apply the new shape from shading algorithm.

Table 1: Rule base for knowledge-based shape-from-shading. This allows for the process being either Object Recognition (OR), or producing a surface normal map.

purposes, estimates of properties such as surface curvature can be calculated directly from the surface normals. For robot navigation purposes an estimate of protrusion of the surface beyond the edges will often be sufficient, which can also be approximately estimated from the surface normals.

Our algorithm was tested against synthetic images generated by the POV Ray ray tracing package to allow full verification of results compared to ground truth, and run against real images taken through a minitron javelin CCD camera, and from the ORL face database. The synthetic images were used both raw, and with varying amounts of gaussian noise added.

The images we used were:

- a synthetic sphere;
- a synthetic dodecahedron;
- a camera image of an egg;
- a camera image of a guitar; and
- face images from the ORL face database.

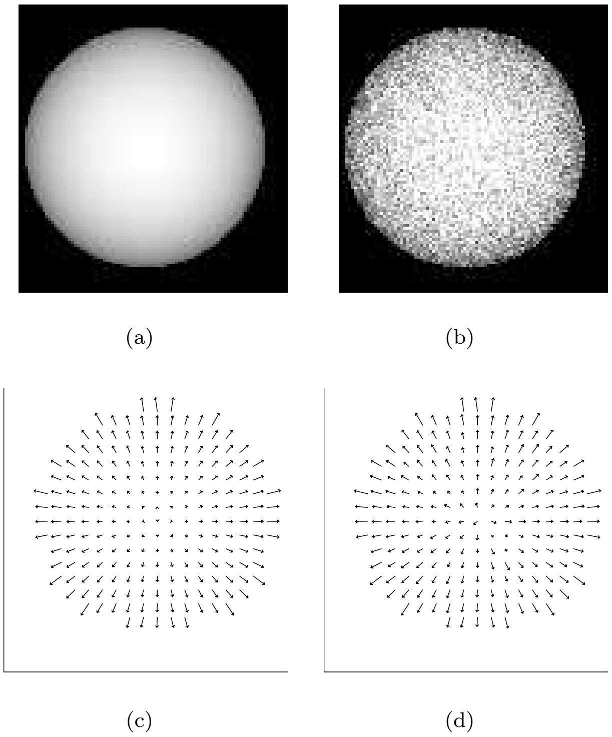
6.1. Synthetic Images

The gaussian noise was generated with a signal to noise ratio based on the mean signal (intensity value) of the image. Figures shown are measured in decibels:

$$10 \log_{10} (\frac{s}{n})^2 .$$

Fig. 5 shows the performance of our algorithm against a synthetic sphere. This demonstrates our algorithm is able to automatically find the shape of a curved object with curved occluding boundaries. The boundary normals are set without intervention based on the normal to the tangent of the occluding edge, assuming the location of the edge is found prior to this process. Fig. 5 (c) shows the needle map extracted for the synthetic sphere of Fig. 5 (a). The image with gaussian noise added at a signal to noise ratio of 10db is shown in Fig. 5 (b), and Fig. 5 (d) shows the extracted surface normal map using a 9 point mean filter. For Fig. 5 (a) with no noise, the error from ground truth is 9.1%. This is largely due to the digitisation errors in the shape of the edge resulting in a less than perfect estimate of the surface normal at the occluding boundary. This error can be seen at the edge surface normals, particularly at the top. This could be improved somewhat by using a more accurate discrete approximation to the tangent than the 3 point estimate that we used for these experiments. The surface normal map for the sphere was initialised to be entirely pointing out of the page, consistent with a flat surface.

Fig. 6 shows our system's performance on a dodecahedron. In this case there are no visible curved occluding boundaries or singular points. No existing variation based system we have seen could produce surface normals without explicitly setting boundary conditions. With a noiseless image the error was 2.9%. With the magnitude of the noise equal to the magnitude of the signal, the surface normal pattern still shows the shape of the object. The surfaces are still visibly planar as shown in 6 (d), despite the image being quite distorted 6 (b). The surfaces for the dodecahedron were initialised individually, consistent with the knowledge-based approach. The leftmost surface was initialised to point to the left-hand side of the image, similarly, the rightmost surface was pointing to the right-hand side. The surface facing the viewer was pointing directly toward the viewer, and the upper surface was pointing straight up. In



(a) (b)

(c) (d)

Figure 5: The synthetic image of a sphere (a) image without noise, (b) image with gaussian noise of 10db SNR, (c) the surface normals calculated for the noiseless image, (d) the surface normals calculated for the noisy image.

a typical case for a recognition system, these surfaces would be initialised based on an estimate of object orientation.

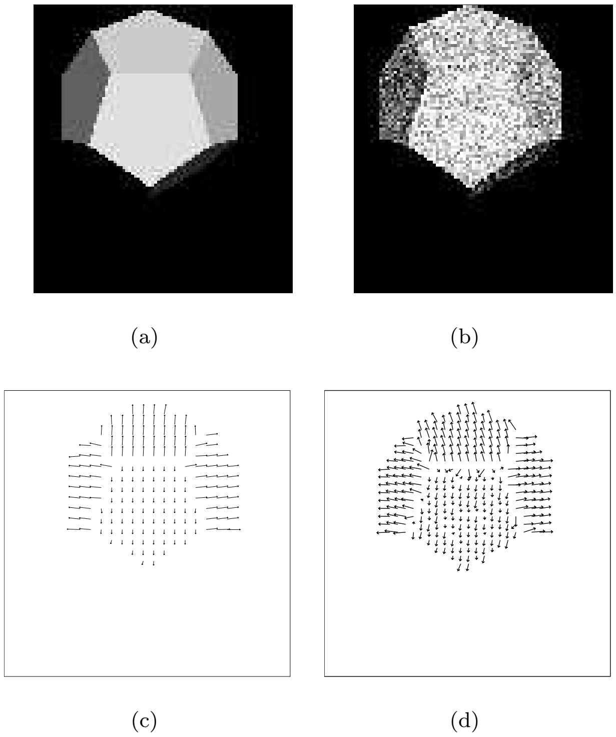


Figure 6: The original dodecahedron image. (a) Clean image, (b) Image with gaussian noise added with a magnitude equal to the signal. (c) Calculated surface normals for the clean image, (d) Calculated surface normals for the noisy image. We ignore the two lower surfaces as too little surface area is shown to contribute much.

Table 2 shows the performance against the sphere and dodecahedron images with varying amounts of noise. The first column shows the performance using a nine point mean, the second column shows performance against the image using raw intensity values. The results show that although there is noticeable degradation of performance with the addition of noise, it is reasonably graceful. They also show that the mean filter gives a small negative effect for images with very low noise, but an increasingly positive effect with larger noise level. We also attempted trials with a 4-point mean filter, the effect was similar, but less significant.

SNR **Sphere**	**Error with mean**	**Error without mean**
No noise	0.093	0.091
40 db	0.096	0.095
20 db	0.104	0.117
15 db	0.115	0.147
10 db	0.140	0.200
Dodecahedron		
No noise	0.040	0.029
40 db	0.040	0.030
20 db	0.045	0.045
15 db	0.051	0.062
10 db	0.069	0.100
0 db	0.159	0.255

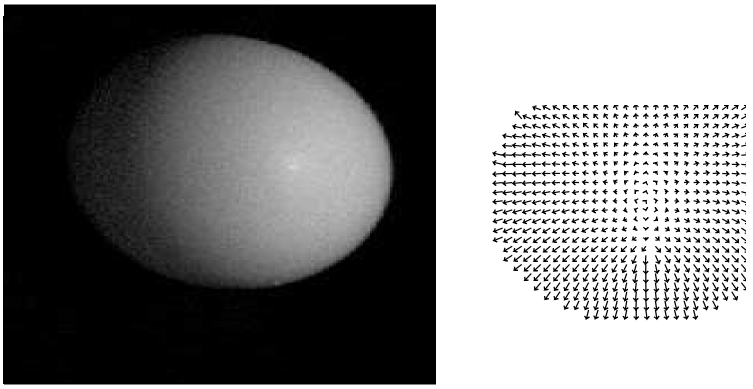
Table 2: For the signal to noise ratio shown, after 1000 iterations the results were as shown in the second column. The third column shows the performance when a 9 point mean filter is used over the image.

The use of a mean filter here may yield a computation time advantage in the case where knowledge-based shape from shading is used to derive an approximate value, such as is the case for object recognition. If only approximate shape is required, we may perform the calculation at a lower resolution based on local means. This will not result in the degradation of edges because our knowledge-based approach allows us to apply the mean filter separately across surfaces, and along edges.

6.2. Real Images

The images shown here are large (256*256), and as such took a significant amount of time for the process of regularisation to converge. We apply Horn's suggestion[15] of modifying the weight of the regularisation. By beginning with a large weight (1) on regularisation, the solution converges. Once the solution has converged we reduce the regularisation component to be small (0.005) so the method can converge to a more correct solution. The regularisation component is not removed entirely because, we cannot expect an exact solution with real data.[15] The solutions still took thousands of iterations to converge, which highlights the problems of dealing with high resolution images at full scale, rather than finding a solution at a scale reduced with the use of a mean filter.

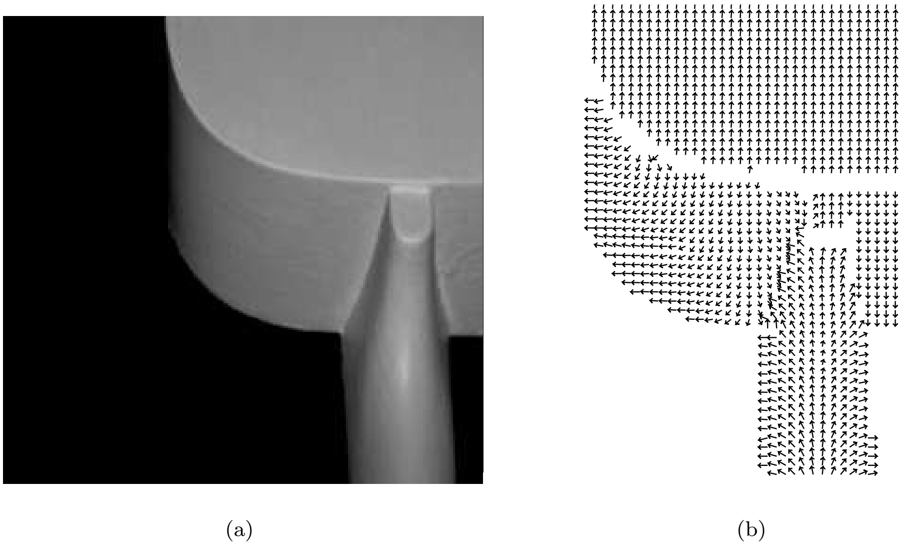
Fig. 7(b) shows the performance of our algorithm on an egg. Although there is no clear method of precisely evaluating the performance of the system in this case, the surface normal needle map can be seen to correspond approximately to the shape of the egg.



(a) (b)

Figure 7: The A camera image of an egg, and the surface normals extracted by our process.

Fig. 8(b) shows the surface normals extracted from the image of a guitar, Fig. 8(a). This shows our method is capable of deriving the shape of complex objects, with a combination of planar surfaces, occluding limb boundaries, and orientation discontinuities. However, as the full boundary information for the surface cannot be inferred, some of the surfaces have not fully converged to the correct solution. With planar surfaces initialisation based on known edges ensures this is not a problem. However, for the large curving surface on the body of the guitar, in the region near the occluding non-limb boundary, the solution is not correct. Partial solutions in the case of incomplete boundary information is discussed in Ref. 18. The partial information extracted can still be used by a knowledge-based system.



(a)

(b)

Figure 8: A camera image of the back of a guitar, and the shape extracted.

6.3. Domain Knowledge

In the domain of face recognition, we can make strong assumptions about the underlying structure to help us extract the shape of faces. For instance, the presence of a nose in the centre of the face, will be an appropriate assumption for frontal face images. Assumptions of this nature allow boundary conditions to be found, and can provide good initialisation of surface shape which could dramatically decrease the computational time required to find the shape of the face. To illustrate, we give examples of two faces from the Olivetti and Oracle Research Laboratory database of faces, see http://www.cam-orl.co.uk/facedatabase.html. The surface normals at the boundaries of the face were set as occluding limb boundaries. (Note this only fully bounds the face if there is little or no hair!) We initialised the normals for the faces to a convex ellipsoidal shape, which gives the shape from shading process a significant start. More complex models based on standard face shape, and more generally correct boundary assumptions could clearly be applied.

After only 200 iterations the solution had converged. The normals only changed on average by 0.6% in the next 800 iterations. If we began with a flat surface and regularisation weight of 0.005 the solution was visibly incorrect after thousands of iterations. If we ran with an initially high weight for regularisation the solution also took thousands of iterations to converge.

7. Conclusion

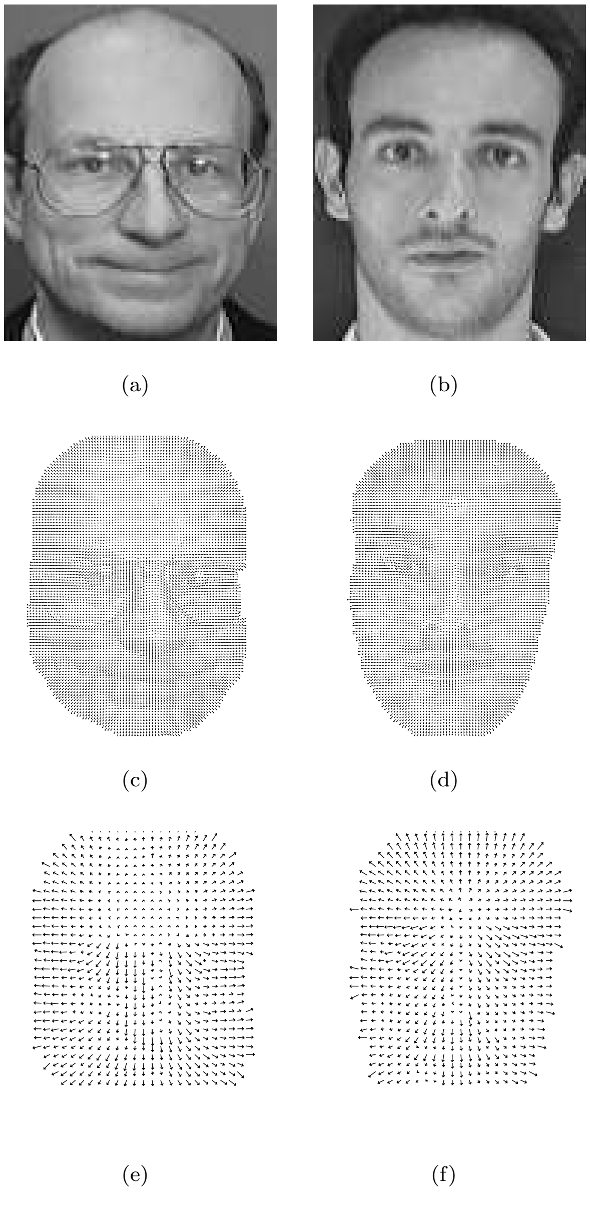
We have presented a novel approach to the shape from shading problem, which applies knowledge of the object in view to aid in inferring shape. By using a knowledge-based approach we are able to derive the shape for complex objects which are only piecewise smooth, and may have no limb boundaries or singular points. We derived a method for uniquely finding the shape along orientation discontinuities within the image, where the angle of the adjoining surfaces is known. This method can be applied to object recognition where it can be used without user intervention. We also demonstrated that the application of domain knowledge can yield computational advantages.

We have demonstrated some advantages of knowledge based image processing, particularly here for robot navigation and domain specific scene understanding.

8. Further Work

This work has demonstrated the efficacy of a knowledge-based approach to image processing, a powerful paradigm for image processing tasks. There are many other image processing tasks where this paradigm could be applied.

Within shape-from-shading specifically, knowledge-based solutions to other standard problems could be found. Such as problems with more complex lighting models, and with reflectance models that vary across the object.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 9: Shape calculated from faces (a) The first face image (b) another face (c) derived normals of the first face with a high density of displayed points, (d) high density image of normals for the second face (e) derived normals with a low density of displayed points, (f) low density image of normals for the second face.

Further, this paper is preliminary work for integrating shape information with edge-based object recognition. The next necessary step is to build an integrated edge/surface shape object recognition system.

1. N. M. Barnes and Z. Q. Liu, "Vision guided circumnavigating autonomous robots", *IEEE Trans. Robotics and Automation,* (1997) revised.
2. I. Biederman, "Human image understanding: Recent research and a theory", *Computer Vision, Graphics, and Image Processing*, **32** (1985) 29–73.
3. M. J. Brooks and B. K. P. Horn, "Shape and source from shading", in *Shape From Shading*, eds. B. K. P. Horn and M. J. Brooks, The MIT Press: Cambridge Massachusetts, London England, 1989, pp. 53–68.
4. A. R. Bruss, "The eikonal equation: Some results applicable to computer vision", in *Shape From Shading*, eds. B. K. P. Horn and M. J. Brooks, The MIT Press: Cambridge Massachusetts, London England, 1989, pp. 69–87.
5. R. Davis, "Expert systems: Where are we? and where do we go from here?", in *Knowledge-Based Systems: Fundamentals and Tools*, eds. O. N. Garcia and Y.-T. Chien, IEEE Computer Society Press, Los Alamitos, California, 1992, pp. 7–26.
6. S. J. Dickinson and D. Metaxas, "Integrating qualitative and quantitive shape recovery", *International Journal of Computer Vision*, **13**, 3 (1994) 311–330.
7. F. P. Ferrie and M. D. Levine, "Where and why local shading analysis works", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **11**, 2 (1989) 198–206.
8. R. Fikes and T. Kehler, "The role of frame-based representation in reasoning", *Communications of the ACM*, **28**, 9 (1985) 904–920.
9. O. N. Garcia and Y.-T. Chien, "Introduction", in *Knowledge-Based Systems: Fundamentals and Tools*, eds. O. N. Garcia and Y.-T. Chien, IEEE Computer Society Press, Los Alamitos, California, 1992, pp. 1–6.
10. M. P. Georgeff and A. L. Lansky, "Procedural knowledge", *Proceedings of the IEEE*, **74**, 10 (1986) 1383–1398.
11. A. J. Gonzalez and D. D. Dankel, *The Engineering of Knowledge-Based Systems: Theory and Practise*, Prentice Hall, 1993.
12. D. Harwood, M. Subbarao, H. Hakalahti, and L. S. Davis, "A new class of edge-preserving smoothing filters", *Pattern Recognition Letters*, **6**, 3 (1987) 155–162.
13. D. A. Holton and J. W. Lloyd, *Algebra and Gemetry*, Charles Babbage Research Centre:Manitoba, Canada, 1978.
14. B. K. P. Horn, "Obtaining shape from shading information", in *The Psychology of Computer Vision*, ed. P. H. Winston, McGraw-Hill Book Company, New York, NY, 1975, pp. 115–155.
15. B. K. P. Horn, "Height and gradient from shading", *International Journal of Computer Vision*, **5**, 1 (1990) 37–55.
16. B. K. P. Horn and M. J. Brooks, *Shape from Shading*, The MIT Press:Cambridge, Massachusetts:London, England, 1989.
17. B. K. P. Horn and R. W. Sjoberg, "Calculating the reflectance map", *Applied Optics*, **18**, 11 (1979) 1770–1779.
18. K. Ikeuchi and B. K. P. Horn, "Numerical shape from shading and occluding boundaries", in *Shape From Shading*, eds. B. K. P. Horn and M. J. Brooks, The MIT Press: Cambridge Massachusetts, London England, 1989, pp. 245–299.
19. K. N. Kutulakos and C. R. Dyer, "Global surface reconstruction by purposive control of observer motion", *Artificial Intelligence*, **78**, 1-2 (1995) 147–177.
20. C. H. Lee and A. Rosenfeld, "Improved methods of estimating shape from shading using the light source coordinate system", in *Shape From Shading*, eds. B. K. P. Horn and M. J. Brooks, The MIT Press: Cambridge Massachusetts, London England, 1989, pp. 323–347.
21. D. Lee, "A provably convergent algorithm for shape from shading", in *Shape From Shading*, eds. B. K. P. Horn and M. J. Brooks, The MIT Press: Cambridge Massachusetts, London England, 1989, pp. 349–373.
22. J. Malik and D. Maydan, "Recovering three-dimensional shape from a single image of curved objects", *IEEE Transactions on Pattern Analysis and Machine*, **11**, 6 (1989) 555–566.
23. D. Marr, *Vision : a computational investigation into the human representation*

and processing of visual information, W.H. Freeman, New York, 1982.

24. M. Minsky, “A framework for representing knowledge”, in *The psychology of computer vision*, ed. P. H. Winston, McGraw-Hill, New York, 1975, pp. 211–277.
25. S. K. Nayar, K. Ikeuchi, and T. Kanade, “Surface reflection: Physical and geometrical perspectives”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **13**, 7 (1991) 611–634.
26. A. Pentland, “Local shading analysis”, in *Shape From Shading*, eds. B. K. P. Horn and M. J. Brooks, The MIT Press: Cambridge Massachusetts, London England, 1989, pp. 443–487.
27. A. P. Pentland, “Linear shape from shading”, *International Journal of Computer Vision*, **4**, 2 (1990) 153–162.
28. A. A. G. Requicha, “Representation of rigid solids: Theory, methods, and systems”, *Computing Surveys*, **12**, 4 (1980) 437–464.
29. H. Schultz, “Retrieving shape information from multiple images of a specular surface”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **16**, 2 (1994) 195–201.
30. P.-S. Tsai and M. Shah, “Shape from shading using linear approximation”, *Image and Vision Computing*, **12**, 8 (1994) 487–498.