

University of South Wales



2059578

**Recognition and Orientation of Specular
Objects from the Analysis of Plane
Polarised Light**

Paul Thomas Fairney

*Submitted to the Council for National Academic
Awards in partial fulfillment of the requirements
for the degree of Doctor of Philosophy*

*Department of Computer Studies,
The Polytechnic of Wales,
Pontypridd.*

June 1989

Abstract

Recognition and Orientation of Specular Objects from the Analysis of Plane Polarised Light.

P.T.Fairney.

We have used the intersection of polarised reflectance contours to determine surface orientation. This information is used to both to recognise and to determine the orientation of dark, specular, dielectric objects. The system uses three images of an object viewed through a plane polarising filter at different rotations.

The objects are described by their Extended Gaussian Images (EGIs). The process always provides a description of only that part of the object which is visible to the camera, i.e. a viewer-centred description. This description of the object takes the form of a histogram. The cells of the histogram represent surface orientations, and the frequency in each cell is the visible area of the object at that orientation.

This sensed histogram is compared with a database of pre-computed, viewer-centred, model orientation histograms. The best match determines the type of object and its orientation in space. Matching with a pre-computed database is fast, efficient and determines orientation to within an average uncertainty of less than 7 degrees of arc from fewer than 30 comparisons.

Acknowledgements

I wish to express my sincere thanks to my supervisors Dr. B.F. Jones, Dr. R.J. Wiltshire, and Dr. D.H. Smith for their helpful comments and advice on this thesis. My particular gratitude to Dr. B.F. Jones for his encouragement and enthusiasm and above all for his constructive and positive advice during the course of this work.

I would also like to thank D. Illing for her helpful comments on this text.

Table of Contents

List of Symbols.	<i>v</i>
1. Introduction.	<i>1</i>
2. Calculating the Reflectance Map of Specular Dielectric Objects Viewed Through a Polarising Filter.	<i>10</i>
2.1 The Co-ordinate System.	<i>10</i>
2.2 The Relationship Between Surface Orientation and the Degree of Polarisation of Reflected Light.	<i>11</i>
2.3 Calculation of the Reflectance Map of a Dielectric, Specular Sphere Viewed Through a Polarising Filter.	<i>15</i>
3. Proposal for using Polarised Light to Recognise Objects.	<i>24</i>
3.1 Obtaining a Unique Triple of Intensity.	<i>24</i>
3.2 Testing the Feasibility of a Computer Vision System using a Computed Model of a Cylinder.	<i>28</i>
4. Equipment Used.	<i>30</i>
5. Test Object Description, Representation and Orientation.	<i>34</i>
5.1 Object Description.	<i>35</i>
5.2 Object Orientation.	<i>37</i>
6. System Calibration.	<i>39</i>
6.1 Locating the Sphere and Estimating its Radius.	<i>41</i>
6.2 Selecting the Sampling Points.	<i>42</i>
6.3 Quantisation of Radiance Values.	<i>47</i>
6.4 Building the Look-up Table.	<i>51</i>

6.5 Segmentation Thresholds.	54
7. The Database of Viewer-Centred Descriptions.	57
7.1 Computing the Viewer-Centred Model Descriptions.	58
7.2 Features of the Model Object.	65
7.3 The Stored Orientation Histogram.	69
7.4 Normalising the Database of Descriptions.	71
7.5 Storing the Database of Descriptions.	72
7.6 Inserting a Histogram into the Database.	75
7.7 An Example of Inserting a Histogram into the Database.	77
7.8 Dealing with Negative Addresses.	80
7.9 Non-Linear Scaling of Radiance Values.	81
8. Object Recognition and Orientation.	84
8.1 Building the Sensed Description.	84
8.2 Features of the Visible Object.	86
8.3 Retrieving a Histogram from the Database.	89
8.4 Matching Image and Model Orientation Histograms.	91
9. Results.	93
9.1 Results: Orientation Only.	94
9.2 Results: Recognition and Orientation.	96
9.3 The Effects of Noise.	97
9.4 Results: Recognition and Orientation using a Variable Threshold.	101
9.5 Repeatability.	103

10. Discussion.	105
10.1 Practical Considerations.	117
10.2 Future Work.	119
11. Conclusions.	122
References.	123

List of Symbols

(α, β) : orientation	10
α : latitude	10
β : longitude	10
x, y, z : co-ordinate axes.	10
L : radiance [$\text{Wm}^{-2} \text{steradian}^{-1}$]	11
L_i : recorded radiance values.	39
\mathbf{n} : surface normal	12
i : angle of incidence	11
e : angle of emittance	12
r : angle of refraction	20
g : phase angle	14
ω : solid angle	12
η : distance from viewer to surface patch.	12
Φ : radiant flux. [Watts]	12
ζ : vector length	10
ξ : vector direction.	15
E : the electric field.	16
H : the magnetic field.	16
$\hat{\mathbf{p}}$: the electric field unit vector in the plane of incidence.	16
$\hat{\mathbf{s}}$: the electric field unit vector perpendicular to the plane of incidence.	16.
R_p' : reflected amplitudes of electric field in plane of incidence from an absorbing surface.	18
R_s' : reflected amplitudes of electric field perpendicular to the plane of incidence from an absorbing surface.	18
R_p : the reflected amplitude in the plane of incidence.	21
R_s : the reflected amplitude perpendicular to the plane of incidence.	22

A_p : the incident amplitude in the plane of incidence.	18
A_s : the incident amplitude perpendicular to the plane of incidence.	18
T_p : the transmitted amplitude in the plane of incidence.	22
T_s : the transmitted amplitude perpendicular to the plane of incidence.	22
$\hat{\mathbf{f}}$: polarising filter rotation unit vector.	18
μ : the refractive index	20
P : the degree of polarisation.	24
t_i : threshold; used to segment sphere from background during calibration.	41
σ_i : standard deviation of calibration image of sphere.	41
τ : estimated radius of the sphere.	42
τ' : computed radius of sphere.	42
τ_i'' : projected radius of each grid circle.	46
ψ : the angular distance between two vectors.	43
ε : the angular distance between grid circles.	45
ε' : angular distance between sample points on each grid circle.	46
t_a : threshold; used to help determine on which sampling points an objects surface may reside.	61
h : average foreshortening factor.	66
A_a : apparent surface area of the visible object.	66
A_t : true surface area of the visible object.	66
N_i : normalised radiance values.	81
S_i : Non-linear scaled radiance values.	82
M_{ab} : moments, where $a, b = 0, 1, 2, \dots$	86
p : principal axis of the 2-D projection of the object.	67
$C(x), C(z)$: Centroid of the 2-D projection of the object.	87
t_p : threshold; upper and lower limits of principal axis.	87
t_f : threshold; fixed threshold used to determine the larger visible cell frequencies of the sensed histogram.	87

t_v : threshold; variable threshold used to determine the larger visible cell frequencies of the sensed histogram .	87
s : standard deviation of sensed histogram.	87
t_h : threshold; upper and lower limits of the average foreshortening factor.	87
f_{m_i} : frequency in cell i of model histogram.	91
f_{s_i} : frequency in cell i of sensed histogram.	91
S : residual sum of squared differences.	91
t_m : matching threshold.	91
λ : signal to noise ratio (snr).	97
σ_g : standard deviation of Gaussian distribution.	97
μ_g : mean of the Gaussian distribution.	97
f_r : uniformly distributed random number.	97
\bar{f} : mean frequency of sensed orientation histogram.	98
ρ : correlation coefficient.	99

CHAPTER 1.

Introduction

Computer Vision is widely accepted as a set of serial processes that include four main stages of image analysis: image restoration, segmentation, description and model matching. We wish to recognise and estimate the orientation of a randomly oriented object from multiple images. As such the task falls largely within this classical concept.

Human beings capture 2-D images of reflected radiance. By using the available cues we perceive the 3-D shape of objects within the image. One such cue is shading; artists mimic the effects of shading in paintings to communicate depth and shape.

The human visual system assumes a single light source,¹ normally above the scene, i.e. the sun. The effects of shading in such circumstances help humans to gather information about the shape of objects and hence to recognise them. These ideas form the basis of several computer vision methods in which, using carefully controlled lighting, the orientation of an object is determined from its reflected radiance. In computer vision the relationship between reflected radiance and surface orientation is called the reflectance map.²

An image captured by a T.V. camera is a 2-D array of reflected radiance values. Four factors influence these recorded values: surface orientation, illumination, surface material and imaging geometry. Given the same illumination, fixed imaging geometry and surface microstructure, reflected radiance is determined wholly by surface orientation. The reflectance map is of practical use because it can be constructed empirically in the form of a look-up table. This relates scene radiance to surface orientation and is used to build descriptions of objects.

As far back as 1966 Rindfleisch⁵ used known lunar photometric properties, along with radiance measurements taken from images captured by a Ranger spacecraft to find surface elevation of hills in the maria of the moon; the maria refers to the dark areas or 'seas' on the lunar surface.

A Lambertian surface reflects light such that the radiance is the same for all viewing angles. When it is illuminated by a distant collimated source the reflected radiance depends only upon the angle of incidence.³ A single image is insufficient to determine a unique surface orientation; three images are required⁴ each viewed from the same position but with different light source directions. Photometric Stereo⁴ uses this principle to empirically determine surface orientation from multiple views of the same object by creating differently shaded images of a viewed object. In contrast highly specular surfaces have little shading information since the dominant rays reflected from the surface are specular.

Reflected radiance depends upon the light source distribution, surface material and surface orientation. Horn² uses the reflectance map to give image radiance as a function of surface orientation for a fixed imaging geometry. The reflectance map shows scene radiance as a function of surface orientation in a viewer-centred co-ordinate system,³ and is usually depicted as a series of contours of constant radiance.

Woodham⁶ uses several images of the same object but with varying incident illumination to determine surface orientation from radiance values. He calls the method Photometric Stereo. A Lambertian surface is assumed where reflected radiance depends solely upon the angle of incidence.

The assumption of uniform surface reflectance is relaxed by Horn et al⁷ using a reflectance factor which varies across the surface being analysed. Silver⁸ finds the

surface albedo, which is the reflecting power of a matt surface, to develop a more complex model of surfaces with non-uniform reflectance properties.

Many everyday objects have surfaces that combine both a specular and a Lambertian component. Coleman & Jain⁹ use four light sources to obtain 3-D shape information about visually textured and specular surfaces. They find that the radiance values created by specular reflection are too high and give incorrect look-up table references. They use the fourth light source to detect and to remove specular reflection.

Ikeuchi¹⁰ uses a distributed light source obtained from an uneven illumination of a diffusely reflecting plane to adapt photometric stereo for a bin of highly specular components. The light source distribution affects the accuracy of the resulting surface orientations.

Ray et al¹¹ look at the problems of implementing Photometric Stereo on objects with non-Lambertian surfaces. They conclude that normal light sources can be used only in the absence of strong specular reflection and shadows.

Horn et al¹² use fluorescent tubes to illuminate specular objects in a bin of similar objects to determine estimates of object orientation.

Brou¹³ uses laser range data to produce a description of the viewed object. The method works well for white objects possessing a slightly rough surface but is less suited to dark highly specular objects which are difficult to scan. The recognition of glossy objects is difficult because they are characterised by their conspicuous reflective properties; namely they reflect a virtual image of the surrounding scene. The nature of rays reflected from such a surface presents problems when illuminating glossy surfaces.

The shape of an object determines what the camera sees at each orientation, and its size determines visible area. A suitable representation of object shape is required to enable matching of image descriptions to model descriptions. A convex object can be uniquely defined by the area and orientation of its faces.^{14,17} A Gauss sphere is a unit sphere used to represent surface orientation, and objects are characterised by their Gaussian Images.¹⁷ The Gauss image is formed by placing the object at the centre of a notional Gaussian sphere which completely encloses it. Projecting the orientation of each surface of the object onto the sphere in the same orientations gives us the Gauss mapping. The Extended Gaussian Image (EGI), (Bacszy¹⁵ & Horn¹⁶), is an extension to this and relates surface orientation to surface area. The area of a surface can be represented on the Gauss sphere by a mass equal to the surface area at that orientation.

The EGI itself is a continuous representation of orientation and surface area and any practical representation of an object's shape must be quantised. The orientation histogram is a discrete approximation of the EGI and is created by tessellating the Gaussian sphere into a fixed number of cells; each cell is represented by a unique orientation. Each cell of the orientation histogram corresponds to a cell on the tessellated sphere. Surface area at each discrete orientation is represented by the frequency in the corresponding cell of the histogram. EGIs and their discrete representation, the orientation histogram, are described by Horn.¹⁷ The orientation histogram is used to describe both prototypical object models and the visible or sensed object. Rotating the prototypical description until it aligns with the viewed description gives the estimated orientation of the sensed object. Each rotation of the model requires the reprojection of its surfaces onto the Gaussian sphere to produce an orientation histogram for subsequent comparison with sensed data. Recognition is regarded as a separate process in systems where more than one type of object is present.

EGIs are translation and rotation invariant; the object's description will not vary regardless of where the object lies on the 2-D the image plane. Similarly object rotation will result in an equal rotation of its EGI.

Ikeuchi¹⁸ uses the EGI to help recognise 3-D objects. He notes that incomplete observations of objects create matching problems. To help solve this problem he computes several orientation histograms of the object from the same viewing direction. One histogram contains the full visible hemisphere of surfaces, and the remaining histograms contain partial views of the object. The amount of the visible object contained within each partial view is determined by a visible disk, i.e. equivalent to viewing the EGI through a variable diameter iris. Each visible disk is defined by a circle whose centre is the y-axis, *see figure 2*; each has a different radius and therefore contains only the proportion of the visible hemisphere within the corresponding disk. These are pre-computed for each viewing direction to form a 3-D look-up table; the axis of least inertia and the projected area of each disk are computed and added to each histogram to constrain possible viewing directions. A matching function determines object orientation after comparing the observed distribution of mass with the prototypes of possible orientations. It is noticeable that the question of object orientation and recognition has been separated. Ikeuchi¹⁹ proposes a method of rotating the object-centred EGI; he uses a geometric modeller to rotate the model and produce viewer-centred descriptions (EGIs) of the rotated object for matching purposes.

Brou¹³ uses combinations of spherical tessellations of 4-D polytopes to provide an even sampling of the sphere; he uses up to 5880 points or orientations. Tessellations based upon a regular polyhedron, such as an icosahedron, are used. An icosahedron has twenty identical triangular faces and sixty vertices; one model description is required for each viewing direction determined by a vertex orientation. Both model and object (sensed) descriptions are built using identically tessellated spheres. The

icosahedron allows one sphere to be rotated into alignment with the second sphere in sixty different orientations. For each orientation tested, the model is rotated into position and its surfaces projected onto the Gaussian sphere before matching can occur.

Horn²⁰ has adopted Brou's 4-D tessellations for matching EGIs. The space of possible rotations is reduced by using features of the visible object, i.e. the axis of least inertia and the centre of visible mass.

A reflectance map technique requires the reflected radiance to vary as the orientation of the viewed surfaces changes. Rays reflected from specular surfaces are coplanar with the surface normal and the incident ray, and the angles of incidence and reflection are equal. The camera captures only rays reflected along the viewing axis. Hence we require an extended light source if we are to capture specular reflections from every point on the object.

Unpolarised light incident upon an object becomes polarised when reflected or scattered. Only when light is reflected at normal or grazing angles is there no polarisation.²¹ Unpolarised light from an extended source when incident upon a specular dielectric surface is reflected as plane polarised light. The degree of polarisation depends only upon the angle of incidence and the complex refractive index of the reflecting material. The use of polarised light in computer vision has, with few exceptions,^{22,23} been largely overlooked.

Polarisation by dark specular dielectric surfaces can be very strong. Darker surfaces absorb more of the incident light so that the rays that are reflected are very highly plane polarised. Polarised light is used by Koshikawa²⁴ to determine surface orientation of specular objects; polarimetry is used to measure the degree of polarisation, and images are not used. Koshikawa et al²² and Terashi et al²³ adapted the method for image processing. Eight images of a specular object illuminated by

circularly polarised incident light are required. The camera views the object through a quarter wave plate to enable the analysis of the state of polarisation of reflected rays, and the surface normals are computed from this information.

Specular dielectric surfaces when illuminated by an extended source and viewed through a polarising filter portray a reflectance map of radiance contours. We will investigate the use of polarised reflectance maps to derive surface orientation and hence descriptions of viewed objects.

Object models are generally described using a feature of the object such as the centre of mass; these are termed object-centred descriptions. Rotating this description into alignment with the sensed description is computationally expensive. Brou¹³ finds that, with 1920 rotations and comparisons, his average error is 10.7 degrees of arc. Even with 5880 rotations and comparisons, errors of up to 17 degrees of arc are possible.

If both model and sensed descriptions are described in terms of what is visible to the camera, then the need for spherical tessellations is removed. A fixed grid of points on the surface of the visible hemisphere is sufficient to provide a discrete sampling of the surface of possible orientations. Brooks²⁵ uses a fixed grid of points to discretely sample the continuum of normals over the surface of a sphere. Model descriptions can be pre-computed using a fixed grid of points and stored in a database of similar descriptions in a once only operation. We call these descriptions viewer-centred and will use them to recognise the object and to determine its orientation.

To summarise, reflectance map techniques are difficult to implement on highly specular surfaces using fluorescent tubes or collimated sources. Our experiments indicate that specular highlights, such as the virtual image of the lamp, cause errors in reflected radiance measurements and therefore do not lend themselves to reflectance map techniques, and in particular Photometric Stereo.

Matching object-centred descriptions involves a model rotation, reprojection and comparison for each orientation attempted. This can be computationally expensive. When this type of matching is used, object recognition becomes a separate problem, and the task is reduced to object orientation.

By contrast we will describe a method which builds descriptions of highly specular objects by using the reflected, polarised radiance-contours. This description is sufficient to both identify the object and estimate its orientation.

Figure 1 summarises the previous work in this area and provides a comparison with our own results.

Researcher	No. of Images	Look-up Table	Matching			Object		No. of Comp.	Average uncertainty in degrees of arc
			rot	rep	comp	recog	orient		
Ikeuchi ¹⁸	3	yes	yes	yes	yes	yes	yes	?	?
Ikeuchi ³⁷	3	yes	yes	yes	yes	no	yes	?	?
Horn et al ¹²	3	yes	yes	yes	yes	no	yes	720	5-10
Brou ¹³	*	no	yes	yes	yes	no	yes	5880	7.3
Terashi et al ²³	8	no	yes	yes	yes	yes	yes	?	?
Horn ²⁰	3	yes	yes	yes	yes	no	yes	≈100	5-10
Fairney(<i>this work</i>)	3	yes	no	no	yes	yes	yes	≈28	<7

Figure 1. Summary of approaches.

Legend..

rot : rotation.

recog : recognition

rep : reprojection

orient : orientation

comp : comparison

? : don't know

≈ : approximately.

** : Brou uses lasers to produce a depth map of the viewed object.
Differentiation is used to convert the depth information to a map of
normal vectors.*

CHAPTER 2.

Calculating the Reflectance Map of Specular Dielectric Objects Viewed Through a Polarising Filter.

This chapter is devoted to a theoretical explanation of our proposed method of computer vision. Firstly, *section 2.1* describes our co-ordinate system. We relate surface orientation to the degree of polarisation of the reflected light in *section 2.2*. Finally, the reflectance map of a dielectric sphere is calculated in *section 2.3*.

2.1 The Co-ordinate System.

We have chosen to represent surface orientation in terms of latitude and longitude (α, β). This is closely related to spherical polar coordinates.

It is convenient to fix our co-ordinate system with the y-axis parallel to the viewer direction or optical axis, *see figure 2*.

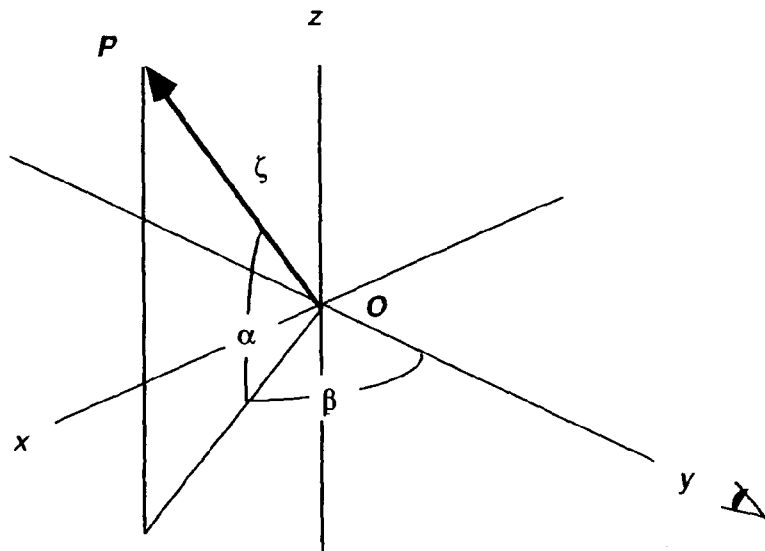


Figure 2. The coordinate system.

In figure 2,

$$x = \zeta \cos \alpha \sin \beta$$

$$y = \zeta \cos \alpha \cos \beta$$

$$z = \zeta \sin \alpha,$$

the vector \vec{OP} is given by

$$\vec{OP} = \zeta \cos \alpha \sin \beta \hat{i} + \zeta \cos \alpha \cos \beta \hat{j} + \zeta \sin \alpha \hat{k}. \quad (1)$$

Unit vectors are denoted by a circumflex.

2.2 The Relationship Between Surface Orientation and the Degree of Polarisation of Reflected Light.

Radiance is a precisely defined radiometric quantity; when radiance is weighted with the spectral response of the eye or camera, we obtain the photometric quantity luminance. Luminance is often confused with brightness which is a subjective, visual sensation. It is therefore difficult to obtain a precise measure of the brightness of an object since it changes as, for example, the luminance of the background changes.

The radiance of a scene is the most important radiometric quantity used in computer vision or, for that matter, human vision. The reasons for this are twofold; firstly imaging devices such as eyes and cameras generate a signal which is proportional to radiance.²⁶ Secondly the radiance of an object is undiminished by distance, assuming no absorption or scattering in the intervening space.²⁶ If we are to model a method of computer vision, we must be able to calculate the radiance of a scene. Radiance, L , is defined as follows. Consider a surface patch of area, δA , which is illuminated by a distant light source. The incident rays make an angle i with the normal

to the surface, see figure 3. The angle between the surface normal, n , and the viewer is e . The foreshortened area is $\delta A \cos e$.

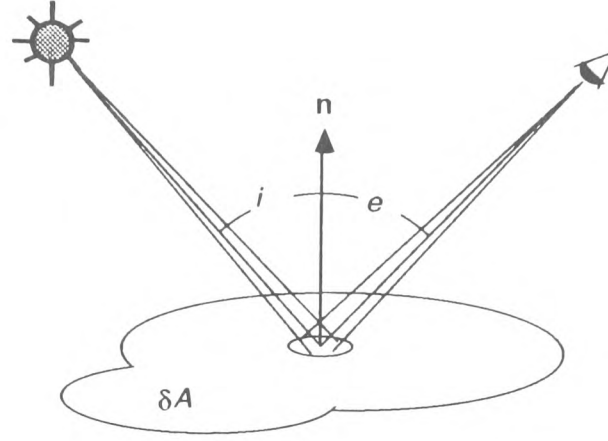


Figure 3. The illumination of a surface patch.

The solid angle $\delta\omega$ subtended by δA at the viewer³ is given by

$$\delta\omega = \frac{\delta A \cos e}{\eta^2},$$

where η is the distance from the viewer to the surface patch. In the *limit*, the $\delta A \rightarrow 0$

equation becomes

$$d\omega = \frac{dA \cos e}{\eta^2}. \quad (2)$$

The radiance³ is the reflected radiant flux, Φ , where $\Phi = \Phi(A, \omega, i, e)$ per unit area and contained within the solid angle $d\omega$. The radiance reflected by the surface in the direction of the viewer is³

$$L(e) = \frac{\partial^2 \Phi}{\partial \omega \partial A \cos e}. \quad (2a)$$

In the case when ω , A , Φ are constant, equations (2) & (2a) combine to give

$$L(e) \propto \frac{\Phi}{\cos^2 e}. \quad (3)$$

A foreshortening factor of $\cos e$ must be included both for the area and the solid angle.

Equations (2) and (3) constitute the simplest form of reflectance map which is a mapping from reflected radiance to orientation. There are two further factors which must be taken into account if the reflectance map is to be used successfully to determine surface orientation; they are the lighting and the nature of the surface.

The two extremes of surface type are the perfectly diffuse (Lambertian) surface and the perfectly specular (mirror) surface. When a collimated beam is incident upon both surfaces the resulting reflections are very different.

A diffusing surface, see figure 4, has the same reflected radiance irrespective of viewing direction.

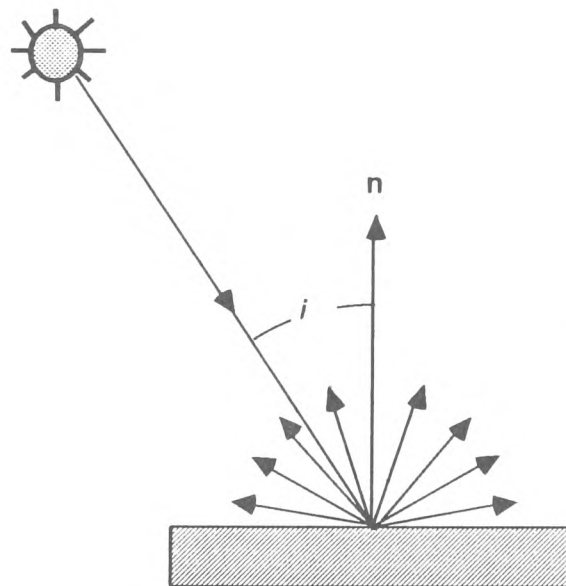


Figure 4. Diffuse reflection.

The radiance of the Lambertian surface depends only upon the angle of incidence i . If the direction of the light beam is known, then the orientation of the surface may be determined from radiance; this is the basis of the method referred to as Photometric Stereo.⁴

At the other extreme, a perfectly specular or mirror surface, see figure 5, reflects light such that the angles of incidence, i , and reflection, e , are equal. The reflected ray is coplanar with the normal to the surface and the incident ray, and the angles i and e are equal.

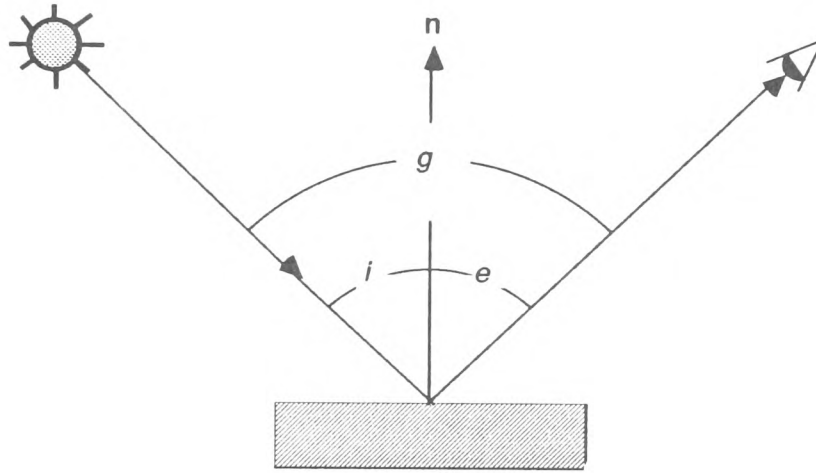


Figure 5. Specular reflection.

The phase angle, g , is defined to be

$$g = i + e = 2i = 2e. \quad (4)$$

A distant eye or camera picks up rays of light travelling in a single direction. If every point on the object is to be visible, reflected rays must emerge parallel to this direction; to achieve this we must illuminate from all directions using an extended light source.

The light reflected from specular dielectric and (to a lesser degree) metallic objects, is linearly polarised. The degree of polarisation depends upon the angle of incidence and on the refractive index and absorptivity of the material; it is measured by viewing the object through a plane polarising filter. In principle, the orientation of the surface normal with respect to the direction of viewing can be deduced from the degree of polarisation and the direction of the plane of polarisation. This forms the basis of our proposed method of computer vision.

2.3 Calculation of the Reflectance Map of a Dielectric, Specular Sphere Viewed Through a Polarising Filter.

To be able to model our computer vision system we must first calculate the polarised reflectance map of a sphere. A sphere is used because its surface has elements oriented in every possible direction; these are used to relate surface orientation to a triple of radiance values to calibrate our working system. We calculate the reflected radiance at every point on the visible hemisphere and determine the effect of viewing the sphere through a polarising filter oriented at any angle.

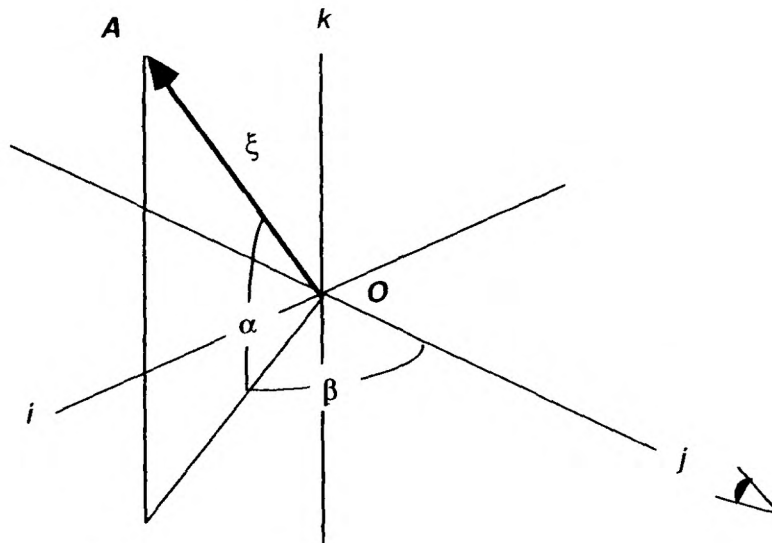


Figure 6. The orientation of a surface patch.

An arbitrary point A on the sphere's surface is located in the direction defined by ξ , from the origin, O , see figure 6, where

$$\xi = \cos \alpha \sin \beta \hat{i} + \cos \alpha \cos \beta \hat{j} + \sin \alpha \hat{k}. \quad (5)$$

Electro-magnetic waves comprise two fields, the electric field and the magnetic field, see figure 7.

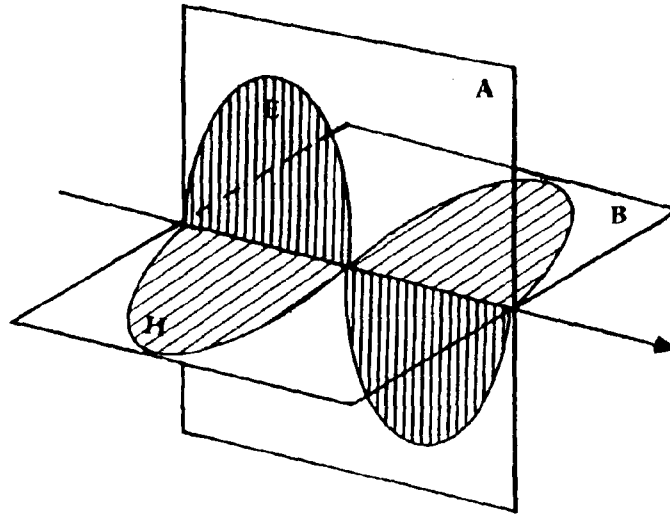


Figure 7. The electric field, E , vibrates in plane A. Plane B contains the magnetic field, H , and is orthogonal to plane A. Both E & H are perpendicular to the wave motion.

To determine the orientation of these vibrations relative to our axes, we must choose one of these two vectors. The electric vector plays a dominant role in optical measurement and because of this the electric vector is generally favoured when describing the effects of polarisation.²⁷

The electric field has components in two directions represented by the vectors \hat{p} and \hat{s} . These vectors are defined in two directions: \hat{s} is normal to both the plane of

incidence and the direction of propagation, and $\hat{\mathbf{p}}$ is in the plane of incidence and normal to the plane of propagation, see figure 8.

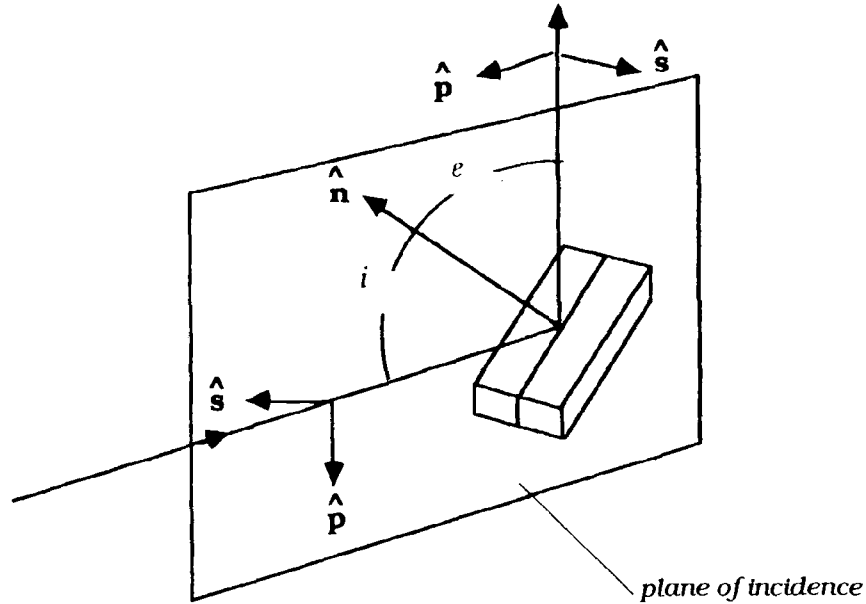


Figure 8. The plane of incidence.

The vectors $\hat{\mathbf{s}}$ and $\hat{\mathbf{p}}$ are defined so that an observer looking in the direction of propagation would make a clockwise rotation from positive $\hat{\mathbf{p}}$ to positive $\hat{\mathbf{s}}$, i.e

$$\hat{\mathbf{p}} \times \hat{\mathbf{s}} = \hat{\mathbf{j}}. \quad (6)$$

Where \times denotes the vector or cross product.

In unpolarised light, the direction of the electric vector will vary from one wave packet to the next. Over a finite period of time, all directions perpendicular to the direction of propagation will be occupied.

When the electric vector is confined to a single direction, perpendicular to the direction of propagation, the light is said to be linearly polarised.

A polarising filter placed in the path of an electro-magnetic wave produces a beam of light whose electric vector vibrates in a single plane. To calculate the polarised reflectance map of our sphere, we must calculate the radiance that passes through the polarising filter from each point on the surface of the sphere. The radiance, L , passing through the polarising filter, see figure 9, will be shown to be (equation 28)

$$L(e) \propto \left[\left| \frac{R'_p}{A_p} \hat{\mathbf{p}} \cdot \hat{\mathbf{f}} \right|^2 + \left| \frac{R'_s}{A_s} \hat{\mathbf{s}} \cdot \hat{\mathbf{f}} \right|^2 \right] \frac{1}{\cos^2 e}.$$

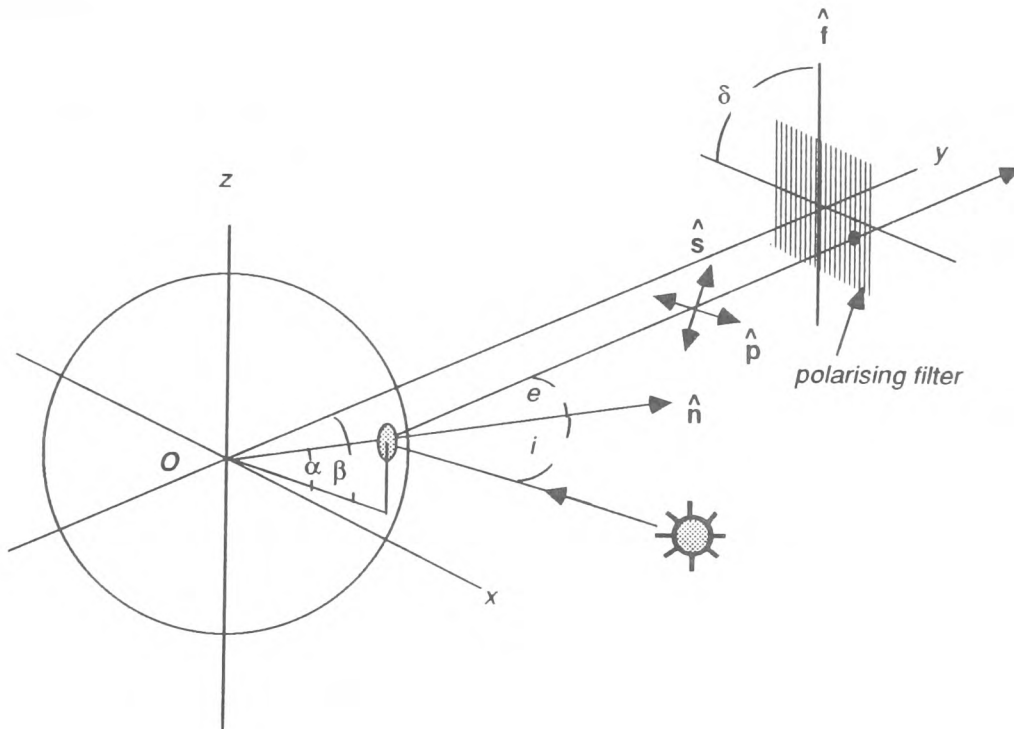


Figure 9. The radiance passing through a polarising filter.

Where:-

$\hat{\mathbf{s}}$ & $\hat{\mathbf{p}}$ are the electric field unit vectors.

A_s & A_p are the incident amplitudes of the electric field vectors $\hat{\mathbf{s}}$ & $\hat{\mathbf{p}}$.

R_s' & R_p' are the reflected amplitudes of the electric field vectors $\hat{\mathbf{s}}$ & $\hat{\mathbf{p}}$ from an absorbing surface.

$\hat{\mathbf{f}}$ is the rotation vector of the polarising filter .

This expression is derived as follows.

Fresnel's equations²⁶ are expressions for the magnitudes of the $\hat{\mathbf{s}}$ and $\hat{\mathbf{p}}$ vectors reflected from every point on the surface of the sphere. Vector algebra is used to resolve the electric vectors at the plane of the polarising filter.

The modulus of the reflected electric field resolved along $\hat{\mathbf{f}}$ are squared and summed. The values are squared because imaging devices detect the square of the reflected amplitude.²⁶ The foreshortening effect is achieved by dividing the sum by $\cos^2 e$.

The origin of the coordinate system appears at the centre of the circular projection of the sphere when viewed along $-\hat{\mathbf{j}}$, (see figure 6). The latitude and longitude (α, β) are found for any point from its coordinates with respect to the projected origin:

$$\sin \alpha = \frac{|z|}{|\zeta|}. \quad (7)$$

Because we are dealing with the unit sphere this becomes

$$\sin \alpha = |z| \quad (8)$$

and

$$\sin \beta \cos \alpha = |x|. \quad (9)$$

The distant observer sees only those rays which are reflected along $\hat{\mathbf{j}}$.

For a specular sphere, the angles of incidence, i , and reflection, e , are equal, so

$$\cos e = \hat{\mathbf{n}} \cdot \hat{\mathbf{j}} \quad (10)$$

which gives

$$\cos e = \cos \alpha \cos \beta. \quad (11)$$

The angle of refraction, r , see figure 10, in the dielectric sphere is given by Snell's law²⁶

$$\sin r = \frac{\sin i}{\mu} \quad (12)$$

where μ is the refractive index.

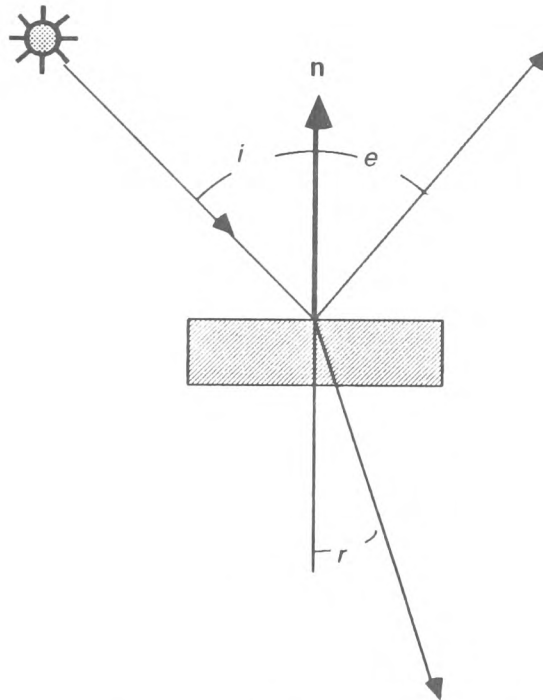


Figure 10. The angles of incidence, i , reflection, e , and refraction, r .

The plane of incidence contains the normal to the surface, $\hat{\mathbf{n}}$ and the reflected ray. The orientation of this plane can be described by the direction of its normal

$$\hat{\mathbf{s}} = \frac{\hat{\mathbf{n}} \times \hat{\mathbf{j}}}{|\hat{\mathbf{n}} \times \hat{\mathbf{j}}|} = \frac{\hat{\mathbf{n}} \times \hat{\mathbf{j}}}{|\hat{\mathbf{n}}||\hat{\mathbf{j}}| \sin i} = \frac{\hat{\mathbf{n}} \times \hat{\mathbf{j}}}{\sin i}. \quad (13)$$

When $i = 0$, $\hat{\mathbf{s}}$ is indeterminate.

The vector $\hat{\mathbf{p}}$ in the plane of incidence is perpendicular to both $\hat{\mathbf{s}}$ and $\hat{\mathbf{j}}$, and is defined in an analogous way to $\hat{\mathbf{s}}$ giving

$$\hat{\mathbf{p}} = \hat{\mathbf{s}} \times \hat{\mathbf{j}}. \quad (14)$$

If the polarising filter is rotated by an angle δ from the horizontal we can represent the rotation by a vector

$$\hat{\mathbf{f}} = \cos \delta \hat{\mathbf{i}} + \sin \delta \hat{\mathbf{k}}. \quad (15)$$

The electric vectors of the reflected ray must be resolved along $\hat{\mathbf{f}}$ to determine the proportions of $\hat{\mathbf{s}}$ and $\hat{\mathbf{p}}$ that pass through the polarising filter at any rotation, i.e.

$$\hat{\mathbf{p}} \cdot \hat{\mathbf{f}} = \frac{\cos \alpha \sin \beta \cos \delta + \sin \alpha \sin \delta}{\sin i} \quad (16)$$

and

$$\hat{\mathbf{s}} \cdot \hat{\mathbf{f}} = \frac{\sin \alpha \cos \delta - \cos \alpha \sin \beta \sin \delta}{\sin i}. \quad (17)$$

We have now calculated two components of the radiance equation. The amplitudes of the incident, A , and the reflected, R , electric vectors are given by Fresnel's equations:²⁶

$$\frac{R_p}{A_p} = \frac{\tan(i - r)}{\tan(i + r)} \quad (18)$$

and

$$\frac{R_s}{A_s} = \frac{-\sin(i-r)}{\sin(i+r)} \quad (19)$$

The minus sign in (19) indicates a change of phase on reflection. Equations (16) and (17) do not apply to the special case of normal incidence.

Fresnel's equations²⁶ for the transmitted vector amplitude are

$$\frac{T_p}{A_p} = \frac{2 \sin r \cos i}{\sin(i+r) \cos(i-r)} \quad (20)$$

and

$$\frac{T_s}{A_s} = \frac{2 \sin r \cos i}{\sin(i+r)} \quad (21)$$

Many dielectric materials are absorbing, and the effect of the absorption is estimated roughly from the boundary conditions

$$A_p + R_p = T_p \quad (22)$$

and

$$A_s + R_s = T_s \quad (23)$$

The change in the transmitted vector amplitude, T , arising from absorption changes the reflected vector amplitude, R , as follows. If T is changed by δT

$$\frac{\delta R_p}{A_p} = \frac{\mu \delta T_p}{A_p} \quad (24)$$

and

$$\frac{\delta R_s}{A_s} = \frac{\delta T_s}{A_s} \quad (25)$$

If the fraction, κ , of the transmitted ray is lost by absorption, the modified reflected amplitudes R'_p , are

$$\frac{R'_p}{A_p} = \frac{R_p}{A_p} - \frac{\delta R_p}{A_p} = \frac{R_p}{A_p} - \frac{\mu \kappa T_p}{A_p} \quad (26)$$

and

$$\frac{R'_s}{A_s} = \frac{R_s}{A_s} - \frac{\delta R_s}{A_s} = \frac{R_s}{A_s} - \frac{\kappa T_s}{A_s}. \quad (27)$$

Using equations (16), (17), (26) & (27) we obtain a relationship for the flux, Φ , when viewed through the polarising filter. The calculated flux is the square of the modulus of the electric field vectors in the $\hat{\mathbf{p}}$ and $\hat{\mathbf{s}}$ directions when resolved along $\hat{\mathbf{f}}$. The flux, Φ , emerging from the polarising filter is therefore

$$\Phi \propto \left[\left| \frac{R'_p}{A_p} \hat{\mathbf{p}} \cdot \hat{\mathbf{f}} \right|^2 + \left| \frac{R'_s}{A_s} \hat{\mathbf{s}} \cdot \hat{\mathbf{f}} \right|^2 \right].$$

From equation (3), we must divide flux by the foreshortening factor to obtain the radiance

$$L(e) \propto \left[\left| \frac{R'_p}{A_p} \hat{\mathbf{p}} \cdot \hat{\mathbf{f}} \right|^2 + \left| \frac{R'_s}{A_s} \hat{\mathbf{s}} \cdot \hat{\mathbf{f}} \right|^2 \right] \frac{1}{\cos^2 e}. \quad (28)$$

We assume that $A_s = A_p$ and they are constant over the entire surface of the sphere. We now describe how this theoretical basis can be used to test the feasibility of Polarimetric Stereo.

CHAPTER 3.

Proposal for using Polarised Light to Recognise Objects.

For our proposed method of computer vision to succeed, it must be capable of associating unique triples of intensity with surface orientations on the sphere. *Section 3.1* explains why this is not possible without modifications to the extended source. In *section 3.2* we test the method using spheres and objects computed from our model.

3.1 Obtaining a Unique Triple of Intensity.

The locus of equal angles of incidence for light waves reflected along the y-axis is a series of concentric circles whose centre is latitude/longitude (0,0). These loci also apply to constant degrees of polarisation, since this depends only on the angle of incidence for a sphere with a uniform refractive index and absorption coefficient. The degree of polarisation²⁷, P , of the reflected wave is:-

$$P = \left| \frac{R_p^2 - R_s^2}{R_p^2 + R_s^2} \right|. \quad (29)$$

The intervention of a polarising filter between the viewer and the sphere lifts the circular symmetry. We obtain a pattern which has fourfold symmetry arising from two perpendicular axes with mirror symmetry. Near the centre of the projected sphere, the pattern is a figure of eight, *see figure 11*, which relaxes to an ellipse further away from the centre. Near the edge, the pattern breaks up into a complicated series of concentric circles.

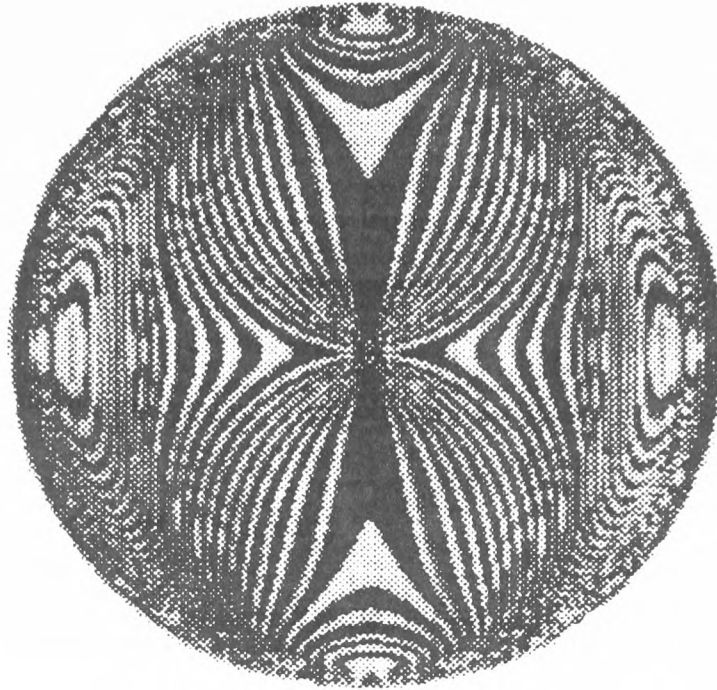


Figure 11. The polarised contours of light reflected from a sphere.

The contours of a polarised reflectance map represent areas of the image with the same degree of polarisation. A contour of constant radiance on the sphere represents infinitely many orientations, *see figure 12a* . A second image captured using a different rotation of the polarising filter will result in the two radiance contours intersecting at four points, *see figure 12b*. The four points, or orientations, will be represented by the same pair of radiance values. A third rotation of the filter will reduce the points where three contours intersect to just two, *see figure 12c* , each will correspond to the same triple of radiance values.

As the polarising filter rotates, so the pattern also rotates about latitude/longitude(0,0) since this point is the centre of the circularly symmetric degree of polarisation. A unique triple of radiance is not possible because of the mirror symmetry of the polarised reflectance map.

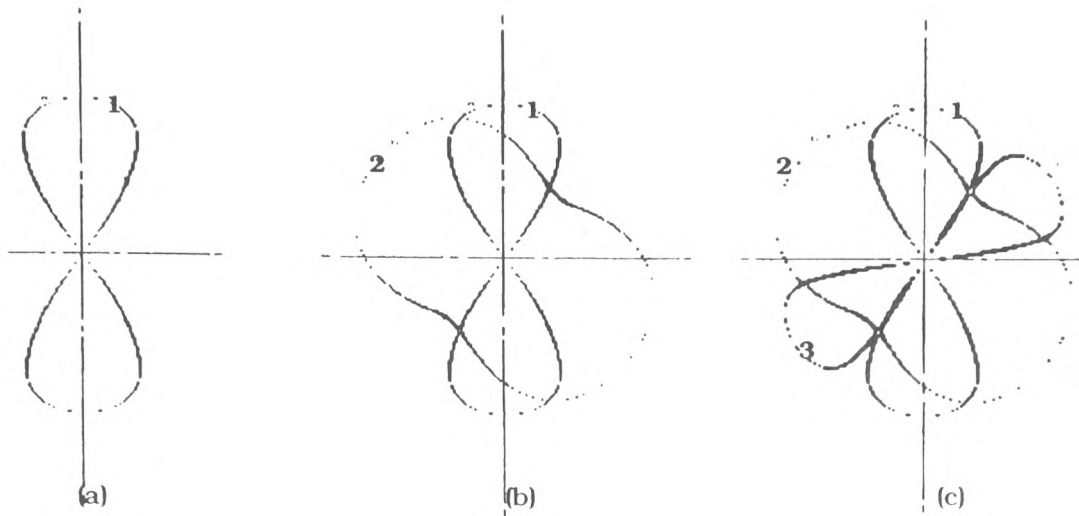


Figure 12. The intersection of contours of constant radiance using an extended source. The pixel grey-level of each contour is as follows (1) 72 (2) 98 and (3) 68. These grey-level values also apply to figure 13.

To lift this ambiguity and obtain unique triples of radiance we must lift the symmetry of the pattern by using an extended light source which produces different radiances over areas of the sphere. We built into the model this facility for the four quadrants of the image and we will describe this in greater detail in chapter 4.

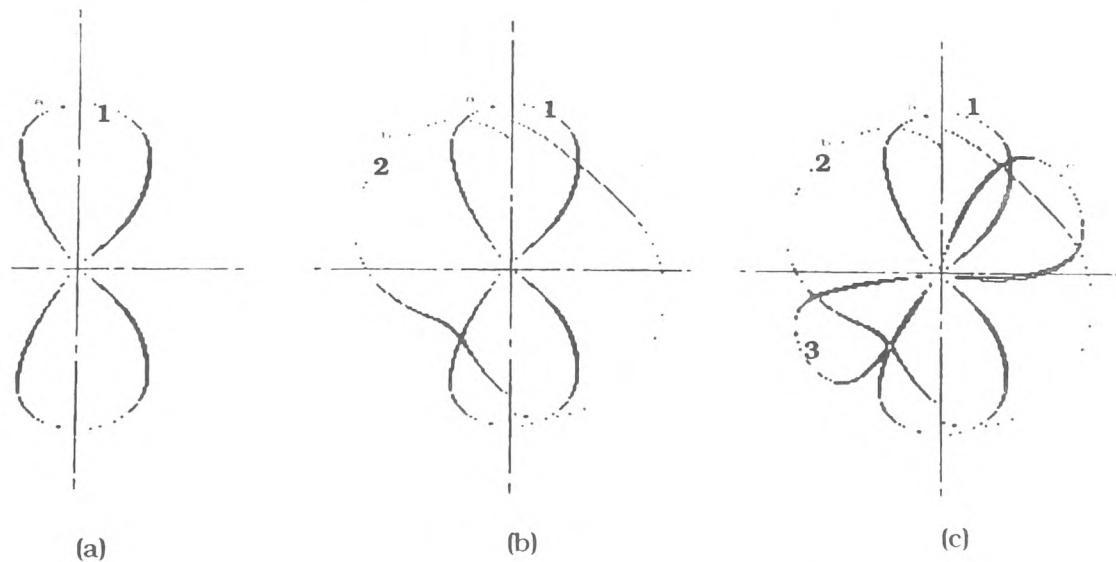


Figure 13. The intersection of contours of constant radiance using uneven illumination. We model this by giving positive x values a different absorption factor than the negative x values.

Modifying the extended source to vary the incident illumination results in asymmetric contours. These contours produce a single intersection, *see figure 13c*, enabling a unique triple of radiance to be associated with surface orientation.

In order to simplify the task we make certain assumptions. Firstly orthographic projection is assumed; for distant objects the irradiance of the image is proportional to the radiance of the object.³ Secondly illumination will remain constant and all viewed objects will have similar reflective properties. Finally the use of a fixed imaging geometry will ensure image consistency. These assumptions allow us to concentrate upon relating surface orientation to polarised radiance values.

Three images of the object are captured using a fixed camera position; for each image a different rotation of the polarising filter is used. This method produces a triple of radiance values for the corresponding pixel in each of the three images. Our modified extended source creates a variation in illumination; when combined with filter rotations, points are created where three contours of constant radiance intersect. These intersections will typically yield a unique surface orientation.

By using a sphere as a calibration object we can determine the orientation at each point on the sphere where three contours of constant radiance intersect. We will use this information to empirically create a look-up table of surface orientations, in which each cell is indexed by a triple of radiance. The look-up table is produced in a calibration stage before the system is used. Once the look-up table is created, surface orientations of viewed objects are determined simply by table look-up. The key to the successful method of Polarimetric Stereo is a reliable look-up table.

Polarimetric Stereo produces an orientation histogram of the viewed object whose cells represent surface orientation. The frequency in each cell represents the visible surface area at that orientation; this is the discrete approximation to the EGI. This

description is compared to pre-computed descriptions of the object(s) to achieve both recognition and orientation. To simplify the matching process and efficiently compute a satisfactory result, we use viewer-centred descriptions of both model and image data.

3.2 Testing the Feasibility of a Computer Vision System using a Computed Model of a Cylinder.

To test the feasibility of using polarised reflectance maps to generate object descriptions, a look-up table is built from three calculated images of the sphere, each computed with a different rotation of the polarising filter. To test the resulting look-up table, three images of a cylinder are generated. We consider the cylinder to be a solid bounded by a cylindrical surface and two parallel planes (bases) which cut the surface. The axis of a cylinder is its line of symmetry passing through the mid points of its two bases. The computed cylinder has a height of 100 pixels and a base of radius 60 pixels.

We describe the cylinder in a fixed position, where the axis of symmetry is aligned with the z-axis and the origin of the coordinate system coincident with the centre of mass of the cylinder. The x,y,z coordinates of each point on the circumference of both bases of the cylinder is found. Using a rotation matrix we rotated each coordinate about the origin relative to the selected orientation.

Lines connecting the two bases and representing the cylindrical surface of the cylinder are computed. These lines are in effect a series of planar surfaces whose grey levels must correspond to their orientation. Each visible connecting line has an orientation which relates to a point on the sphere. The radiance at this point on the computed sphere is found and used to draw the connecting line. Three computed images of the cylinder are produced in this way.

This method ensures that each group of three images of the cylinder use the same refractive index and absorption factor as the computed spheres. It also guarantees that corresponding images of the sphere and the cylinder have the same filter rotation.

Using this method we can produce test images of the cylinder at any orientation. The resulting orientation histograms are matched against the database of pre-computed cylinder histograms, as will be described in detail in *chapter 7*.

We determine the attitude of the computed model by calculating images of the cylinder at many different orientations. The average uncertainty in orientation for all tests involving the computed images of cylinders and the database of pre-computed histograms of cylinder orientations, is 5.2 degrees of arc.

CHAPTER 4.

Equipment Used

All images used to test our Polarimetric Stereo method are captured using a Sony Camcorder CCD (charge coupled device) camera. CCD cameras have two distinct advantages over inexpensive Vidicon cameras; firstly they have a linear response to radiance, and secondly the resulting images suffer less geometric distortion.¹²

Once captured, images are processed using a GEMS image processing machine. This has a DEC LSI 11/73 processor running an RT/11 operating system. The usable internal memory of the machine is 50k bytes, which includes space required for file handling input/output buffers. The GEMS software, necessary to access images, reduces the total available memory by a further 7k bytes; the limited amount of memory remaining for user programs presents many problems.

The GEMS image processing equipment consists of four 1024 * 1024 8 bit image planes; two image planes can be combined to enable storage for 1024 * 1024 16-bit integers. A displayed image consists of 512 * 512 8-bit pixels giving 256 grey levels. The problems created by the small internal memory of the processor are partially overcome by using spare image planes as additional memory. In effect these image planes enabled a substantial increase in the available RAM.

The GEMS software allows the capture of frames directly from a video camera or from a video recorder, and provides a library of Fortran subroutines to access and manipulate image data. All programs are written in Fortran 77; they are however machine dependent due to the dependence of the equipment on the GEMS software to access image data.

To produce an approximation to an extended light source, a wooden box was constructed (see figure 14); the box is a cube of side approximately 600mm. A 75mm. diameter viewing hole was cut in the centre of one face; objects are mounted on the internal face directly opposite this aperture.

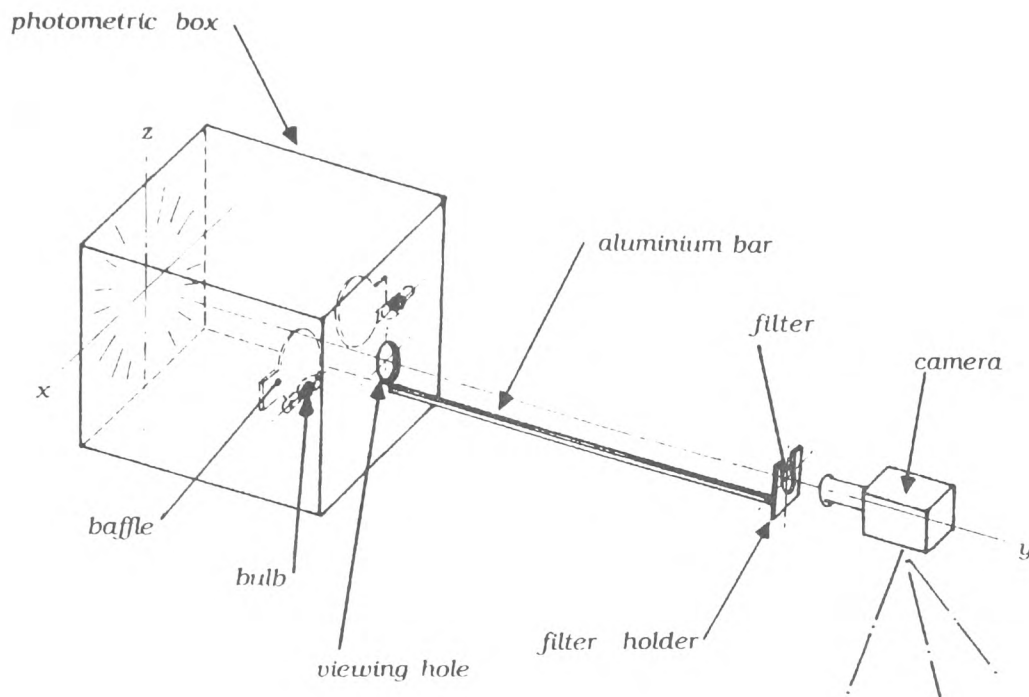


Figure 14. The experimental set-up.

Two 100 watt light bulbs are mounted inside the box, one per side, as far from the object as possible. Each light source is obscured from the object by a baffle, which is mounted directly in front of each bulb.

The interior of the box and the baffles are painted with a proprietary matt white emulsion paint to approximate a uniform extended source. Later, see chapter 9, we use four different shades of emulsion in each of four quadrants of the box, (see figure 15). Finally we reduce the number of different shades to just two, see chapter 9.

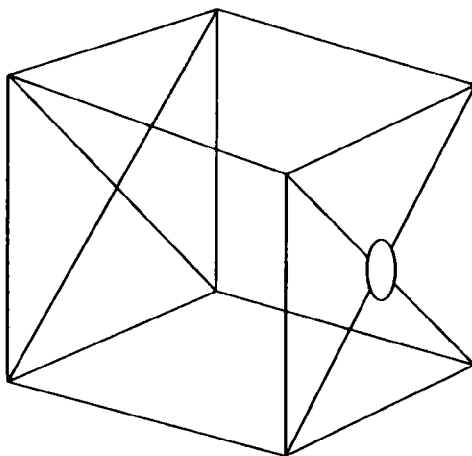


Figure 15. The four quadrants of the photometric box

A Hoya 58mm. plane polarising filter is placed between the object and the camera, (figure 14). This filter was originally mounted on the camera lens, but manual rotation of the filter is difficult to achieve without defocussing the camera. We place the filter in a homemade wooden holder; this is mounted at the end of a 1m. long aluminium bar, which is in turn attached to the front of the photometric box.

The filter is marked at selected angles to ensure consistent rotations for image capture. A piece of steel wire is attached to the filter holder and fixed in position as a calibration mark for the filter rotation.

The rotation marks on the filter are approximately 1mm. wide and the steel wire is 0.8mm. in diameter; these enable us to ensure that with moderate care we are always within 0.5 degrees of arc of a consistent rotation.

The objects are mounted on the rear vertical interior face of the box using a small hole drilled at the centre of this face. The horizontal x-axis passing through this fixing hole is marked, see figure 14. Using this axis as a datum, calibration marks at 15 degree

intervals through the full 360 degree rotation are drawn using a large accurate engineer's protractor.

The objects are mounted upon a jig in such a way that they can be rotated independently about three axes. The jig is a hinged platform which is fixed to the interior face of the photometric box. The hinged platform is aligned with the calibrated marks on the interior surface. We also use two right angled wedges. The first provides orientations at 30 degrees and 60 degrees and the other at 45 degrees. The wedges and the hinged jig are also covered with matt white emulsion.

CHAPTER 5.

Test Object Description, Representation and Orientation

A sphere is used to calibrate the system. Two objects are used to test the system namely a plastic cylinder and a plastic toy. We used several spheres of different diameters ranging from 115mm. down to 40mm. The cylinder has a diameter of 30mm. and is 50mm. in length. The toy is a compound object which consists consists of two tapered hollow cylinders a and b, *see figure 16*; the larger cylinder, b, has a planar face, c, at the narrow end to which is joined the large end of the smaller cylinder. The larger cylinder has a diameter of 25mm. and a length of 26mm., whilst the smaller, a, is 10mm in diameter, with a length of 40mm. The angle of taper of both cylinders is approximately 3.5 degrees in opposite directions.

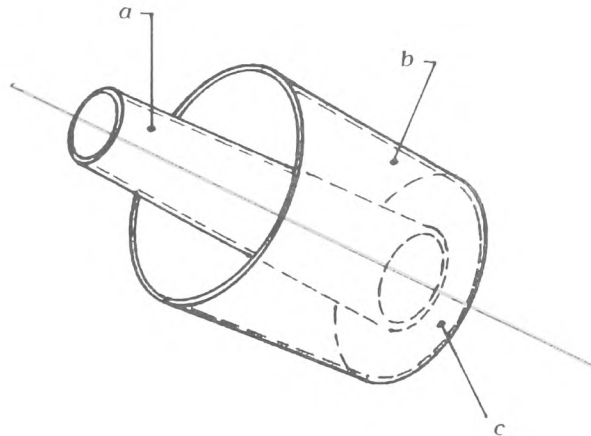


Figure 16. The toy.

Two sets of test objects are used; one set is sprayed black and the other blue. A proprietary brand of acrylic paint is used to achieve a dark uniform highly specular surface on both. Silicone polish is applied to all of the objects and when buffed their specularly is further increased.

The cylinder and toy both have planar and curved surfaces; the toy is a far more complex shape possessing both convex and concave surfaces. The EGI of both the toy and the cylinder will be fairly similar for many orientations of both objects, *see figure 17*; the taper of the compound object is the only truly distinguishing feature that differentiates between the objects. The choice of two similarly described objects will provide a severe test for any system using EGIs.

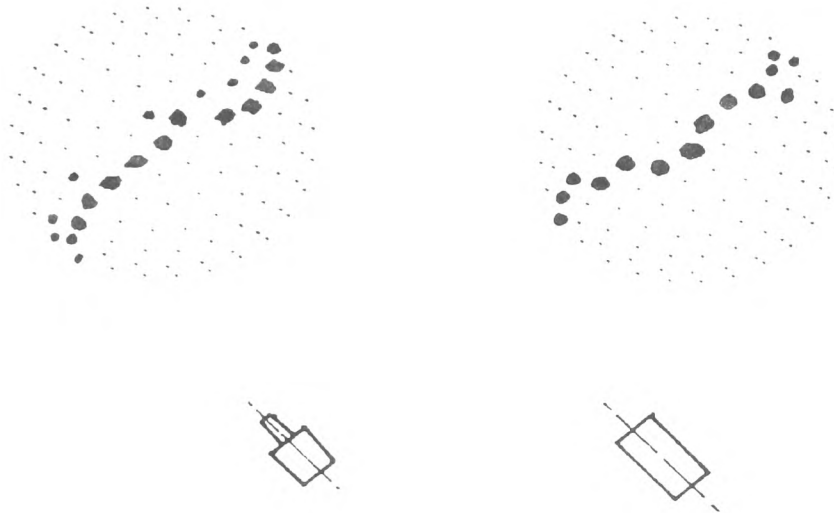


Figure 17. The EGIs of the toy and cylinder.

5.1 Object Description.

Each object is described in terms of planar surfaces so that curved surfaces are divided into a finite number of planar facets. A description is created that relates the object's surfaces to one key face. This representation gives the angular distance between this key surface and all other surfaces of the object. We position this primary face at (0,0).

A discrete representation of an object's shape such as an EGI implies that the fine detail of curved surfaces will be lost. It is therefore a coarse description, so that object descriptions are kept to essential details. *Table 1* gives a description of a cube.

Face	1	2	3	4	5	6
lat/long.	(0,0)	(0,90)	(0,180)	(0,270)	(90,?)	(-90,?)
area(mm ²)	2500	2500	2500	2500	2500	2500

Table 1. Specification of a cube of side 50mm. (?) indicates that the longitude is undefined at the N & S poles)

From this data a more comprehensive table is produced giving the angular distance between any pair of surfaces.

Face	1	2	3	4	5	6	area(mm ²)
1	0	90	180	90	90	90	2500
2	90	0	90	180	90	90	2500
3	90	180	90	90	180	0	2500
4	90	180	90	0	90	90	2500
5	90	90	90	90	0	180	2500
6	90	90	90	90	180	0	2500

Table 2. Angles between surface normals of a cube.

Using the viewer-centred approach, the shape of a cube is such that a maximum of three faces are visible. This description gives three adjacent surfaces compared to the six faces required by an object-centred description. An advantage of this method is a reduction in the necessary computation.

Using *Table 2* and knowing the sampling point orientations, we can compute descriptions of the visible object at any orientation. Surface area details are included to allow calculation of apparent and true surface areas at each surface orientation.

5.2 Object Orientation.

Both test objects are solids of revolution. The orientation of a solid of revolution is fully specified by the direction of its axis of symmetry. The orientation of both the toy and the cylinder are determined by the latitude and longitude of their axes of symmetry. An object such as a cube has three identical axes of symmetry; in such cases another measure of orientation is required, i.e. we may arbitrarily choose the orientation of the largest visible surface as the cube's orientation.

To determine the uncertainty in the object's orientation we need to ensure that the known orientation of the test object in each image is accurate to within a known tolerance. To achieve this, we place a mirror on the rear surface of our photometric box. When the camera was clearly visible in the resulting image we considered the camera and photometric box to be aligned. Vertical alignment is achieved with a spirit level.

Using GEMS software routines we devised a program to display a series of radial lines on the overlay plane of the screen. These are displayed at 45 degree orientations in the x-z plane with each line passing through the centre of the image plane, see *figure 18*. This enabled us to align accurately the calibration marks on the rear inner surface of the box with those on the image overlay plane. This served two purposes, firstly it ensured the camera's field of view is parallel to the y-axis of the photometric box and secondly it allowed the calibration marks in the box to be aligned accurately.

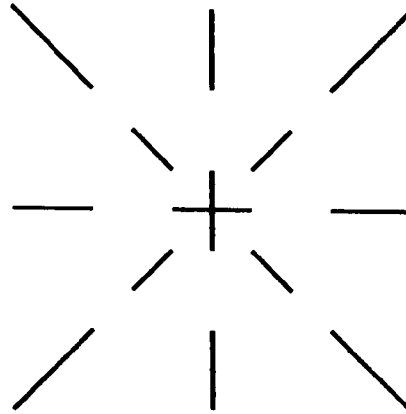


Figure 18. The radial lines on the image overlay plane are used to align accurately the calibration marks inside the photometric box.

The object is lined up with marks on the mounting bracket to ensure good mounting accuracy. Having taken much care to ensure as accurate a set-up as possible, we believe that the principal axes of our objects are oriented within an uncertainty of less than one degree of arc.

CHAPTER 6.

System Calibration

This chapter explains the various processes we require to empirically determine the surface orientations of viewed objects. A convenient method is to use a look-up table indexed by measured radiance values. The look-up table must be constructed in an operation that occurs before the recognition/orientation process; we term this process calibration. A sphere which has all possible surface orientations, is used to calibrate the system by relating surface orientation to reflected radiance.

Calibration is achieved by capturing three images of the sphere, each with a different rotation of the plane polarising filter. From these three images we relate triples of radiance values to surface orientation. The look-up table is therefore 3-Dimensional.

Orientation is specified as the latitude/longitude (α, β) pair of the point on the Gauss sphere where a vector emerged. We associate the radiance values of a particular pixel in each of the three images with its orientation. The orientation corresponding to these radiance values (L_1, L_2, L_3) is entered in the table. When the system is on-line the look-up table is used to find the orientation of a pixel given its three radiances.

The images of the sphere tell us the radiance value recorded at each pixel (x, z) on the sphere for each filter rotation. By finding the orientation at each pixel, the mapping becomes

$$(\alpha, \beta) \rightarrow (L_1, L_2, L_3).$$

Inverting this mapping enables us to determine surface orientation from a triple of observed radiance

$$(L_1, L_2, L_3) \rightarrow (\alpha, \beta).$$

Using the calibration images of the sphere we implement this mapping by constructing a 3-D look-up table indexed by image radiance values. The table contains surface orientations which correspond to the radiance values of its indices.

The look-up table is constructed as follows.

The sphere projects a circular disc in the x-z plane, knowing the centre of this disc and its radius enables us to relate each pixel on the visible disc to an orientation on the visible hemisphere. *Section 6.1* describes this operation.

Using the raw intensity values as indices to the look-up table is impractical, since the table size is the cube of the range of radiances recorded over each sphere. A further complication arises because measuring image radiance values is unreliable due to noise. A practical solution to these problems is to limit the number of values radiance can take, by quantising recorded radiances into bands; we explain this in *section 6.3*.

In principle the number of possible orientations over a visible hemisphere is infinite. The number of orientations we can calculate over the hemisphere is finite and is determined by the number of pixels occupying the visible disc. By restricting the number of recorded radiance values over the sphere, we have restricted the number of orientations that can be represented by a triple of radiance. Our viewer-centred approach also influences how we represent surface orientations on the sphere. Our solution is to use sampling points, these are described in *section 6.2*.

Having decided how we will measure radiance, represent surface orientation, and having determined the position of the calibration sphere and its radius, we can calibrate the system by building the look-up table. Our method is illustrated in *section 6.4*.

Finally it is useful to determine segmentation thresholds for later processing at this stage. *Section 6.5* describes how these are determined.

6.1 Locating the Sphere and Estimating its Radius.

We used several spheres of different dimensions in the course of our experiments. Because the sphere is not precisely positioned at the centre of the image, we need to locate the sphere and to determine the centre of its visible disk and its radius.

A single image of the sphere viewed through the plane polarising filter is captured. A Prewitt edge detector²⁸ is used to locate the edges of the sphere. Occasionally noise, caused by damage to the white painted background, would be detected as edges. To help ensure consistently accurate location of the spherical disc we smoothed the image with a Gaussian filter²⁹ before using the edge detector. The Gaussian filter operates over a 3*3 window of the image effectively removing such blemishes.

The image produced by the edge detector is thresholded to produce a binary image in which the outline of the sphere is well defined. The Prewitt operator tends to reduce areas of the image where shallow or no gradients are present to zero grey level; typically this will be the background. Areas of the image with steep gradients, e.g. the edge of the sphere, will produce the highest grey levels in the resulting image. An image is produced which is relatively straightforward to threshold into sphere and background. We compute the mean, \bar{x} , and the standard deviation, σ_i , of the resulting image; the threshold value, t_i , is found from

$$t_i = \bar{x}_i + 2\sigma_i.$$

This gives consistently good results. In practice the threshold value can take any one of a range of values and still produce a satisfactory segmentation.

The outline of the sphere is found at the poles and the north/south distance and east/west distance bisected to give an estimate of the centre and radius, τ , of the visible disc. Starting at the west pole the outline of the disc is followed; using Pythagoras' theorem the distance, τ' , from the estimated centre to the edge is found.

The distance τ' is calculated for each pixel at the edge of the sphere. We test this distance τ' against our estimated radius, τ ; a difference of ± 1 pixel is regarded as satisfactory, and τ is assumed to be the radius of the visible disc. If the average distance of τ' is outside these limits then the process is abandoned and a new set of images must be captured.

At the edge of the sphere a discrepancy in the estimated centre of more than two pixels will push the final grid circle off the sphere in at least one position. At the extremity of the sphere a single pixel can affect accuracy of sampling point orientations by about 12 degrees.

This extremely simple method gives consistently accurate results; we test the method using both real and model images of circular discs. To test the accuracy of the estimated centre, the image of the sphere is thresholded. The centroid of the resulting image is then found and compared to the estimated centre; the (x,z) coordinates of the estimated centre and the calculated centroid are always within the allowable tolerance.

6.2 Selecting the Sampling Points.

If we quantise the radiances to a limit of 16 values, we restrict the size of look-up table to 16^3 . In consequence we can only represent 16^3 orientations on the visible hemisphere with different quantised triples of radiance.

To be able to represent orientation by a quantised triple of radiance we need to divide the visible hemisphere into a discrete number of cells. The surface of the visible hemisphere is divided into patches or cells, each represented by an orientation (α, β) , which will usually be the orientation at the centre of the cell. We term these orientations sampling points.

We need to calculate the angular distance between any pixel and sampling point on the visible hemisphere. Point P represents the sample point and Q the pixel, see figure 19.

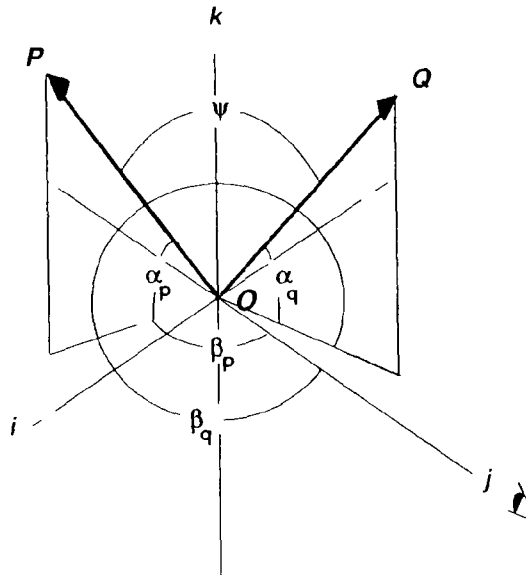


Figure 19. Two points on the visible surface of a unit sphere are separated by an angle ψ .

The angle ψ between unit vectors \vec{OP} & \vec{OQ} is given by the dot product

$$\cos \psi = \frac{\vec{OP} \cdot \vec{OQ}}{|\vec{OP}| |\vec{OQ}|} \quad (30)$$

where

$$\vec{OP} = \cos \alpha_p \sin \beta_p \hat{i} + \sin \alpha_p \sin \beta_p \hat{j} + \cos \beta_p \hat{k} \quad (31)$$

and

$$\vec{OQ} = \cos \alpha_q \sin \beta_q \hat{i} + \cos \alpha_q \cos \beta_q \hat{j} + \sin \alpha_q \hat{k}. \quad (32)$$

The dot product of these is given by

$$\begin{aligned} \vec{OP} \cdot \vec{OQ} &= \cos \alpha_p \sin \beta_p \cos \alpha_q \sin \beta_q + \\ &\quad \cos \alpha_p \cos \beta_p \cos \alpha_q \cos \beta_q + \\ &\quad \sin \alpha_p \sin \alpha_q. \end{aligned} \quad (33)$$

Equation (33) simplifies to

$$\cos \psi = \cos \alpha_p \cos \alpha_q \cos (\beta_p - \beta_q) + \sin \alpha_p \sin \alpha_q \quad (34)$$

where ψ is in the range

$$-\pi/2 \leq \psi \leq \pi/2.$$

The surface of the sphere is divided into 115 cells, each of which is represented by a single orientation. Each orientation (sampling point) represents the pixels surrounding the sampling point that are closer to it than to any other sampling point. The criterion for closeness is the angular distance between the sampling point P and the pixel Q . The angle ψ between these two points is found from *equation (34)*, and is used to find the sample point orientation for which ψ is a minimum.

The effect of quantising both radiance values and surface orientation means we will only be able to estimate surface orientation and consequently object orientation, to within an uncertainty determined by the degree of quantisation and sampling point cell size.

Our decision to use viewer-centred descriptions simplifies the matching process and removes the need to use tessellations. Rotations of one description onto the other

are unnecessary because we store viewer-centred descriptions which contain only details of the visible surfaces of the object at a variety of orientations. A fixed grid of points will suffice for both sensed and stored descriptions.

We require a set, or grid of points on the surface of the sphere that will enable us to sample the hemisphere of possible orientations. To achieve a reasonably well distributed set of points we divide the visible hemisphere into equi-spaced grid circles, see figure 20.

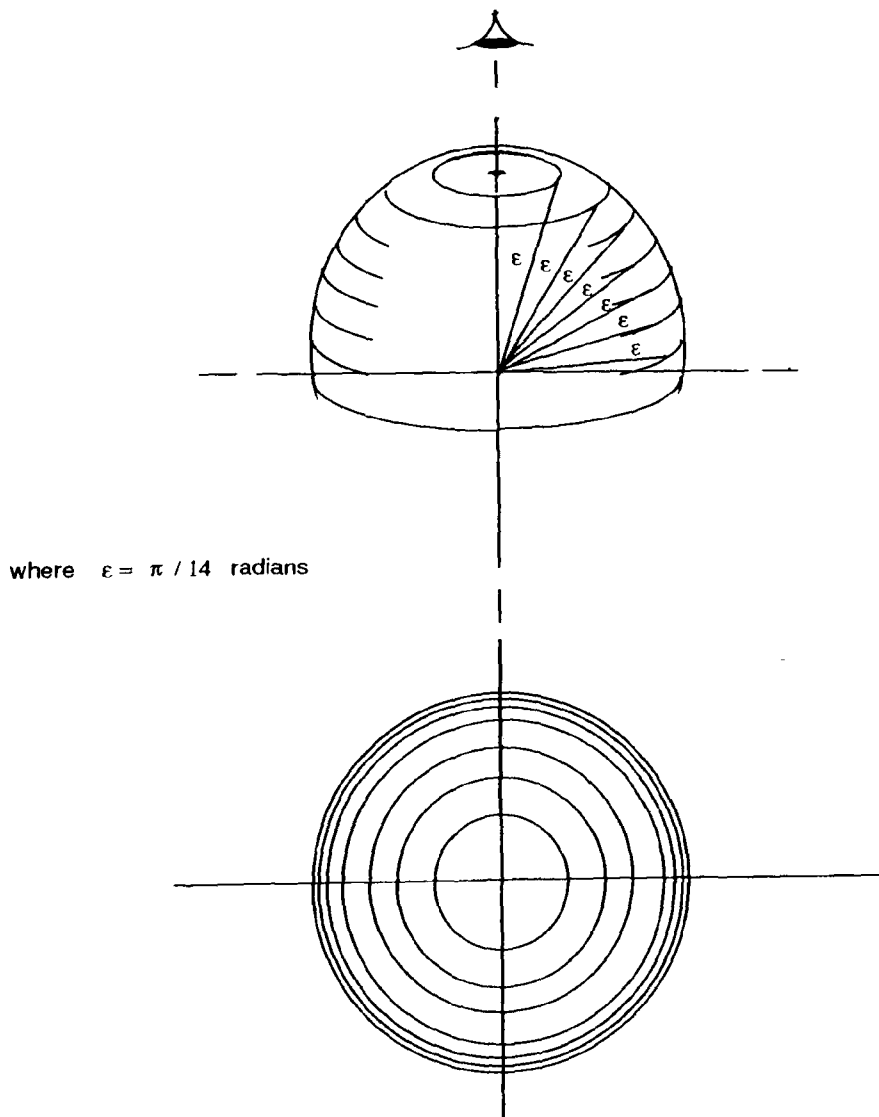


Figure 20. The grid circles.

These are spaced $\pi/14$ radians apart. Each grid circle is divided to find points on the circle that are approximately $\pi/14$ radians apart. The grid circle at $\pi/2$ radians is not visible to the camera, see figure 20.

The projected radius τ_i'' of each grid circle is

$$\tau_i'' = \tau \sin \left[\frac{i \pi}{14} \right] \quad (35)$$

where i is the grid circle 1,2,3,...,6 and τ is the estimated radius of the visible disc. Finding the circumference, c , of this circle gives us the length of arc on which to place sample points. The distance d between two points separated by the grid angle ϵ is

$$d = 2 \tau \sin \left[\frac{\epsilon}{2} \right]. \quad (36)$$

The number of points on a grid circle is c/d which gives a non-integer value. We will define the value ϵ' to be an integer value.

$$\epsilon' \leq \left\lceil \frac{c}{d} \right\rceil. \quad (37)$$

We stagger the start point on each grid circle from the x-axis to ensure variation.

This method produces 115 visible sampling points on the hemisphere, starting at the point (0,0); this is the point on the visible hemisphere where the y-axis intersects the surface of the sphere.

The sampling points are used to determine the surface orientations of viewed objects; they are also used to describe object orientation. There are however too few sampling points to describe a hemisphere of possible object attitudes, e.g. a cylinder lying horizontally on the x-z plane has an axis of symmetry oriented at $\pi/2$ radians to the y-axis. A further grid circle at $\pi/2$ radians is computed and stored. This additional

circle of points increases the total number of sampling points by 29 taking the total to 144. Points 116-144 are not visible to the viewer.

The complete set of 144 sampling points are used to estimate object attitude. Only the subset of 115 points on the visible hemisphere, (*see figure 21*), form the cells of the orientation histogram.

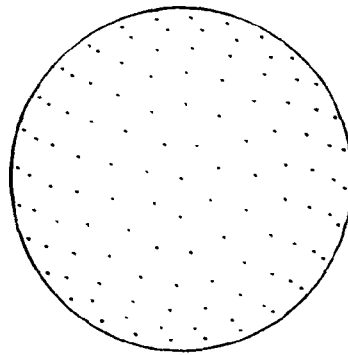


Figure 21. The grid of fixed sample points.

We use this fixed grid of sampling points to represent both the sensed image data and the stored pre-computed histograms. Both are representations of the visible object, one from the image and the other of the model.

6.3 Quantisation of Radiance Values.

In practice, the spread of recorded radiance values over an image may typically be in the grey-level range 50-250, requiring a 200^3 look-up table to hold all possible combinations. The effects of noise and fluctuations in the power supply to the source illumination, when added to the resulting images, tend to ensure that radiance measurements are unreliable sources of information. The large amount of memory

required to hold a potentially unreliable look-up table is not practical. To overcome these problems and to ensure more reliable data, we place recorded radiance values in bands. Typically we may reduce the number of grey levels to 8,12,16 or 24 by combining them into bands in each image. This reduces the size of look-up table necessary, to the cube of the number of bands in each of the three images.

We choose to quantise radiance values into 16 bands; therefore we use a 16^3 look-up table. Ideally all cells of the look-up table will contain an orientation (α, β) which will relate to a triple of radiance on the calibration sphere. Ideally an even distribution over all sixteen cells will give the best results. An uneven distribution of radiance values over the evenly spaced cell widths will greatly reduce this possibility. For example, if radiance values are distributed over ten bands in each image then the maximum number of entries we can achieve will be 10^3 . Using this method we find the distribution is extremely uneven, *see figure 22*.

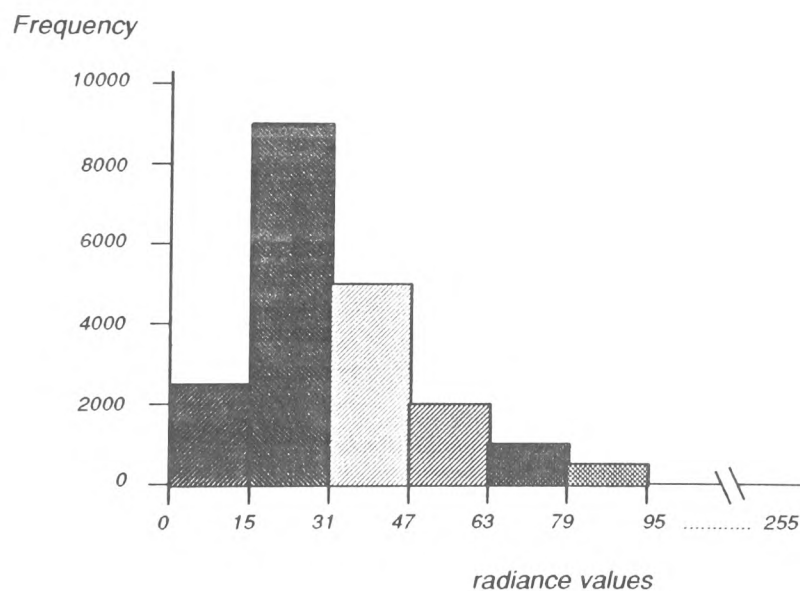


Figure 22. Distribution of radiances over the sphere.

We require the number of pixels in each class interval to be evenly distributed. The precise position of the class boundaries is determined by interpolation. Three images of the sphere are captured each with a different rotation of the polarising filter. We divide the range of possible radiance values into 32 equally spaced bands.

0-7,8-15,.....,248-255

Scanning the three images we construct histograms of the frequency and distribution of radiance values in each of the three images.

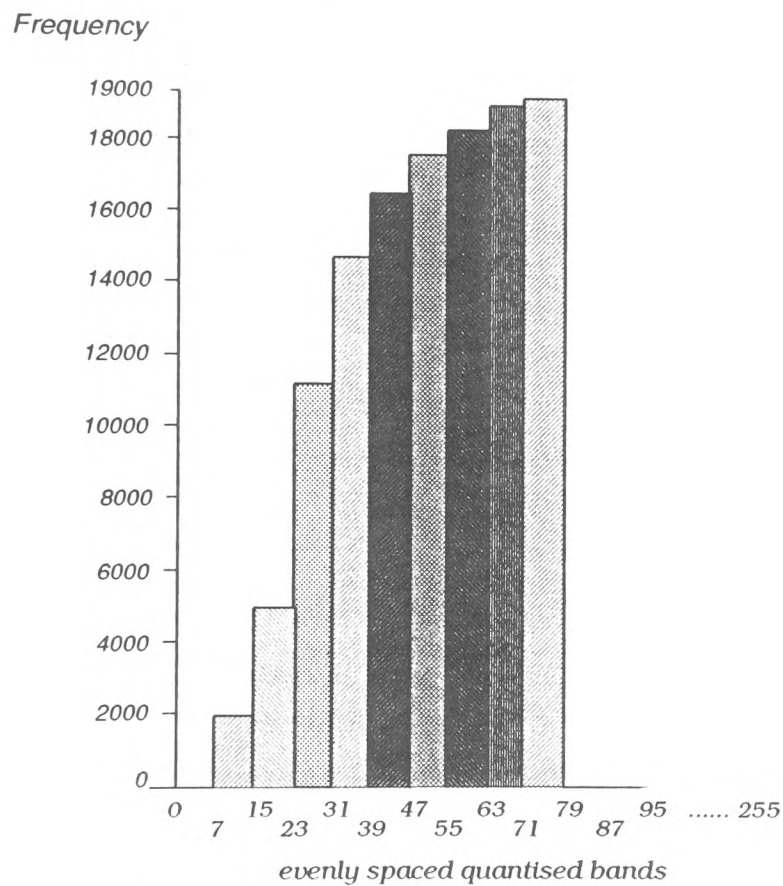


Figure 23. The cumulative sum of frequencies.

We use this information to interpolate band widths for each of the images. For each image we form the cumulative sum of grey-level frequencies, see figure 23, and perform

an inverse linear interpolation to estimate radiance band widths for each of the three images.

Ideally $\frac{1}{16}$ of total frequency in each quantised band will produce the maximum number of entries in the look-up table. By varying the quantised band widths in this manner, a more even distribution of entries in each band is achieved; see figure 24.

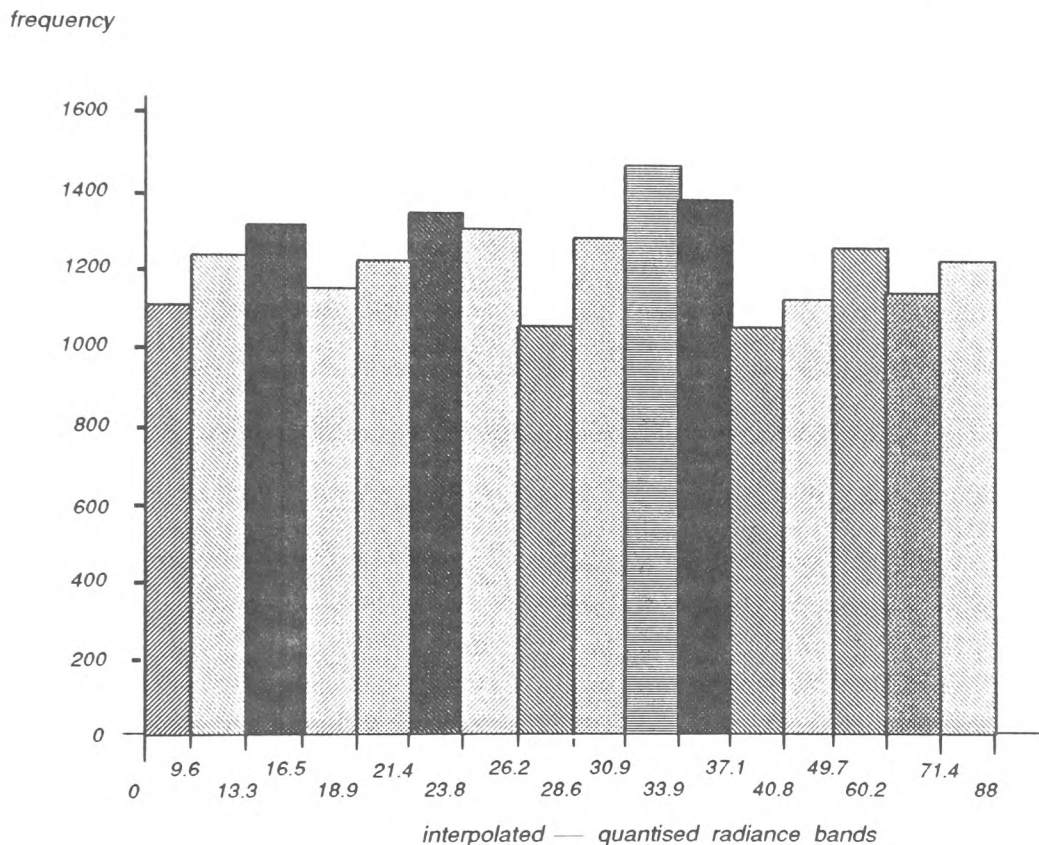


Figure 24. The radiance distribution in the interpolated band widths.

The resulting band widths are non-integer values. Later, when scanning objects, we modify the measured radiance values using a non-linear scaling function; we describe this in chapter 7. The resulting modified boundaries are also non-integer and it is these we wish to place into our interpolated quantised band widths.

This increases the number of entries in the look-up table, and in the process, improves its accuracy and reliability. When interpolated band sizes are used the number of entries in the look-up table is doubled. When combined with the non-linear scaling method the number of entries rises from approximately 600 to 2500.

Experiments with grids containing larger numbers of sampling points indicate that as the number of points placed on the surface of the sphere increases, so allocation of unique triples of quantised radiance to these points becomes more difficult. As the number of points on the visible hemisphere increases, the distance between each point and the area of each cell reduces. This has several side effects. Firstly some triples of quantised radiance will be pushed into adjacent cells. Secondly the reduced area of a sample point patch may reduce the frequency of particular triples. Allocation of radiance triples to a sample point orientation is by frequency so that both of these can affect the allocation of triples. The result may be that some sampling points will not be represented in the look-up table. It is important that all sampling points have at least one unique triple of radiance values representing them in the look-up table, i.e. we must have a complete table.

With the fixed grid of sampling points described, a complete look-up table is produced. The need to fill in any entries by any other method such as interpolation¹² is unnecessary.

The sixteen interpolated bands in each of the three images are used both for table building and subsequent processing.

6.4 Building the Look-up Table.

The three images of the sphere are always scanned in the same order and this order also applies to images of objects during subsequent processing.

Each pixel on the visible disc is scanned and its orientation (α_q, β_q) on the sphere is found using

$$\sin \alpha = \frac{|z|}{\tau} \quad (38)$$

and

$$\sin \beta_q = \frac{|x|}{\tau \cos \alpha_q} \quad (39)$$

where τ , is the estimated radius of the visible disc, and (x, z) are the coordinates of the pixel on the 2-D image plane. Knowing the orientation of each pixel (x, z) in terms of latitude/longitude (α_q, β_q) and the orientation of each sample point (α_p, β_p) we can compute the sample point closest to each pixel using *equation (34)*. The closest sampling point being the one for which the distance ψ is a minimum.

The process of building the look-up table begins at sampling point $(0,0)$. The quantised radiance in each of the three images are stored in a list of possible triples for $(0,0)$. All the surrounding pixels, which are closer to $(0,0)$ than any other sample point are examined and their triples added to the list of possible values. If a triple is already present then its count is incremented. In this way a histogram of frequency of occurrence of each triple of radiance is formed.

The process is repeated for each visible sampling point in turn so that we produce a total of 115 histograms of radiance triples, one for each of the visible sampling points.

We also need to calculate the visible and true area. Each pixel represents a foreshortened area, *see figure 25*, which is related to the true area as follows, i.e.

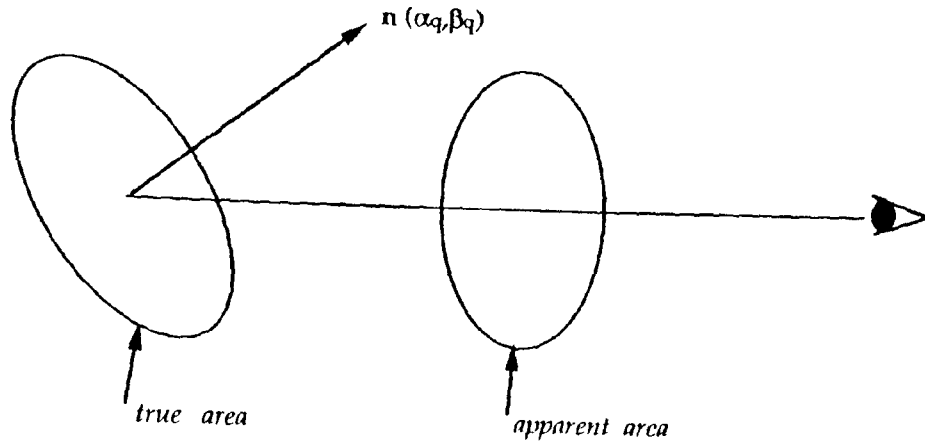


Figure 25. Diagram illustrating the foreshortening factor

$$\text{true area} = \frac{\text{apparent area}}{\cos \alpha_q \cos \beta_q} \quad (40)$$

where (α_q, β_q) are the latitude and longitude of pixel q .

The result is a series of histograms whose cells represent a quantised triple of radiance and whose frequency corresponds to the true surface area represented by that triple on the sampling point.

We are now in a position to use these data to build the look-up table. The histograms are repeatedly scanned until all recorded triples of radiance are allocated to sampling points. Allocation is by largest area first; this is represented by the frequency in the cells of the histograms. The largest frequency of a particular triple is chosen to represent the sampling point belonging to that histogram in the 3-D table. All other occurrences of this triple in the remaining 114 histograms are deleted. When a triple is

deleted its frequency is lost. In practice this does not create any problems, but could be improved in the future. This method ensures that the mapping of each triple is unique.

6.5 Segmentation Thresholds.

Objects are segmented from background by thresholds. We use information from the three images of the sphere to determine both an upper and lower limit to threshold radiance values.

Our coloured specular spheres are viewed against a white background, and the background produces the highest radiance values in each of the images. The largest reflected radiance values on the sphere occur at grazing angles where the reflected light is unpolarised and the foreshortened area tends to zero.

The surfaces of objects whose normals are nearly parallel to the x-z plane have the highest radiance values. On the sphere this occurs with pixels at the boundary of the visible disc. Many orientations of objects will result in the highest radiance values of the object occurring where the object and the white background merge making segmentation less obvious.

We find that the last but one pixel on the sphere subtends an angle of approximately 78-82 degrees. The largest radiance value at this angle in the three images of the sphere are found. These three values become the upper threshold for each image in all future images of objects.

The lowest radiance values will occur when surfaces are not directly illuminated by the extended source, *see figure 26*.

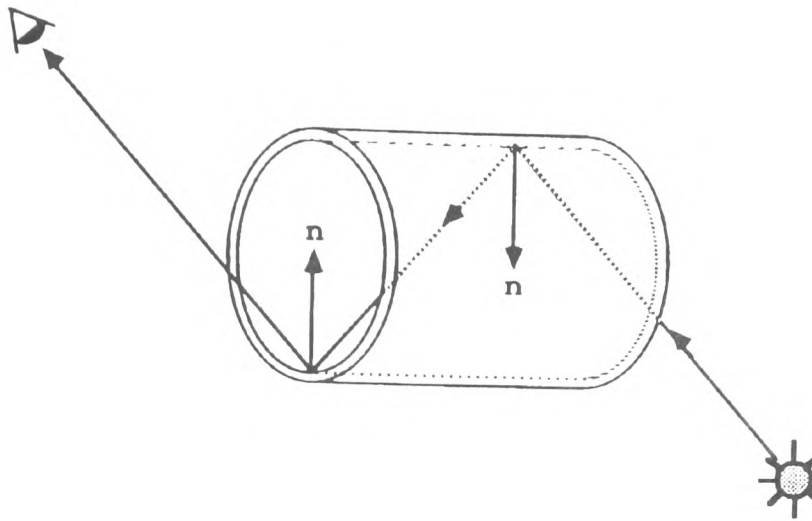


Figure 26. An interreflected surface.

Typically this may occur with concave surfaces, *see figure 27*. In such cases the extended light source will ensure the same occurrence in all three images regardless of the rotation of the polarising filter. The images of the sphere are scanned for the lowest radiance value in each of the three images; this gives us the lower bounds of our thresholds, one for each image.

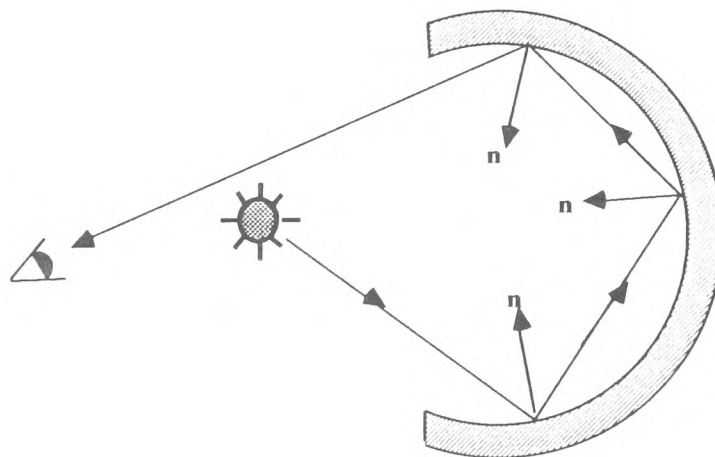


Figure 27. Rays reflected from a concave surface.

Both low and high radiance thresholds are determined during calibration, from examination of the three images of the sphere. The effects of noise can distort these values; we use the average of adjacent pixels to determine the thresholds. This increases the lower values and tends to reduce the higher values. The result is a robust and consistent segmentation.

CHAPTER 7.

The Database of Viewer-Centred Descriptions

Polarimetric Stereo produces a description of the sensed object in the form of an orientation histogram. To determine the viewed object's identity and orientation we need to match the sensed orientation histogram with model data. Matching like descriptions will simplify the process; we therefore compute model descriptions in the form of orientation histograms. The database consists of a series of pre-computed orientation histograms; each histogram is a model description of the object visible to the viewer (camera) at a specific orientation. We refer to these as viewer-centred descriptions.

These model descriptions are pre-computed and stored as 16-bit integers by combining two 8-bit image planes. The algorithm we use to build each description is described in *section 7.1*.

We use features of the object's 2-D projection to help reduce later comparisons. We compute the principal axis, the average foreshortening factor and the area of self-shadowed surface for each stored histogram. These are placed in a header record and stored as part of the computed orientation histogram in the database. *Section 7.2* describes these features and our methods of computation. The layout and contents of each stored histogram are defined in *section 7.3*.

The frequency in each cell of the sensed orientation histogram corresponds to the number of pixels of the visible object at that orientation, while the model histogram is produced from a table that specifies surface area in terms of square millimetres. In *section 7.4* we bring both sensed and model descriptions into correspondence.

To retrieve stored orientation histograms efficiently we need to store model histograms in a way that allows selection of relevant records for possible comparison. We explain how we achieve this in *section 7.5*. The algorithm we use to retrieve records determines how they are inserted into the image plane. How we place records and pointers in the database for selective retrieval is stated in *section 7.6*. To illustrate the algorithm, in *section 7.7* we insert an example histogram into the database.

The range of integers available for addressing records is in the range $-2^{15} \dots +2^{15}-1$. In *section 7.8*, our method for coping with negative addresses is described.

Marks on the surface of viewed objects may produce unrecognised pixels. Non-linear scaling of measured radiance values, described in *section 7.9*, partially overcomes this problem.

7.1 Computing the Viewer-Centred Model Descriptions.

We have divided our visible hemisphere into a discrete number of patches; the orientation of each patch is represented by a normal to the surface at the corresponding sampling point.

The orientation of an object such as a cylinder can be described by determining the orientation of one of its bases. We can compute the database of orientation histograms for the cylinder by orienting this base over the hemisphere of sampling points; the sampling points perpendicular to the viewer axis are included to ensure a complete description.

The toy has two different bases each producing very different descriptions of the visible object. We must orient both bases over the visible hemisphere to compute a

database for this object; we must also be able to determine which end of the toy we are viewing.

Each surface patch of the hemisphere contains an infinite number of surface orientations, *see figure 28*. These are represented by a single orientation, the normal to the surface at the sampling point. The orientation of randomly oriented objects or their surfaces will rarely coincide with the surface orientations at sampling points. As an object's orientation is varied over a sampling point patch, so visible surfaces may be translated from one surface patch to another, *see figure 29*, forming different orientation histograms in the process.

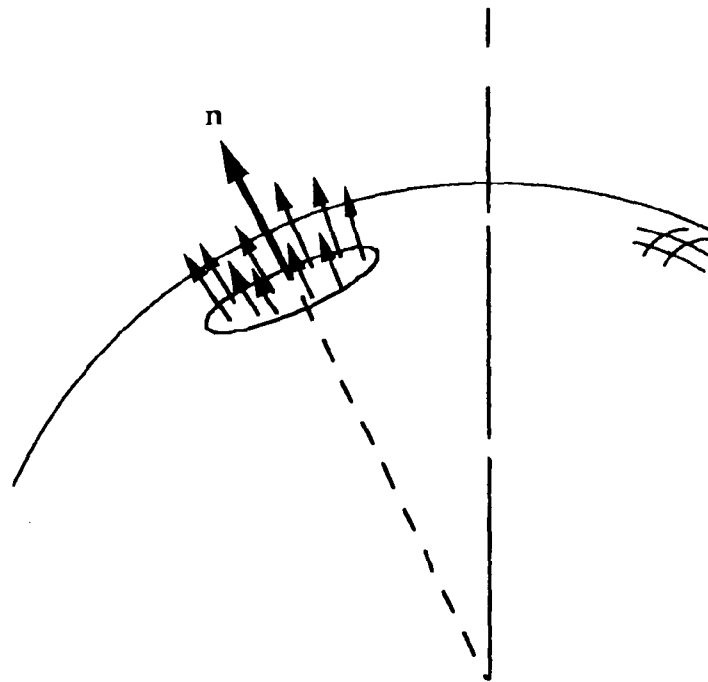
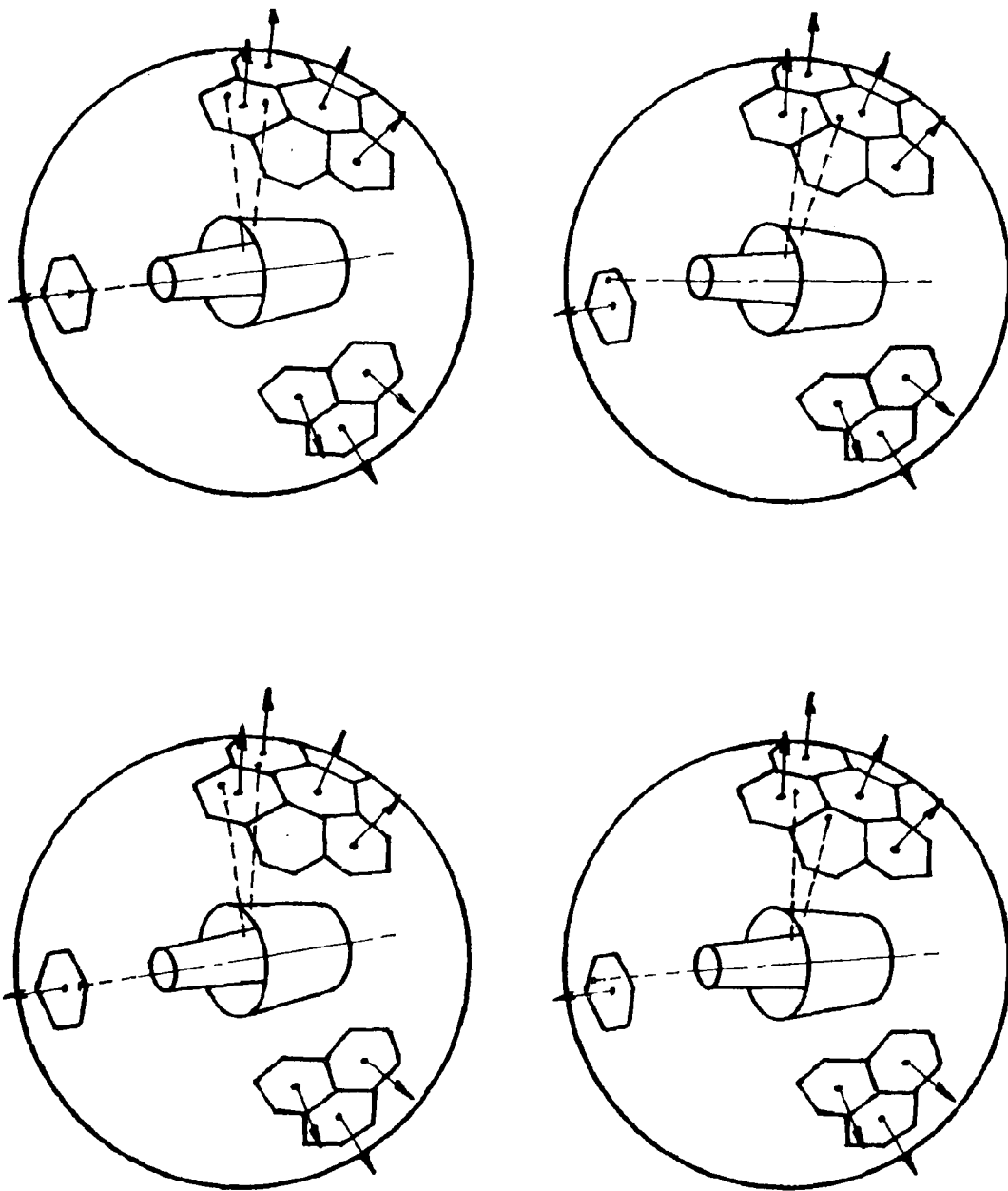


Figure 28. Each sampling point patch on the surface of the sphere contains infinitely many different surface orientations.



Figures 29. Slight variations in the orientation of an object can push some surfaces into adjacent sampling point cells. In the process different orientation histograms are formed; the orientation of the object is still represented by the same surface normal.

Image noise and look-up table errors also contribute to variations in the sensed orientation histogram. These tend to push frequencies into adjacent cells of the orientation histogram.

We wish to sample the set of possible orientation histograms for each sampling point to produce a database of model descriptions. This will enable the matching of sensed and model data to provide both recognition and orientation capabilities. We have devised a method which attempts to deal with all of these problems. To reproduce model orientation histograms which cope with the problems stated we use a threshold value, t_a , in conjunction with *equation (34)* to determine which of the object's surfaces will be visible at each orientation and the sampling points on which they are oriented.

Typically 10 - 20 orientation histograms of the object are built at each sampling point. The number will depend upon the shape and complexity of the object. A further factor is the size of threshold t_a ; the larger the value of t_a , the greater the possible variation in computed surface orientations and consequently the larger the number of computed histograms. We experimented and found that a value of $t_a \approx 15\%$ of the angular distance from *table 2* gave good results.

Each object in the database is computed in the same way. A description, *table 1*, is produced and *table 2* is computed from it to provide the surface data necessary to build a database of orientation histograms.

Our method of computing each database description is presented as a pseudo code algorithm; comments are labeled as follows /* .

```

/ *** main *** /

for all major_object_surfaces mf do    /* see below
  for all sample_points sp do          /* sp = 1,144
    align normals of mf and sp
    for all object_surfaces k do       /*i.e. k=1,58(toy)
      if (k  $\neq$  mf) do
         $\theta$  = angle between normals of object surfaces mf and k /* table2
        for all visible sample points sv do    /* sv = 1,115
          calculate  $\psi$           /*angle between normals of sp and sv
          if ( $\theta - t_a \leq \psi \leq \theta + t_a$ ) then
            push sv onto basestack
          endif
          previous_stack = basestack
          last_surface = k
        endfor
      endif
    min = 1, max = No.of object surfaces
    find_all_vis_surfaces(min,max,stackptr)

/* We have now computed a series of stacks many of which are empty, these represent
surfaces which are invisible at this orientation (the axis of symmetry is the normal to
sampling point sp). We ignore these empty stacks. The number of non-empty stacks will not
be greater than the number of visible surfaces at this orientation. (The effects illustrated by
figure 29 will determine the number of cells of the orientation histogram that contain
non-zero frequencies) */

/* The computed orientation histogram for this object orientation will have frequencies in
the cell numbers corresponding to the values at the top of each non-empty stack. Stackptr tells
us which surface of the object each stack corresponds to. The frequency in each cell is the
foreshortened area found from the area in  $\text{mm}^2$  in table 2 and the orientation  $(\alpha_p, \beta_p)$  of the
normal to sampling point sp. */

```

```

while basestack not empty
  check_histo_for_duplication
  if duplicate then
    delete current histogram
  else
    compute_foreshortened_frequency
    compute_object_features /* see section 7.2
    normalise_histogram /*see section 7.4
    write_histogram_to_image_plane /* see section 7.5
  endif
  empty = false
  ptr = stackptr
  while stack(ptr) not empty
    pop_stack(ptr)
    if stack(ptr) empty then
      ptr = ptr - 1
    else
      min = top_of_stack(ptr)
      call find_all_vis_faces(min,max,ptr)
      empty = true
    endif
  endwhile
endwhile
endfor
endfor
endfor

```

/ find all sampling points on which visible surfaces of the object can reside */*

```

proc find_all_vis_surfaces(min,max,ptr)
for all object_surfaces j do /*j = min,max
  while (j ≠ mf) or (j ≠ k) do
    stackptr = j
     $\psi_a$  = angle between normals of object surfaces mf and j /*table 2
     $\psi_b$  = angle between normals of object surfaces last_surface and j /* table 2
    for each visible_sample_point sk do /*sk = 1,115
      calculate  $\theta_a$  /*angle between normals of mf and sk

```

```

    calculate  $\theta_b$     /*angle between normals of sk and top of previous_stack
    if  $(\theta_a - t_a \leq \psi_a \leq \theta_a + t_a)$  and  $(\theta_b - t_b \leq \psi_b \leq \theta_b + t_b)$  then
        push sk onto stack(stackptr)
    endif
    previous_stack = stack(stackptr)
    last_surface = stackptr
endfor
endwhile
endfor

/** End of Algorithm */

```

Many objects display a degree of symmetry. This symmetry can be utilised to reduce the amount of computation necessary to build a database. As an example the cylinder database can be computed by specifying one of its bases as the major face, *mf*, in the above algorithm. The toy has two such major surfaces, i.e. its bases.

Our test objects are solids of revolution. For certain orientations of the toy, some of its visible surfaces may be partially occluded and will therefore provide a reduced contribution to the frequency of the corresponding cell of the orientation histogram. The contribution will however be constant for each sampling point on a particular grid circle. We include this information when computing the model orientation histograms. We pre-calculated the proportion of each surface visible on each of the six grid circles. This information is included in our program to calculate the orientation histograms of our toy.

Occasionally due to object symmetry duplicates will be produced. Our method of histogram storage, see section 7.5, indicates possible duplicates and enables selective retrieval of histograms to find and subsequently delete duplicate records.

The method will produce many histograms which vary little from previously computed histograms. We examined successive histograms computed in this manner and found that the two histograms may be identical in all but one cell. Two implications follow. Firstly, it would be generally impractical to store all possible combinations of orientation histograms. Secondly, as sensing errors ensure that matching can only be achieved with an element of uncertainty, storing the complete set would not guarantee greater accuracy. We used similarity as a basis for rejecting some histograms and as a result reduce the number of histograms stored to a manageable level.

We keep a copy of the last orientation histogram to be stored in the database. As each successive histogram is computed it is compared to the stored copy. If more than half of the current histogram cells are present in the copy, we reject the current version and compute the next; only cells containing a frequency are checked. This method produces orientation histograms at each sampling point and provides a varied subset of stored orientation histograms.

The major disadvantage to this method will occur with objects possessing a great many visible surfaces. These will present problems for such a method because the orientation histogram can only represent 115 visible surfaces. The number of different orientation histograms it is possible to compute will also increase significantly with such objects. These criticisms will apply to both object-centred and viewer-centred methods.

7.2 Features of the Model Object.

So far we have computed prototypical data about the orientation of visible surfaces and their apparent areas. We also compute information about both the model descriptions and the visible object which, when matching, are used to reduce the

number of comparisons required to achieve a match. When comparing stored descriptions with sensed descriptions it is more efficient to pre-compute information about each model histogram and store it in a header record as part of the stored histogram.

We use features of the visible object to reduce the number of comparisons needed with stored descriptions to achieve a match. Computing the same features for the sensed object will allow an initial comparison of the two sets; we can reject many stored descriptions on the basis of this initial comparison.

The average foreshortening factor, h , of the visible object (this is derived from equation 40) is defined as

$$h = \frac{\text{apparent surface area}}{\text{true surface area}} = \frac{A_a}{A_t}. \quad (41)$$

The average foreshortening factor, see figure 30, is sometimes referred to as the centre of visible mass.²⁰

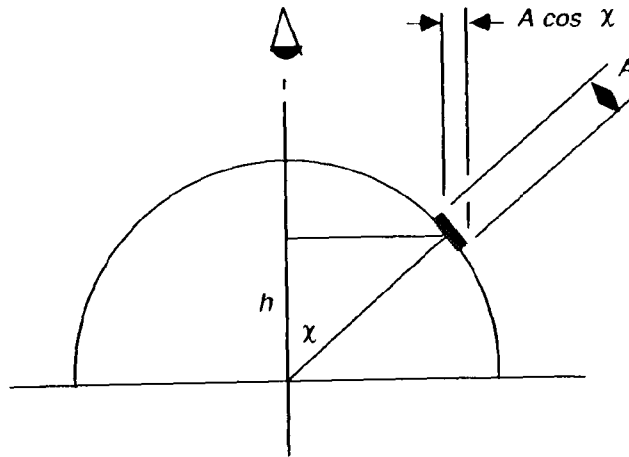


Figure 30. Diagram of the average foreshortening factor, h .

The frequency in each cell of the orientation histogram is summed to give the apparent surface area, A_a . The true surface area of the object, A_t is found from

$$A_t = \sum_{i=1}^{115} \left[\frac{\text{frequency}_i}{\cos \alpha_i \cos \beta_i} \right] \quad (42)$$

where (α_i, β_i) is the orientation of cell i .

This value will vary with the orientation of the object; it is therefore of use in determining object attitude. If our sensed object is in a similar attitude to that of the computed model description then both centres of mass will be comparable. The resulting value is not unique but it is of use in helping reject descriptions that represent very different attitudes to that of the sensed object.

A further useful feature is the orientation of the principal axis, p , of the 2-D projection of the visible object. Our test objects are solids of revolution whose attitude is determined by the orientation of their axes of symmetry. When computing each model orientation histogram, the object's axis of symmetry is aligned with the normal to each sampling point in turn. The orientation (α_p, β_p) of this normal will determine the orientation of the object's axis of symmetry and hence its orientation for each model description. By projecting this axis of symmetry onto the 2-D image plane, we can find the orientation of the principal axis, p , from

$$\tan p = \frac{\sin \alpha_p}{\cos \alpha_p \sin \beta_p} \quad (43)$$

A last feature is computed which depends upon the surface type, shape and orientation. Specular surfaces reflect light in a plane that contains both normal to the surface and the incident ray. When visible surfaces are occluded from the incident ray, any incident light is a result of reflection from other surfaces, *see figure 31*. Dark specular

surfaces absorb large amounts of incident light; with each reflection the resulting radiance becomes less intense.

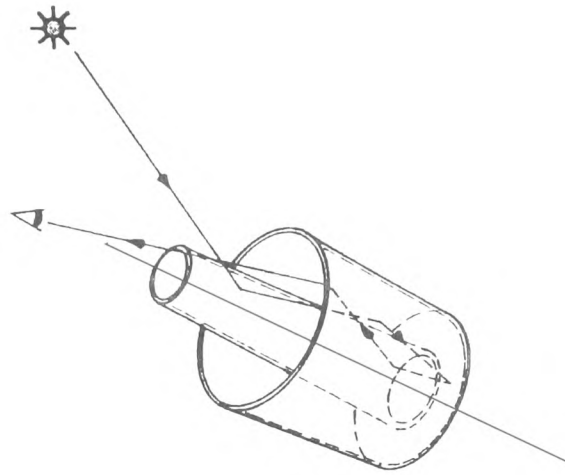


Figure 31. Self-shadowed surfaces.

Unpolarised light incident upon a dielectric surface becomes plane polarised. Further reflection of the reflected plane polarised light changes the degree of polarisation.

Both of these factors contribute to radiance values which produce spurious estimates of orientation. We take this potentially unreliable information and convert it into a useful form as follows. Certain objects such as our toy have non-convex surfaces; when visible these will produce incorrect estimates of orientation. The lower threshold values computed during the calibration stage are used to determine self-shadowed surfaces; we increment a count of pixels each time these lower thresholds are invoked. The resulting total represents a visible surface area; this area will vary with orientation for objects such as our toy. In the case of convex objects it will always be zero.

We can predict where and when this will occur and its apparent area. From this information we can pre-compute the apparent area of occurrence in each of the stored

descriptions and store the value in each header record. This information is used as part of the matching process.

7.3 The Stored Orientation Histogram.

The database consists of a series of orientation histograms. These are stored in the database as a series of 1-D arrays. Each array consists of a header record followed by the orientation histogram.

Every header record has six fields

1	2	3	4	5	6
No. of visible surfaces	area of self shadowed surface	p	α, β	h	name

Figure 32. The header record portion of the orientation histogram.

The stored histograms are variable length records, whose length depends upon the number of object surfaces visible at that orientation. The number of visible surfaces is placed in position one of this record. The length of each stored histogram, l , is found by

$$l = (\text{header}(1) * 2) + 6$$

where l is an array of 16-bit pixels.

Positions 2,...,5 contain information about the object that are determined by its orientation. Positions 2,3 & 5 relate to the 2-D projection; these are collectively termed features of the visible object and are particularly useful because their values are determined by the object's orientation.

The contents of fields 2,3 & 5 are as follows. The second field contains the visible area of the object in pixels, which is not directly illuminated or self-shadowed. We use this feature to improve the match.

Position three holds the orientation of the principal axis, p , of the 2-D projection of the object. The orientation of the principal axis is expressed in degrees of rotation about the centroid from the x-axis on the positive z-axis, see figure 33 below. This helps to select model records for matching.

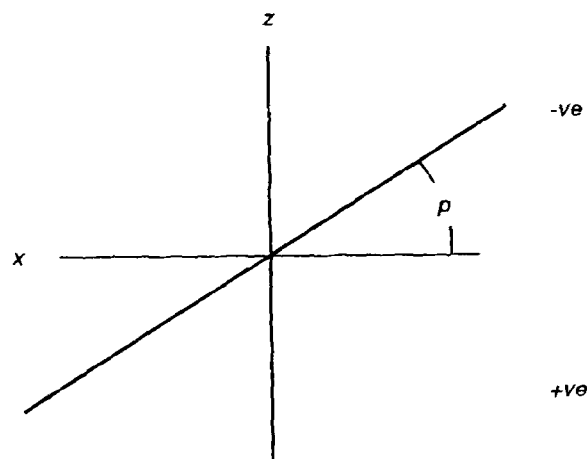


Figure 33. The orientation of the principal axis of the 2-D projection.

Position five holds the ratio of visible area to true area of the object at that orientation; this is the average foreshortening factor, h .

Position four contains the orientation of the object. This is determined by the sampling point whose normal is parallel to the axis of symmetry of the object and is determined when calculating the model orientation histogram.

The name of the object is held in field six; this is an integer code with which we identify the model. The toy has two identifying codes, which also specify which end of the object is visible.

The remainder of each record is the calculated orientation histogram. Each visible surface of the object is represented by two pieces of information; the sample point number corresponding to the normal to the surface (α_p, β_p) and the frequency representing the apparent surface area at that orientation.

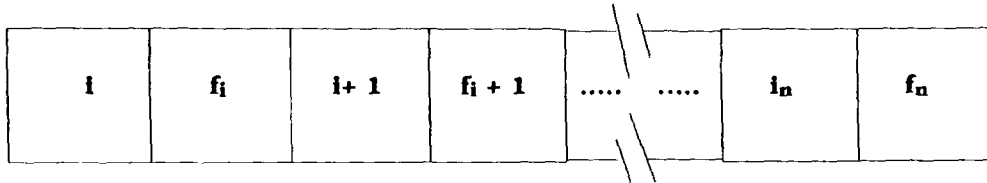


Figure 34. The cells of the orientation histogram; i is the sample point number and f_i is the frequency of cell i .

7.4 Normalising of the Database of Descriptions.

Both model and sensed descriptions are derived using different measurements of area. The visible area in mm^2 of each model surface is found from table 2. The sensed orientation histogram describes visible surface area in pixels, i.e. the frequency in each cell.

To match sensed descriptions with model descriptions we must use the same type of information. The cells of both histograms are identical, only the description of surface area (frequency in each cell) varies. The area of each visible surface can be represented by the proportion of the total visible area of the object it occupies. We achieve this by normalisation; each histogram is normalised by dividing the frequency in each cell by the sum of these frequencies.

Normalisation improves matching because we compare two sets of data in the identical format. It is however scale invariant because two objects identical in all but

scale become indistinguishable from each other; their proportions at each surface orientation will be identical.

7.5 Storing the Database of Descriptions.

Object-centred methods require four actions. First, the model description is rotated into the required position. Second, the normal of each surface is projected onto the Gaussian sphere. Third, the model orientation histogram is calculated from the resulting Gaussian Image. Finally the sensed orientation histogram is compared to the model histogram.

This procedure is performed for each different rotation (orientation) of the model. The overheads of such a method are expensive,¹³ and furthermore over a period of time the same data are re-computed many times. A viewer-centred approach using a database of orientation histograms which has been pre-computed and stored in a once only operation is a more efficient method.

When this viewer-centred method is combined with a novel method of storage and retrieval, based on embedding pointers to records in the database, the process becomes a computationally efficient method.

The viewer-centred descriptions are stored on a 16-bit image plane, *see chapter 4*. This provides RAM storage for $1024 * 1024$ 16-bit integers. Our ability to address this amount of data is limited by the range of 16-bit integers, $-2^{15} \dots +2^{15}-1$. We overcome this limitation by using blocks of the image rather than individual cells or pixels. Using a block size of 16 cells increases our addressing range to $-2^{19} \dots +2^{19}-1$; this allows us to address $1024 * 1024$ 16 bit integers in blocks of 16. We are now able to use the entire image plane to store our database; the number of possible separate addresses being limited to a maximum of 2^{16} .

As each new description is computed it is inserted into the database at the next convenient contiguous empty location; i.e. the descriptions are stored by order of computation only. An efficient search of the database will require a selective retrieval of the stored descriptions.

There are two main reasons for requiring a selective retrieval of records. Firstly, it may be possible to compute the same histogram more than once from different sampling points; storing duplicate records is inefficient. This is particularly true for objects with a high degree of symmetry. To locate duplicate records efficiently, we must avoid a computationally expensive global search and comparison of the existing database. Secondly and most importantly, when seeking stored histograms for matching during subsequent processing, we wish to search only a relevant subset of the available records in the database.

We limit search space by embedding pointers to stored histograms in the image plane containing the database. A decision on which pointers to read and follow is obtained from the cell frequencies of the sensed orientation histogram. The same information in the pre-computed histograms is used to determine where pointers are placed in the image plane; these help direct later searches. *Section 7.7* illustrates a histogram and its insertion into the database.

One cell of each pre-computed orientation histogram will have the largest frequency. Where the largest frequency is shared by more than one cell, the first occurrence is selected. To select the row, i , of the image in which the first pointer is placed, we use the sample point number i , where $i = 1, 2, \dots, 115$, corresponding to the orientation of this cell.

To find the location in the row, we examine each cell of the computed histogram and determine those containing a non-zero frequency. The sampling point number

corresponding to each of these is squared and summed; the result is hashed to provide the location, *a* in *figure 35*, as follows

$$Location = (\sum cell\ number_i^2) \text{ MOD } 1024 \quad (44)$$

where MOD means modulo division.

This method of determining where the initial pointer to each stored histogram is placed, helps direct later searches. Each histogram whose initial pointer is stored in row *i*, has its largest frequency in cell *i* of the orientation histogram. We use this information to help limit search space when retrieving records for matching, *see section 8.2*.

Two identical records will hash to the same address; we term this a clash. Occasionally two dissimilar records when hashed also produce a clash. When a clash occurs the pointer at this location addresses an array of pointers. In *figure 35*, *b* indicates this array and each pointer in the array addresses a stored histogram. *Figure 35*, shows the stored histogram at position *c*. The current histogram is compared to each of the histograms addressed by the pointers in the array. If the current histogram already exists, i.e. a duplicate, it is discarded. In this way we avoid duplication of stored data. If the resulting address is empty then no search and comparison is required, and we simply insert the record knowing that there is no possibility of a duplicate existing.

This hashing method provides a high number of unique addresses. This limits the number of records we have to read and compare to detect duplicates during database computation. The method does not guarantee a unique address since several different histograms may hash to the same location. The pointer at each location addresses an array; the array contains further pointers to the addresses of each stored histogram whose cells produced the same row and hash value; invariably these are very few.

The 16-bit image plane is divided into 3 portions; the first two, A & B in figure 35, contain pointers which allow examination of small groups of records. These two areas combined occupy less than 25% of the image; the remaining 784 rows of the 16 bit image plane, area C in figure 35, are available for storing pre-computed histograms.

The first 115 rows, portion A of the image, contain the hash values discussed above. These point to array addresses in rows 116-239, i.e. portion B of the image; each array contains pointers to stored orientation histograms whose hash values are identical. Over 90% of these arrays contain a single pointer; of the remainder, only about 1% contain more than two. As a result, when searching for duplicates, we only need to compare a single histogram on average.

7.6 Inserting a Histogram into the Database.

Our algorithm for placing computed model histograms into the image plane is presented below. It is in pseudo code with comments; these are labeled by /*

```

/ *** main *** /

count = 0
block_size = 16
no_of_blocks = 1024/block_size = 64 /* row holds 64 blocks
start_of_addresses = 116 /* start row of arrays of address pointers
start_of_data = 240 /* start row of stored histograms
Until no more histograms do
    length_of_histogram = header_record[1] * 2 + 6
    row = largest_frequency_of_histogram /*in range 1..115
    column = ( $\Sigma$ cell_number2) MOD 1024 /*in range 1..1023
    if image_location[row,column] = empty then
        image_location[row,column] = count
        count = count + 1
    endif
    find_address
end until
/ *** end main *** /

```

```

/ *** find_address ***/

value = image_location[row,column]
row = (value MOD no_of_blocks)+start_of_addresses /*row no.of pointer array
column = (value MOD no_of_blocks)*block_size /* start position in row
i = 1
read array at image_location[row,column]
Until array[i] = 0 do
    pointer = array[i]
    i = i + 1
    row = (pointer MOD no_of_blocks)+start_of_data /* row holding histogram
    column = (pointer MOD no_of_blocks)*block_size /* start posn in row
    read histogram in image_location[row,column]
    compare to current histogram
    if histograms same then
        discard current histogram
        return to main
    endif
end until
compare current histogram to copy
if similar then
    discard current histogram
else
    Insert_current_record
endif

/ *** end find_address ***/

/ *** Insert_current_record ***/

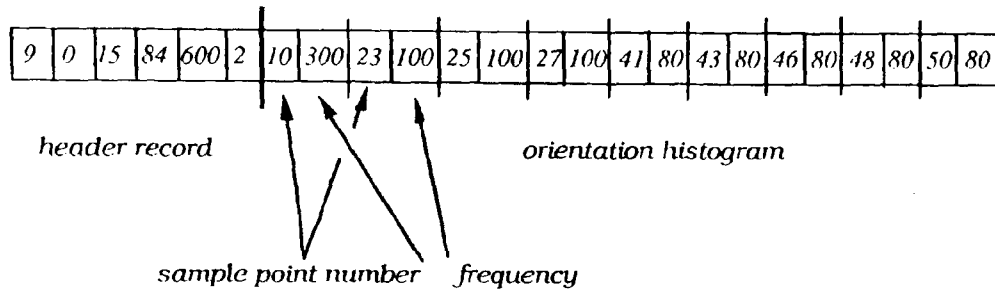
/* DIV means integer division */
copy = current histogram
array[i] = location
location = location + (length_of_histogram DIV block_size) + 1
row = (array[i] MOD no_of_blocks)+start_of_data /*row to insert histo
column = (array[i] MOD no_of_blocks)*block_size /* posn in row for start of histo
write histogram to image plane at image_location[row,column]

/ *** end insert ***/

```

7.7 An Example of Inserting a Histogram into the Database.

To illustrate our histogram insertion algorithm we will assume the following model histogram has been computed and is to be added to the database of model orientation histograms. (Figure 35 should be read in conjunction with this example.)



Variables are those used in the pseudo code algorithm of section 7.6 and are highlighted by the use of a different font. The variables are:-

count, *location*, *largest_frequency_of_histogram*, *length_of_histogram*,
row, *column*, *value*

$$\text{length_of_histogram} = 2 * 9 + 6 = 24$$

Count is a pointer to the next array of addresses in area B (figure 35), and the value of *count* is written to the position in area A found from our hashing routine. *Location* points to the next empty block of pixels in area C in which the next computed histogram will be stored. The value of *location* is placed into the *row* and *column* position of area B addressed by *count*.

We will assume that the current values of *count* = 96 and *location* = 85.

row = *largest_frequency_of_histogram* = 10 (i.e. sampling point 10 frequency = 300).

From our algorithm the initial pointer is in *row* 10, (determined by the sample point number containing the largest frequency), the position in the row is given by *equation (45)*.

$$column = (10^2 + 23^2 + 25^2 + 27^2 + 41^2 + 43^2 + 46^2 + 48^2 + 50^2) \text{ MOD } 1024 = 145$$

This is position a in *figure 35* below.

If *row* 10 *column* 145 is empty, we write the integer in *count*, currently 96, to the image plane at *row* 10, *column* 145 and increment *count*. (For this example we will assume the position is empty -- if this had not been the case the value at this position would be used to locate the array of histogram addresses).

To find the start position for the array of histogram addresses *value = count = 96*,

$$row = (value \text{ MOD } 64) + 116 = 148$$

and the position within the row is given by

$$column = (value \text{ MOD } 64) * 16 = 512$$

this is position b in *figure 35* below. This position is the start of a 1-D array of length 16, in which each non-zero value is a pointer to stored histogram locations. Each of the histograms addressed in this array will have the same hash value and their largest frequency occurs at the same sampling point, i.e. sampling point 10 in this example. When searching for duplicates we will compare these orientation histograms to the current histogram. However in this example, as the initial pointer position at *row* 10 *column* 145 is empty, then the array at *row* 148, *column* 512 is also empty. The current orientation histogram is written to the image plane at the address pointed to by *location* (currently = 85).

$$row = (location \text{ MOD } 64) + 240 = 261$$

$$column = (location \text{ MOD } 64) * 16 = 336.$$

This is position c in figure 35 below.

The value of *location* is written to the first empty position in the array, (position b in figure 35). *Location* is updated as follows

$$location = location + (24 \text{ DIV } 16 + 1) = 85 + 2 = 87.$$

This locates the position in area C where the next computed orientation histogram will be stored.

By addressing the image plane in blocks of 16 pixels, some available space is wasted because histograms rarely have a length that is a multiple of 16. However the ability to use the entire image plane for database storage far outweighs this limitation.

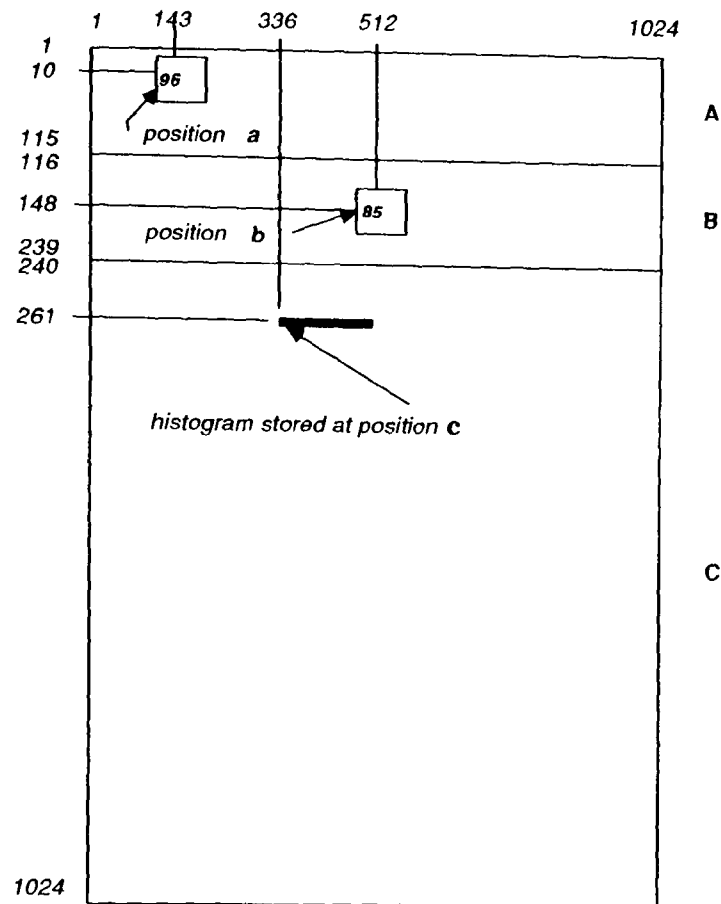


Figure 35. The image plane is divided into three areas, A,B & C. The largest area C contains the computed histograms.

7.8 Dealing with Negative Addresses.

The range of 2^{16} integers is $-2^{15} \dots +2^{15}-1$. When the variable, location, is incremented to a value greater than $+2^{15}-1$ the value becomes negative e.g.

$$32767 + 1 = -32768.$$

To find the address of a histogram on the image plane whose location is negative then we use the following algorithm (using pseudo code)

```

if (array[i] < zero) then
  pointer = array[i] + 216
  row = (pointer MOD no_of_blocks)+start_of_data
  column = (pointer MOD no_of_blocks)*block_size
endif

```

7.9 Non-Linear Scaling of Radiance Values.

Blemishes, scratches or slight variations in reflectance on the surface of an object can lead to unrecognised portions of the visible object.

Ikeuchi et al¹⁹ normalised radiance values defined by:-

$$N_i = \left[\frac{L_i}{L_1 + L_2 + L_3} \right] \quad i = 1, 2, 3. \quad (45)$$

Many different triples of quantised radiance when normalised produce the same normalised triple. This decreases the range of different triples and consequently the number of entries in the look-up table. Using *equation (45)* we only achieve about 600 entries in the look-up table.

Two widely separated sampling points with areas of radiance which, when normalised produce the same triple, create problems. One of the two sampling points will not be represented in the look-up table cell corresponding to that triple of indices. It may result in the sampling point not being represented in the table at all. This creates further problems because the look-up table will be unreliable at these orientations, resulting in incorrect estimates of surface orientation.

Normalisation tends to smooth the orientation histogram by spreading entries more widely. This reduces the value of the average foreshortening factor, h , of the sensed orientation histogram. Consequently it is often different from those of the

pre-computed orientation histograms with a similar orientation. We could not use this feature to help reduce the number of subsequent comparisons.

We scaled the radiance triples using

$$S_i = \left[\frac{L_i}{L_1 + L_2 + L_3} \right] L_i \quad i = 1, 2, 3. \quad (46)$$

This has several benefits. Firstly different quantised values when normalised using *equation (45)*, can result in duplication with a different set of radiance values, e.g

$$L_1 = 100, L_2 = 150, L_3 = 200$$

when normalised give

$$N_1 = .222 ; N_2 = .333 ; N_3 = .444.$$

Similarly the radiance values 50, 75, 100 will normalise using N_i to identical values.

Using *equation (46)* we obtain two different sets of values

$$a) \quad 22.22, 50.0, 88.89$$

$$b) \quad 11.11, 25.0, 44.44.$$

This helps reduce the problem of unrepresented sampling points in the table and increases the accuracy of the look-up table.

Secondly using this method we find the problems caused by inconsistent surfaces are largely removed; very few pixels over a viewed object are unrecognised. This is so because un-normalised triples of radiance, when distorted by noise, will result in false estimates of surface orientation from the look-up table; alternatively if the look-up table cell corresponding to this triple is empty the pixel is unrecognised. All unsmoothed images contain large amounts of noise leading to many unrecognised pixels; this will

adversely affect the sensed orientation histogram. The contributions made by each surface orientation will be distorted.

By dividing each of the three radiance values by their sum we are reducing the effects of noise because the divisor also contains the noisy values. However because we multiply the quotient by the noisy value, the error is re-emphasised; however its overall effect is far less than it would be with no such scaling. As the quantised bands have an upper and lower limit, the scaled values will often fall within the band. It also has another benefit by reducing the effects of silicon defects in the CCD camera.

Thirdly the number of entries in the look-up table increases to approximately 2500. A more complete and accurate look-up table is compiled; this produces improved estimates of surface orientation, and consequently more accurate matching and subsequent results. The improved accuracy of the sensed orientation histogram produces a more accurate estimate of the average foreshortening factor. We use this value to help reduce the number comparisons between the sensed and stored histograms.

CHAPTER 8.

Object Recognition and Orientation

The calibration sequence creates sufficient information to generate viewer-centred descriptions of viewed objects, i.e. the sensed orientation histogram. We describe how this is achieved in *section 8.1*. We also compute features of the visible object, see *section 8.2*; these are used to help keep the number of comparisons to a minimum. Our method of retrieving histograms from the database is illustrated in *section 8.3*. The database of viewer-centred model orientation histograms provides information necessary to be able to recognise the object and to determine its orientation from this description. This is achieved by matching the sensed description with model descriptions stored in the database; we describe this matching process in *section 8.4*.

8.1 Building the Sensed Description.

We capture three images of the unknown object, each viewed through the plane polarising filter. The order and rotation of the filter is the same as the sequence used to capture images of the sphere. We apply the non-linear scaling, *equation (46)*, to all images including the calibration sequence. Using the upper and lower threshold values determined for each image during the calibration stage, the images are segmented into object and background. When all three pixel values, i.e. one pixel from each image at point (x,z) , exceed the upper threshold of the corresponding image, the pixel is considered to be background. Similarly when all three pixel values are below the lower threshold value for each corresponding image it is considered to be a self-shadowed surface and the pixel value set to indicate this; a count of all occurrences of self-shadowing is incremented. Self-shadowed surfaces are described in *section 7.2*; an example of these surfaces on the toy is given in *figure 36*.

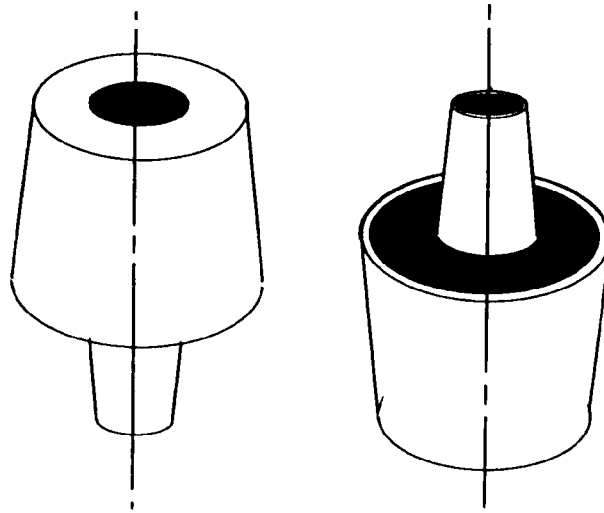


Figure 36. The self-shadowed surfaces of the toy, these are represented by the shaded areas.

Any other combination of radiance values is considered to be on the surface of the object. The radiance values are scaled, using *equation (46)*, to give three new values and each is compared to the quantised radiance bands derived during the calibration stage. This determines in which quantised band each value belongs; the band number (1..16) in each image becomes an index to the look-up table, *see figure 37*.

The look-up table position indexed by these three values contains the sampling point number that corresponds to the orientation associated with these three values. The appropriate cell of the sensed orientation histogram is incremented.

This is repeated for each point in the three images. In this way a description of the visible object is produced; the description relates visible area to surface orientation. An empty look-up table location implies an unrecognised orientation. We take the average radiance value of a 3*3 window around this unrecognised pixel and re-subject the three new values to the above procedures. This we found to be most satisfactory.

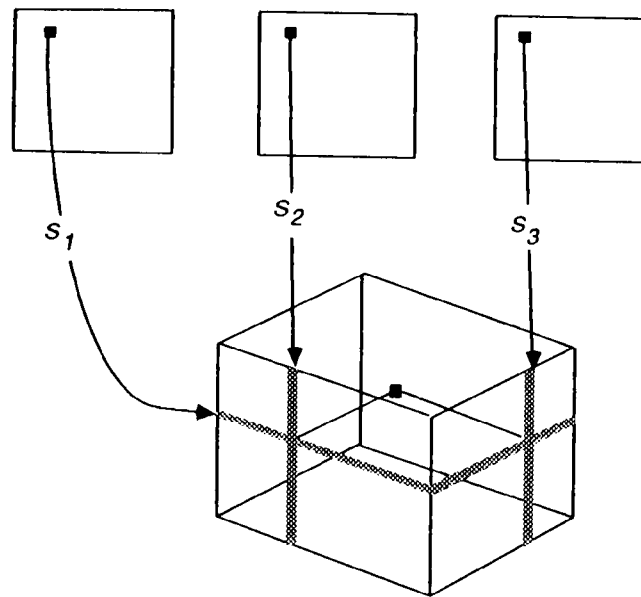


Figure 37. The look-up table is accessed using a triple of values derived from radiance measurements from three images obtained with a polarising filter at three orientations.

We use moments to help reduce the number of comparisons necessary to achieve a match; these are described in section 6.2. The values, M_{ab} , (equation (48)), are accumulated for each pixel of the object during this segmentation stage.

8.2 Features of the Visible Object.

We use features of the visible object first to select model data for possible comparison, and second to make initial comparisons between the sensed orientation histogram and the model data. These features are the larger visible surfaces (frequencies), the orientation of the principal axis, p , and the average foreshortening factor, h . We use the larger frequencies of the sensed orientation histogram to guide our search through the database. These help reduce the number of comparisons with stored descriptions needed to achieve a match. We have already pre-computed and stored some features of the model descriptions; these are held in the header record of each

stored histogram. Computing the same features for the sensed object enables an initial comparison of both sets; the pre-determined thresholds $p+t_p$, $p-t_p$ and $h+t_h$, $h-t_h$ determine whether a full comparison is required.

Records are stored in the database using the visible surface with the largest frequency. The initial search for suitable stored orientation histograms is based upon the larger cell frequencies of the sensed orientation histogram. Initially we used a fixed threshold, t_f , to determine these. This is arbitrarily set at 5% of the normalised sum of frequencies; it is therefore a constant value. A variable threshold, t_v , replaces this and is automatically generated. We use

$$t_v = 3s \quad (47)$$

where s is the standard deviation of the sensed orientation histogram. This value will vary from object to object depending upon orientation and the level of noise; *chapter 9* describes the effect of this value in more detail.

The second feature is found with the aid of moments.³⁰ Using these we can calculate both the position of the centroid and the orientation of the principal axis, p , of the 2-D projection of the object.

The moments, M_{ab} , of a digital image are defined by

$$M_{ab} = \sum_x \sum_z x^a z^b. \quad a, b = 0, 1, 2, \dots \quad (48)$$

The coordinates $(C(x), C(z))$ of the centroid are given by

$$C(x) = \frac{M_{10}}{M_{00}} \quad (49)$$

$$C(z) = \frac{M_{01}}{M_{00}}, \quad (50)$$

where M_{00} is the area.

We redefine the origin to be the centroid of the viewed object. The resulting translation invariant moments are given by:-

$$M_{ab}' = \sum_x \sum_z [x - C(x)]^a [z - C(z)]^b. \quad a, b = 0, 1, 2, \dots \quad (51)$$

These in turn can be expressed in terms of the original M_{ab} , where

$$M_{20}' = M_{20} - C(x) * M_{10} \quad (52)$$

$$M_{02}' = M_{02} - C(z) * M_{01} \quad (53)$$

$$M_{11}' = M_{11} - C(x) * M_{10} . \quad (54)$$

The orientation of the principal axis, p , is

$$p = \frac{1}{2} \tan^{-1} \left[\frac{2 M_{11}'}{M_{20}' - M_{02}'} \right]. \quad (55)$$

We set the threshold value, t_p , to be ± 20 degrees of the principal axis orientation.

To provide the final feature, the sensed orientation histogram is examined and the average foreshortening factor, h , is computed from the cell frequencies using equations (41) and (42). The threshold, t_h , is set at +15%, -10% of h .

A single pass of the three images results in the segmentation of object and background, and produces the sensed orientation histogram. The values, M_{ab} , are accumulated simultaneously with these operations and the moments (equations (49)...(55)) computed prior to retrieving records from the database for subsequent matching.

The sensed orientation histogram is normalised by dividing the frequency in each cell by the sum of these frequencies. Both stored and sensed descriptions are now compatible. The count of self-shadowed surfaces accumulated during image segmentation is considered to be a visible surface and is included in the histogram normalisation process.

Having computed the sensed orientation histogram and the features of the objects 2-D projection, we can retrieve records from the database for possible comparison.

8.3 Retrieving a Histogram from the Database.

We present our histogram retrieval method as a pseudo code algorithm. Comments are included and are pre-fixed by -- /* .. -- in a similar fashion to those of Chapter 7.

```

/** retrieve_a_record */
block_size = 16
no_of_blocks = 1024/block_size = 64
start_of_addresses = 116 /* start row of arrays of address pointers
start_of_data = 240 /* start row of stored histograms
min_value = standard_deviation*3 /* freq.threshold  $t_v$ 
for i := 1 to 115 do
    if sensed_histogram_cell[i] freq greater than min_value then
        find_records
    endif
end for
/** end retrieve_a_record */

/** find_records */
for position = 0,1023 do
    if image_location[i,position] greater than zero then
        value = image_location[i,position]
        row = (value MOD no_of_blocks)+start_of_addresses /*pointer array row no.
        column = (value MOD no_of_blocks)*block_size /* position in row
        read array at image_location[row,column]
    end if
end for

```

```

Until array[i] = 0 do
  pointer = array[i]
  row = (pointer MOD no_of_blocks)+start_of_data /*row containing histogram
  column = (pointer MOD no_of_blocks)*block_size /*position in row
  read histogram in image_location[row,column]
  compare features
  if (p-tp ≤ p ≤ p+tp) and (t-th ≤ h ≤ t+th) then
    compare to sensed orientation histogram
  endif
end until
endif
end for
/** end find_records */

/** end of algorithm */

```

The process of locating addresses of histograms whose location corresponds to a value in excess of $+2^{15}-1$, i.e. a negative value, is described in *section 7.8*.

The database of model object descriptions are stored using pointers to addresses. These pointers are stored in row numbers which correspond to the orientation of the largest visible face (frequency) of the object. The sensed orientation histogram is examined and all cell numbers whose frequency is greater than our threshold t_v , (*min_value* in the above algorithm), are stored. Using these as row numbers, pointers to database records in the corresponding rows are accessed; as a result we examine only histograms whose largest frequency corresponds to one of the larger visible surfaces (frequencies) of the sensed histogram.

Each accessed database histogram is examined and an initial comparison takes place between the sensed principal axis and the value stored in field four of the header record. This determines whether a further comparison is required. Only if the stored value is within the allowed tolerances, $p \pm t_p$, does the initial comparison continue. If

the stored value exceeds these tolerances no further comparison is made with this stored histogram.

Model histograms which pass this initial test have their average foreshortening factors compared to the sensed value. The pre-computed value in position five of the header record must be within $h \pm t_h$ for a full comparison to take place.

8.4 Matching Image and Model Orientation Histograms.

Objects are identified by matching the sensed orientation histogram with a model orientation histogram from the database, using a least squares technique. The sensed orientation histogram is compared with each selected model histogram in turn; the sum of the squares of the differences in the two frequencies for each cell is formed. Let f_{mi} represent the frequency in cell i of the model histogram and f_{si} be the frequency in the corresponding cell of the sensed orientation histogram. We minimise the residual sum of squared differences S defined as follows:-

$$S = \sum_{i=1}^{115} [f_{mi} - f_{si}]^2. \quad (56)$$

We set an initial threshold, t_m , to avoid completing unnecessarily bad matches. The initial value of t_m is arbitrarily determined and is used to increase computational efficiency. With each new set of images, the value of t_m is re-initialised to this commencing value. If during a comparison a residual sum exceeds this threshold the match is abandoned. Once a complete sum of squared differences has been achieved, (we term these full comparisons), the resulting lower residual sum becomes the new threshold value, t_m . Using this heuristic reduces the number of full comparisons to a minimum. Partial comparisons are those which have been terminated by exceeding the threshold value, t_m .

The stored viewer-centred histogram which produces the smallest sum of squared differences is regarded as the best match. In this way we recognise an object and determine its orientation at the same time.

Several matching solutions are possible. We experimented with a correlation method. This produces results similar to those of the least squares method. However the correlation between each selected model histogram and the sensed description must be fully computed. As a result it is not possible to terminate bad matches.

Whilst other forms of matching are feasible, least squares gives satisfactory results and is fast. The method does imply that provided at least one comparison leads to a value of $S < t_m$ we will achieve a result. The selected model histogram will determine both the object's identity and orientation. To illustrate an example of incorrect matching, a cylinder when matched with a database of toy histograms will provide the orientation and identity of the selected toy histogram. This assumes that the resulting value of S is lower than the initial value of t_m . However attempting to fool the system in such a way achieves nothing and is meant only as an implied criticism of our matching method.

CHAPTER 9.

Results

We demonstrated our method using the test objects described in *section 5.1*, a toy and a cylinder. Two sets of test objects are used, these are a blue set and a black set. Both sets include a calibration sphere sprayed with the same acrylic paint used to cover the corresponding test objects.

We present results for both test objects. These have been collated into four groups: the first group detects the orientation of each individual object; these are given in *section 9.1*. The second group, *section 9.2*, illustrates both recognition and orientation of the viewed objects. In *section 9.3* Gaussian noise is added to images in varying degrees to determine the method's ability to withstand noise.

Initially we used a fixed threshold, t_f , to help select stored orientation histograms for possible comparison. The value of t_f is fixed at 5% of the normalised sum of sensed frequencies, i.e. we only search those rows of the database which correspond to cell numbers whose frequencies exceed this value.

We re-tested the method using a variable threshold, t_v , *section 9.4* presents these results. Finally in *section 9.5* a statement on repeatability of results is made.

The results obtained for orientation only, involve using a database for a single object i.e the toy database for toy results and the cylinder database for cylinder results. Recognition and orientation results used a combined toy and cylinder database.

The uncertainty in degrees of arc is computed from the sensed orientation (α_s, β_s) and the true orientations (α_t, β_t) of the viewed object using *equation (34)*. The distinction between full and partial comparisons is made in *section 8.4*.

9.1 Results: Orientation Only.

We now present results for the orientation of the toy. The results are typical in that some were better whilst others were worse.

Orientation		Uncertainty in degrees of arc	Number of Comparisons		Correct end?
true (α_t, β_t)	sensed (α_s, β_s)		Full	Partial	
61,0	64,7	4.3	6	54	yes
13,60	13,51	8.8	8	10	yes
0,61	-9,63	9.2	9	44	yes
-13,300	-15,296	4.0	9	46	yes
49,318	45,308	8.0	4	40	yes
-26,58	-22,62	5.6	3	22	yes
61,354	62,341	6.5	1	52	yes
-29,0	-26,357	3.9	3	3	yes
20,339	12,336	8.3	1	16	yes
14,25	6,25	7.9	2	34	yes
-49,42	-46,27	10.6	4	17	yes
-85,289	-90,0	5.2	1	3	yes
-20,270	-22,270	2.0	4	84	yes
43,14	45,28	9.8	2	7	yes
38,26	36,40	10.8	1	2	yes

Table 3. The estimated orientation of the toy, using a fixed threshold, t_f , and the toy database.

This gives the following:-

average uncertainty in degrees of arc = 7.0

standard deviation, in degrees of arc, of uncertainty in sample = 2.6

average number of full comparisons = 3.9

average number of partial comparisons = 28.9

The sampling points used to represent object orientation cover the visible hemisphere only. These are adequate to fully describe the cylinder; however the

principal axis of the toy passes through two quite different surfaces. In order to distinguish between them, we simply describe orientation in the visible hemisphere and differentiate between each end of the toy. Our results simply state whether the correct end is determined by the system for each set of test images of the toy.

We now present results for the orientation of the cylinder. Once again the results are typical.

Orientation		Uncertainty in degrees of arc	Number of Comparisons	
true (α_t, β_t)	sensed (α_s, β_s)		Full	Partial
0,61	-9,63	9.2	3	3
-13,60	-9,63	4.8	2	4
15,90	22,90	7.0	4	54
-26,58	-34,58	8.0	4	0
-26,302	-28,299	3.6	3	65
30,325	26,330	6.1	3	9
21,319	25,314	6.4	5	9
28,337	35,341	6.7	6	17
-30,35	-37,40	8.0	5	21
-15,270	-9,270	6.0	3	6
10,44	9,38	6.1	4	36
25,345	21,345	3.7	3	4
29,4	24,10	10.1	3	17
7,28	6,25	3.2	1	12
-13,60	-9,63	4.8	12	60

Table 4. The estimated orientation of the cylinder using a fixed threshold, τ , and the cylinder database.

This gives the following:-

uncertainty in degrees of arc = 6.2

standard deviation, in degrees of arc, of uncertainty in sample = 2.0

average number of full comparisons = 4.1

average number of partial comparisons = 21.1

9.2 Results: Recognition and Orientation.

We combined the toy and cylinder databases to test the recognition capabilities of the method. A typical selection of the results is presented in *table 5*.

Object	Orientation		Uncertainty in degrees of arc	Number of Comparisons		Correct end?	recog- nised?
	true (α_t, β_t)	sensed (α_s, β_s)		Full	Partial		
toy	13,60	13,51	8.8	8	14	yes	yes
toy	-49,42	-46,27	10.6	4	17	yes	yes
toy	49,318	45,308	7.9	9	47	yes	yes
toy	-29,0	-26,357	3.9	3	3	yes	yes
toy	20,339	12,336	8.3	1	16	yes	yes
toy	14,25	6,25	7.9	5	21	yes	yes
cyl	30,325	26,330	6.1	3	9	na	yes
cyl	38,337	35,341	6.7	6	17	na	yes
cyl	-30,35	-37,40	8.0	5	21	na	yes
cyl	0,90	77,90	13.0	9	61	na	yes
cyl	25,345	21,345	3.7	3	20	na	yes
cyl	11,44	9,38	6.1	4	37	na	yes

Table 5. The recognition and estimated orientation of both test objects, using a fixed threshold, t_f , and the combined toy and cylinder database. (na = not applicable)

Determining which end of the object we are viewing is only applicable to the toy.

The cylinder results, therefore, contain the abbreviation, na, i.e not applicable.

This gives the following:-

average uncertainty in degrees of arc = 7.6

standard deviation, in degrees of arc, of uncertainty in sample = 2.6

average number of full comparisons = 5.0

average number of partial comparisons = 23.6

9.3 The Effects of Noise.

To discover the method's response to noise, several sets of images of objects were corrupted with added Gaussian noise in varying degrees. We generated random numbers with a Gaussian distribution of mean μ_g and variance σ_g^2 . Using the resulting values we modified pixel values in the corresponding images to simulate the effects noise.

The signal to noise ratio (snr) at level, λ , will be defined by

$$\lambda = \frac{\bar{x}}{\sigma_g} \quad (57)$$

where \bar{x} is the average integer pixel value over the object.

An approximation to a Gaussian distribution with mean, $\mu_g = 0$ and standard deviation, σ_g , can be obtained by producing a random variable, *randnumber*, where

$$randnumber = rand_g * \sigma_g. \quad (58)$$

The value σ_g is an input parameter, and $rand_g$ is a normally distributed, $N(0,1)$, random number. The variable $rand_g$ is determined from a specific form of the Central Limit Theorem of statistics which may be written as:-

$$rand_g = \sum_{r=1}^{12} [f_r - 6] \quad (59)$$

and f_r is a uniformly distributed random number on the range $[-1,1]$.

Each corrupted pixel is modified as follows:-

$$newpix(i) = oldpix(i) + randnumber. \quad (60)$$

Gaussian noise is added to each of the images using this method.

We corrupted 50% of the pixels in each set of three images, the choice of which pixel to corrupt is also made randomly.

Because the sensed orientation histogram is normalised, and the number of histogram cells is constant, the mean cell frequency value \bar{f} is also a constant for all sensed histograms.

Noise in images tends to smooth the frequency distribution of the sensed orientation histogram. This reduces the frequency variance, and hence the standard deviation of the sensed histogram, see figure 38. This is a feature of the fixed number of cells of the orientation histogram, and the constant total frequency, resulting from normalisation.

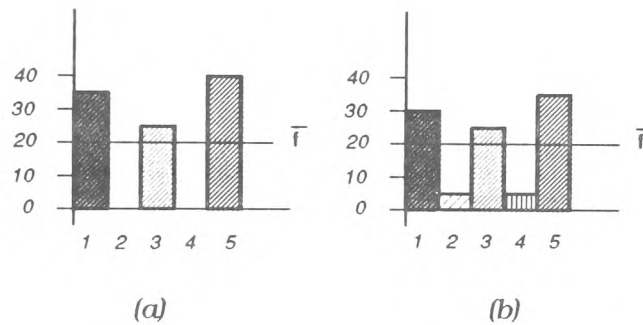


Figure 38. The effects of noise on the cell frequencies of the sensed orientation histogram, $\bar{f} = 20$ for both histograms; the standard deviation for (a) is 17.03 and for the 'noisy' histogram (b) 12.18.

We use the standard deviation to automatically threshold areas of the database for possible comparison, equation (47).

The smoothing effect of noise on the sensed orientation histogram frequencies is further indicated by the value of the average foreshortening factor, h . This value also reduces as noise levels increase and as a result some previously examined histograms are excluded from comparison. In this way the choice of stored histograms for comparison is limited and the probability of finding a good match is reduced. A

variable threshold, t_v , largely overcomes this problem by making available sufficient stored histograms to ensure the best match possible from the available data. It achieves this by examining more histograms made available by the reduced value of t_v .

Alternatively matching could be achieved by computing the correlation between sensed and stored histograms. This method would require the correlation coefficient, ρ to be computed for all selected stored histograms. Our least squares method is computationally more efficient and provides adequate results. We prefer it to a correlation method for this reason.

The correlation coefficient, ρ , reduces with increased amounts of added noise, indicating a poorer match. We include this value in the table as an additional measure of similarity between the two matched histograms. We have computed the value, ρ , for the remaining results to indicate the confidence of our least squares matching method. We could however set a value, t_ρ as a lower limit in the matching process. A match whose correlation coefficient, ρ , is below this value, is not acceptable.

We added Gaussian noise to 50% of pixels in each of the three images of the toy. The toy was recognised and its correct end determined in all but the final result. The results presented in *table 6* are from the same set of three images but with different noise levels. We use a variable threshold, t_v , and the combined toy and cylinder database.

σ_g	corr. ρ	Orientation		uncertainty in degrees of arc	Number of Comparisons		average	principal	std. dev'n
		true	sensed		foreshort. f	axis	of histo.		
		(α_t, β_t)	(α_s, β_s)					Full	Partial
0	0.77	38,308	45,308	6.7	4	52	0.553	-40	19.01
5	0.73	38,308	45,308	6.7	4	54	0.548	-40	16.00
8	0.71	38,308	45,308	6.7	4	52	0.542	-40	14.80
12	0.70	38,308	45,308	6.7	3	54	0.529	-40	12.20
15	0.67	38,308	45,308	6.7	3	60	0.514	-40	11.18
20	0.65	38,308	45,308	6.7	5	56	0.497	-40	10.02
25	0.24	38,308	43,71	83.7	2	1	0.419	19	8.99

-Table 6. Results of Gaussian noise added to 50% of pixels in each of the three images of the toy.

Legend:

σ_g : Standard deviation of added Gaussian noise.

ρ : correlation coefficient,³¹ where

$$\rho = \frac{1}{n} \sum \left[\frac{x - \bar{x}}{\sigma_x} \right] \left[\frac{y - \bar{y}}{\sigma_y} \right].$$

h : average foreshortening factor, equation (41).

p : orientation of the principal axis of objects 2-D projection, equation (55).

s : standard deviation of sensed orientation histogram, where

$$s = \sqrt{\frac{\sum (f_i - \bar{f})^2}{n}}.$$

and

$n = 115$ cells of sensed histogram.

\bar{f} = mean frequency value of histogram.

f_i = frequency of cell i .

The variable threshold, t_v , reduces as the standard deviation, s , of the sensed orientation histogram reduces. As previously stated this allows more stored histograms to be made available for possible comparison so increasing the possibility of a good match.

The incorrect final result is largely due to increased noise affecting the segmentation of object from background. The result is an incorrect estimate of the principal axis of the 2-D projection, leading to an incorrect search area in the database, and a subsequent incorrect estimate of orientation. As the value h decreases, increasing numbers of histograms are rejected because their pre-computed value of h , stored in the header record is outside the values $h \pm t_h$.

The results above were achieved using the photometric box with each of the four quadrants of the box painted a different matt shade, including white, see *chapter 4*. We re-painted the photometric box in two matt shades, white and blue and the results are presented in the next section.

9.4 Results: Recognition and Orientation using a Variable Threshold.

The photometric box to the left of a vertical line passing through the viewing hole was painted white, the remainder of the box was painted blue. We re-calibrated the system and using the least squares matching method combined with a variable threshold, t_v , we achieved the following results.

Object	corr. ρ	Orientation		Uncertainty in degrees of arc	Number of Comparisons		average foreshort. f h	Recog- nised?	Threshold t_v
		true (α_t, β_t)	sensed (α_s, β_s)		Full	Partial			
Toy	0.78	38,308	45,308	6.7	4	36	0.553	yes	56
Toy	0.85	-57,35	-62,19	5.2	5	41	0.545	yes	49
Toy	0.6	61,5	64,7	3.0	4	42	0.559	yes	43
Toy	0.95	45,0	38,12	11.3	2	7	0.643	yes	57
Toy	0.75	-45,0	-39,4	6.7	4	68	0.566	yes	47
Toy	0.75	0,45	-1,51	6.1	4	19	0.584	yes	51
Toy	0.93	45,0	39,356	9.5	7	9	0.467	yes	24
Toy	0.92	-11,44	-15,36	6.7	2	3	0.502	yes	33
Cyl	0.69	-13,60	-13,50	10.1	3	7	0.571	yes	51
Cyl	0.73	-13,300	-15,296	4.1	5	38	0.551	yes	56
Cyl	0.77	7,28	6,25	3.2	2	2	0.466	yes	69
Cyl	0.78	21,319	25,314	6.4	1	2	0.494	yes	72
Cyl	0.77	38,333	35,341	6.7	3	22	0.523	yes	56
Cyl	0.74	-30,35	-37,40	8.1	4	15	0.514	yes	66
Cyl	0.76	-38,308	-40,304	3.5	4	29	0.546	yes	56
Cyl	0.52	36,307	34,302	4.2	3	23	0.551	yes	53

Table 7. Results for both toy and cylinder, the toy was recognised and its correct end determined in all the above images. We used a variable threshold, t_v , and the combined toy and cylinder database.

This gives the following:-

average uncertainty in degrees of arc = 6.4

standard deviation, in degrees of arc, of uncertainty in sample = 2.4

average number of full comparisons = 3.2

average number of partial comparisons = 20.2

We carried out tests on 80 sets of images. Over all tests our average results were as follows.

average uncertainty in degrees of arc = 6.8

standard deviation, in degrees of arc, of uncertainty in sample = 2.51

average number of full comparisons = 4.2

average number of partial comparisons = 24.7

The look-up table for the blue objects contained approximately 2600 entries and for the black toys approximately 2500 entries. The reason may be explained as follows. Firstly, the two colours have a different absorptivity. The darker the object the greater the amount of incident light absorbed by the object, this is due to the 'umov' effect.²¹ Secondly the two colours possess slightly different refractive indices;²¹ consequently the polarised reflectance maps of both blue and black spheres are different. The result of such variation is a difference in the number of entries in the resulting look-up table.

As an additional test, images of the blue objects are presented to the system using the look-up table produced from a black sphere. Similarly images of the black objects are processed using the look-up table constructed from a blue sphere; as expected the resulting estimates of surface orientation are inaccurate. As a result the system fails when presented with objects whose surface material differs from the calibration sphere.

9.5 Repeatability

We repeated these tests at a later date, some of the tests were repeated on four or more occasions. Clearly slight differences in the actual results will be evident because identical images, due largely to noise in the imaging device, are not possible. Individual cell frequencies will vary from image to image. This leads to differences in computed values such as the residual sum of squared differences, S , and the correlation coefficient,

ρ . The average foreshortening factor, h , will also differ as will the number of comparisons.

Despite these changes we achieved the same results in terms of recognition, orientation and uncertainty.

CHAPTER 10.

Discussion

Dark specular materials are highly suited to the use of polarised reflectance contours. This is because dark materials absorb more of the incident illumination than light materials, whilst the incident light that is reflected is highly plane polarised. This is known as the 'umov' effect.²¹

The method detailed is not suitable for all specular surfaces. It is difficult to be precise as to the extent of the possible application because both the absorptivity and the refractive index of the reflecting material contribute to the contours of the resulting reflectance maps, *see figures 39 and 40.*

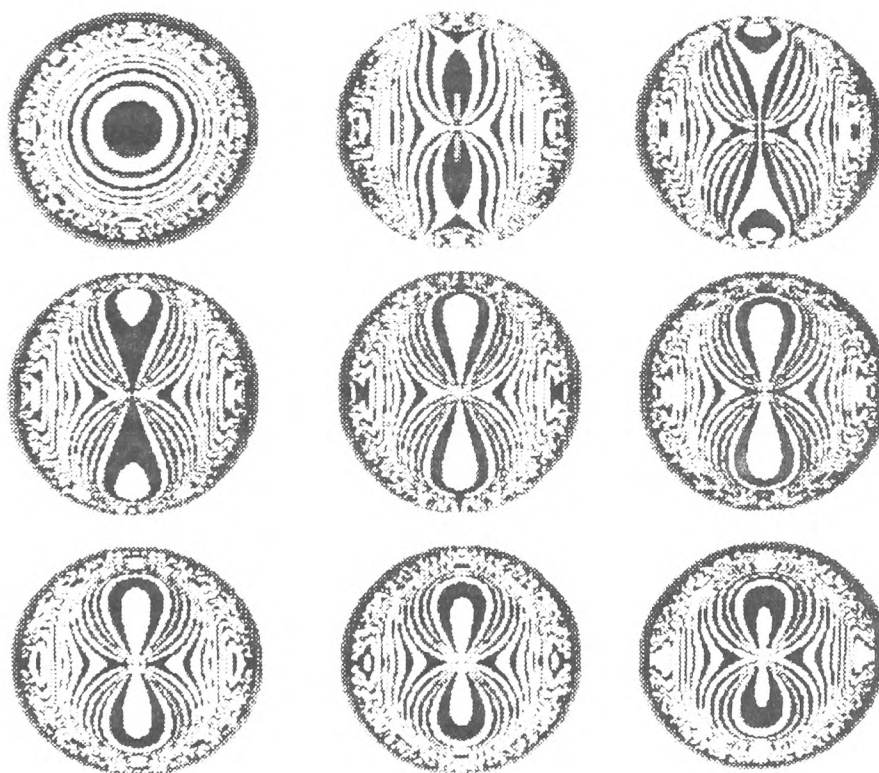


Figure 39. Each of the nine images is computed with a constant absorption factor 0.75.(in our model we assume that absorption is in the range [0..1]), and a different refractive index. The refractive index ranges from left to right, top to bottom, from 1.0 to 3.0 in increments of 0.25.

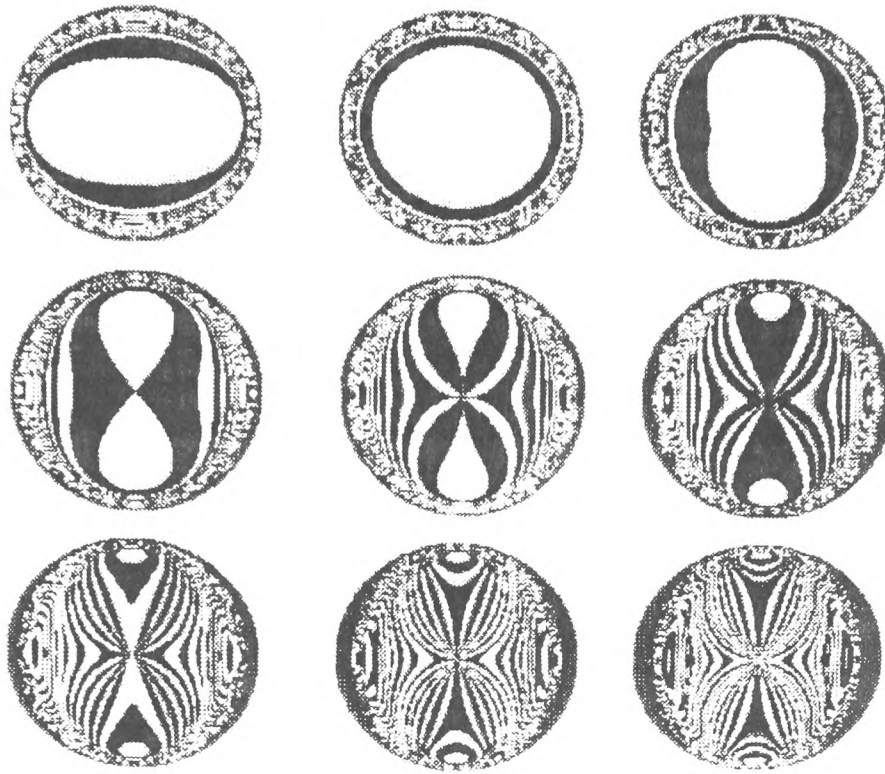


Figure 40. Each of the nine images is computed with a constant refractive index of 1.5, and a different absorption factor. The absorption factor ranges from left to right, top to bottom from 0.1 to 0.9 in increments of 0.1.

Other researchers^{9,10,11} have difficulty in illuminating highly specular surfaces uniformly. Coleman et al⁹ find that specular highlights on the viewed objects cause measured radiance values to be higher by the magnitude of the specular component. These produce incorrect estimates of orientation. A fourth light source is required to detect and hence remove specular highlights. Specularity is detected by a high mean deviation in the four estimated normals, where one normal is obtained from each triple of radiance. A low mean deviation occurs when specularity is not present. The specular highlight is removed by using the three normals with the smallest reflectance factor.

Ikeuchi¹⁰ uses three linear lamps mounted below the level of the objects, thus avoiding direct illumination. The incident light is reflected onto the objects from an overhead Lambertian surface. Some parts of the calibration sphere are only illuminated

by two light sources and as a result surface orientations in these regions cannot be uniquely determined. He also finds that the choice of light source distribution affects the accuracy and extent of these regions.

Ray et al¹¹ experience drastic changes in radiance values due to the specular component. These occasionally exceed the dynamic range of the camera. Highly polished objects produce a virtual image of the light source. They found that these reflections can be avoided by using extra images or by changing the location of the light sources.

We experimented with fluorescent tubes directly illuminating our specular spheres. The light sources themselves were reflected as virtual images whilst the remaining area on the calibration sphere showed little if any variation in reflected radiance. Even when the fluorescent tubes were covered by diffusers the resulting images were only slightly improved. We conclude that such a method would be better suited to objects possessing a high diffuse component.

Using a plane polarising filter to view specular objects illuminated by an extended light source is an effective solution to many of the problems caused by reflections from highly specular surfaces. The result is a polarised reflectance map which we can use to build descriptions of viewed objects.

The symmetry of the polarised reflectance map of the sphere ensures that a unique triple of radiance is not possible without variations in the extended light source radiance, *see figures 12 & 13*.

Painting the photometric box different matt shades is a simple but effective solution. This lifts the potential ambiguities by enabling unique triples of radiance values to be associated with surface orientation.

The reflected radiance of a Lambertian surface patch depends only upon the angle of incidence. A specular surface illuminated by an extended light source and viewed through a plane polarising filter has several dependent factors; the angle of incidence, the refractive index and the absorptivity of the reflecting surface material, *see equations (28) & (18)...(21)*. All of these components combine in varying degrees to produce the variation in the resulting reflectance map. It is the rotation of the polarising filter which determines the axes of symmetry of the polarised reflectance map. The angle of rotation of the polarising filter should not be multiples of 90 degrees, this will simply create duplicate triples of radiance. Although no optimum angles of rotation have been derived, we found that rotations approximately 60 degrees apart worked well.

A large proportion of the visible surface of the sphere has a small variation in polarised reflected radiance. Although highly polarised the reflected radiance is fairly low. Typically 40%-70% of the visible area may have radiance values in the grey-level range 60-90. This area of low radiance changes as the refractive index and absorptivity of the material changes. To achieve as uniform a distribution of pixel occurrences in each of the quantised cell bands as possible, we use a linear interpolation to determine each of the sixteen cell bandwidths in each of the three images of the sphere. When compared to fixed width cell sizes, this method increases the number of entries in the look-up table by a factor of two. An undesirable side effect is to produce cell widths that are fairly narrow in some regions.

Slight variations in reflected radiance can push a triple from one cell to another, *see figure 29*. The result is a large variation in the numbers of different triples of radiance that occur over many sample points by comparison to a Lambertian surface, *see figure 41*.

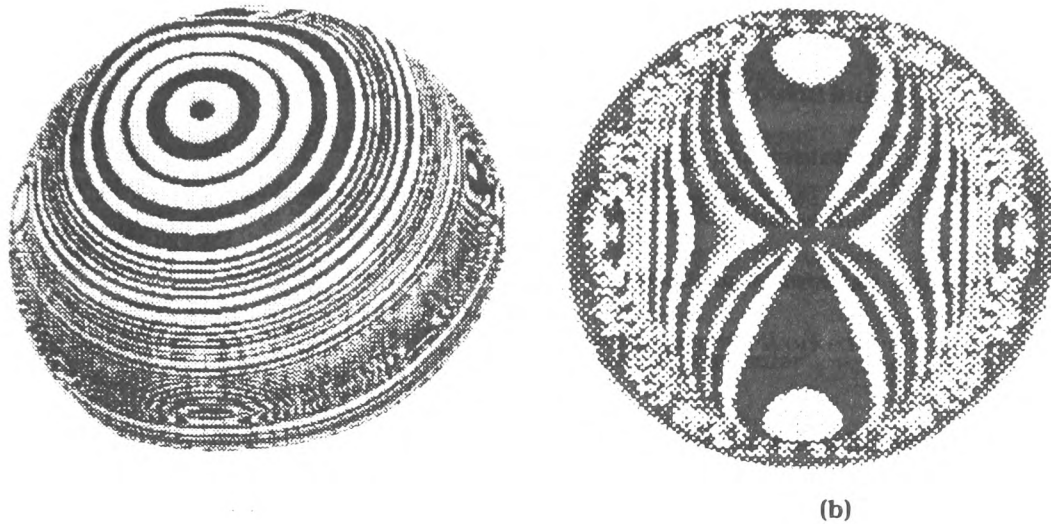


Figure 41. Reflectance contours of a) A Lambertian sphere illuminated by a collimated source, and b) specular sphere illuminated by an extended source and viewed through a polarising filter.

Allocation of triples to a particular sampling point becomes less certain as a consequence, and subsequent sensing is not as accurate as that for Lambertian surfaces. Nevertheless the look-up table generated by this method produces sensed orientation histograms which generate results comparable to those of other workers.^{13,20} The apparent discrepancy between results of Fairney et al³² and these is a direct result of this increase in variation of quantised triples over many sample points. (They use identical objects with Lambertian surfaces so enabling a fair comparison between the two sets of results). In producing narrow cell widths, slight variations in reflected radiance can result in a different quantised triple being produced. The look-up table cell corresponding to these indices may produce an incorrect estimate of surface orientation at this point on the surface of the object. These sensing errors result in a less accurate sensed orientation histogram and subsequently a reduced accuracy in matching.

Computing minimum and maximum threshold levels for each of the images from the corresponding sphere prior to run time provides a simple, effective and robust method of segmenting the object from background. It is simple because a straightforward comparison of only one radiance value from each of the three images with the pre-determined thresholds is sufficient to determine whether the pixel is background, object or a self-shadowed surface.

Only a single visible pixel lies outside the point on the sphere at which our upper threshold values are captured. As a result very little of our visible objects will lie in this region, and consequently very little data is lost. The method is effective because only background, or object orientations approaching 90 degrees to the y-axis can possess all three radiance values greater than these values. A similar explanation applies to the minimum threshold levels. These are determined by scanning the sphere to determine the lowest radiance values in each of the three calibration images of the sphere.

A surface whose incident ray is reflected from another surface of the object will have low radiance values. This occurs because much of the incident light is absorbed prior to reflection. However each subsequent reflection reduces the amount of polarisation and its radiance value. Typically this occurs on concave surfaces, *see figures 26, 27 & 31*, or on surfaces occluded from a direct light source.

By contrast, objects can sometimes be recognised by identifying features such as area and perimeter. A whole group of binary feature extraction systems have been developed, such as moments and fourier descriptors both of which are invariant to translation and rotation. Usually most object features vary with orientation and camera location, and as a result object orientation is often restricted.

Templates are commonly used, and are artificially generated after assuming the object's orientation. The generation of all possible templates and their comparison with

real data is however impractical. As a result these algorithms are generally limited to 2-D translation problems where the object is carefully presented to the imaging system.¹³

Systems using 3-D data obtained from sets of images include Stereo Vision, Photometric Stereo and Shape from Contour. Stereo Vision has received much attention in recent years.^{33,34,35} This generates disparities in a pair of images which are converted to a depth map. By contrast our method produces a map of normal vectors, the orientation histogram.

Recently, laser range finders have been used to create 3-D descriptions of scenes. Algorithms using these are simpler, cheaper and more accurate than image processing algorithms.¹³ Such systems can generate depth maps which are then converted by software to maps of normal vectors.

The viewer-centred approach that we have developed greatly simplifies many of the processes necessary to produce a satisfactory result. Other workers^{10,12,13,18,20} use object-centred descriptions of objects to derive their results. In such systems the objects are described completely relative to a particular feature of the object; typically their centre of mass. These descriptions may be generated by a solid modeller, part of a CAD/CAM system. A geometric modeller greatly simplifies the process of model rotation and subsequent descriptions. However not all potential users of computer vision have access to such systems, which tend to be expensive. Pre-computation of viewer-centred orientation histograms is feasible, and their storage on available image planes practical since most computer vision systems possess more than one.

Object-centred descriptions achieve a match by rotating the prototypical description until it aligns with the sensed object; this is computationally expensive.¹³ It also requires a notional spherical tessellation both for the sensed and prototypical

descriptions of the object. By using tessellations based upon regular polyhedra it is possible to rotate one sphere until it aligns with the other sphere; this aids the matching process.

Tessellations based upon the icosahedron are common; this allows rotation onto itself in sixty different orientations, one for each vertex of the twenty triangles that make up the shape. To rotate the prototypical description, Euler angles or rotation matrices are used. For each rotation, the normals and the areas each represents are projected onto the tessellated sphere. An orientation histogram is produced from this projection and compared to the sensed orientation histogram. Each tested position requires a rotation, a projection and a comparison.

Recently tessellations based upon combinations of 4-D polytopes and their duals producing up to 5880 points upon the sphere have been used to ensure an even sampling of the sphere. Some sixty pre-computed descriptions of the object are stored; each is a description of the object at a pre-determined orientation (one for each vertex of the icosahedron). It is these descriptions that are rotated to determine its orientation. Even with such a fine sampling of the space of possible orientations Brou¹³ states that it is possible to have errors of 17 degrees in estimated orientation. Only when in excess of 1900 comparisons are made does average error in the estimation of objects drop below 10 degrees. However, Horn²⁰ uses moments to reduce the search space; he requires approximately 100 rotations and re-projections to determine orientation to within 5-10 degrees of arc on average.

These systems separate recognition from orientation; when more than one type of object is present in the system a recognition process must first determine its type prior to the activation of the process described above.

Complications arise when the faces of an object lie near the junction of two tessellations, they rarely sit conveniently at the centre. Noise, errors in building the look-up table may result in an object's face being recorded on an adjacent tessellation. Matching prototypical data to this type of sensed data creates problems,²⁰ because such an occurrence can dramatically alter the EGI. This is particularly true if the surface presents a large contribution to the EGI, i.e. the planar end surface of the cylinder.

Our work has concentrated on the recognition and orientation of highly specular dielectric surfaces. To achieve this we have used the effects of polarisation of light on such surfaces to build a description of the viewed object. This description is compared with a database of pre-computed viewer-centred descriptions of objects to achieve both recognition and orientation. Our use of viewer-centred descriptions has several major advantages. Firstly because our database consists of viewer-centred orientation histograms, no rotation of sensed and stored data is necessary. All necessary rotations and reprojections have effectively been performed once only in the process of computing the database of stored histograms. As a result the need to rotate tessellated spheres into alignment does not occur, therefore a fixed grid of sample points is sufficient to describe both our sensed descriptions and our database of pre-computed descriptions. The main disadvantage of our method is the storage space required to hold our database. We stored both the cylinder and the toy databases using 170kbytes or approximately a fifth of the available space on the image plane. This is however offset by the ease of computation.

Added Gaussian noise led to an apparent improvement in the number of comparisons required to achieve a best match. Noise has a smoothing effect on the sensed description by pushing triples of radiance into adjacent cells of the sensed orientation histogram. Because we used a fixed threshold, t_f , to select areas of the database of histograms for possible comparison, fewer frequencies exceeded this value

so reducing the number of available histograms for comparison. Also the sensed average foreshortening factor, h , is depressed; we use this value to compare with the pre-computed values of h , stored in the header record of each pre-computed orientation histogram, prior to any comparison. The effect is to reduce still further the number of stored histograms available for comparison. By using a variable threshold, t_v , which will reduce with the effects of noise, the number of stored histograms available for comparison is increased. Similarly the value of t_v will increase with more accurate sensing, in both cases the result is a more even distribution of stored data available for possible comparison. This increases the probability of the best match being chosen.

Object symmetry is used to reduce the number of possible orientation histograms stored in the database. As an example, the database for a cube is computed as follows. By orienting a single surface of the cube into alignment with the normal at sampling point (0,0), we compute the orientations of the other visible surfaces and their frequencies to produce each histogram. This process is repeated at each sampling point on the sphere up to an angular distance of 45 degrees from the y-axis (0,0). To compute histograms beyond this point will simply duplicate existing stored histograms. Similarly to use more than a single surface of the cube will also duplicate existing data. Most objects portray some symmetry, and with this in mind we can cut the amount of computation necessary to build the database.

Two adjacent computed histograms may be so similar that storing both is ineffective, i.e. in the case of the toy, the current histogram may have 22 visible surfaces. Of these, 21 out of the 22 visible surfaces may be included in the previously computed histogram. We could not distinguish between these positions because 1) our polarimetric system is subject to noise, sensing errors and errors in building the look-up table and 2) both radiance values and the hemisphere of orientations are quantised. Storing both would not necessarily produce a better result. Rejecting histograms on the

basis of similarity with existing stored histograms does mean we throw away a great deal of data; perhaps as much as 60%. However the remaining subset is sufficiently diverse to produce results which are typically within seven degrees of arc.

The stored orientation histograms are addressed by pointers embedded in the image plane containing the database. The position of the initial pointer to a stored histogram is dictated by its largest frequency; each cell's frequency represents the apparent surface area of a visible surface at that orientation. This works because regardless of the shape or orientation of an object, a small group of visible surfaces will have the larger frequencies. One of these will correspond to the largest frequency of the model histogram. We only search those areas of the database which correspond to the larger frequencies of the sensed orientation histogram; as a result our search method will quickly find the appropriate area of the database. In practice we find that this simple method reduces the possible search space to approximately 5-6% of the database on average. Our use of the principal axis, p , and average foreshortening factor, h , reduces this still further. The principal axis rejects about half of the remaining histograms whilst h , reduces the remaining records to less than thirty on average. These remaining histograms are compared to the sensed orientation histogram. The alternative to our use of features is a computationally expensive global search and comparison of the database for each viewed object.

Our use of the principal axis is effective because not all visible surfaces whose frequency exceeds the threshold, t_v , have an orientation that lies within $p \pm t_p$. The value of h computed for each model histogram is determined by its orientation. The value of p is determined by the object's 2-D projection on the x-z plane; clearly an object can take many different orientations and still project the same value of p .

Using h to reject many of the remaining records works simply because its value depends upon the azimuth of the axis of symmetry of the object above the x - z plane. We found that in general the sensed value of h is lower than the pre-computed value because noise and sensing errors tend to push triples into adjacent cells of the sensed histogram. One effect of this is to reduce the computed average foreshortening factor.

The value of t_p is set at 20 degrees. We could reduce this tolerance on principal axis to achieve the same results but with fewer comparisons. However in a bin of parts application, objects partially occluded might result in slight errors in the estimated principal axis of the viewed object's 2-D projection. With this in mind all our thresholds are deliberately increased to allow for such possibilities. It should be pointed out that Horn¹² argues that a partially occluded object would not normally be the target object.

By embedding pointers to records in the image plane, we achieve a much more selective retrieval of stored orientation histograms. This method whilst not guaranteeing unique pointers to addresses, is simple and efficient. Over 93% of pointers are unique; less than 1.2% of all the resulting pointers produce clashes involving more than two stored records. Searching for duplicate records when compiling our database is therefore fast and efficient, keeping the number of comparisons to a minimum. An analogy would be a direct file with synonyms.

The method of addressing records by pointers worked so well that the space we allowed for clashes is excessive, and we could have reduced this considerably, allowing more space for further database objects. Each array has 16 elements but the method was so efficient that these could have been reduced to 6 element arrays. This would have released an additional 72 lines of the 16 bit image plane for stored orientation histograms.

A least squares method is used to compare histograms; as the accumulated residual sum exceeds the previous best match the comparison is abandoned. In this way the majority of comparisons are only partial. When the residual sum is smaller than the previous best the current sum becomes the new best match. By comparison a correlation method would require us to fully compute the correlation coefficient, ρ , for each selected histogram and the sensed histogram, i.e all comparisons would need to be complete. We did not use correlation to select a match for this reason, however it is a very useful indicator of the confidence of each match.

10.1 Practical Considerations

An ideal photometric box has a uniform radiance. To achieve this, the box needs to be spherical with a suspended baffle in the centre of the sphere; the object is mounted on the baffle. Behind the baffle the light source is located hidden from both the object and the viewing hole. The interior should be painted with a photometric paint.

Our photometric box, described in *chapter 4*, creates some problems. Firstly the square shape does not create a truly extended source, because there are always variations in reflected light source radiance. These occur because the reflecting surfaces are not equi-distant from the object. The small size of the box, approximately 50 cms. square, means that the 10 cm. diameter viewing hole is large by comparison. For a fixed distance between the camera and object a large box will be nearer the camera. Therefore a smaller hole would suffice. A virtual image of the viewing hole is seen on all objects, *see figure 42*. The two baffles obscuring the light sources can also be seen. The resulting reflectance maps of our objects were consequently distorted in these regions, *see figure 43*.

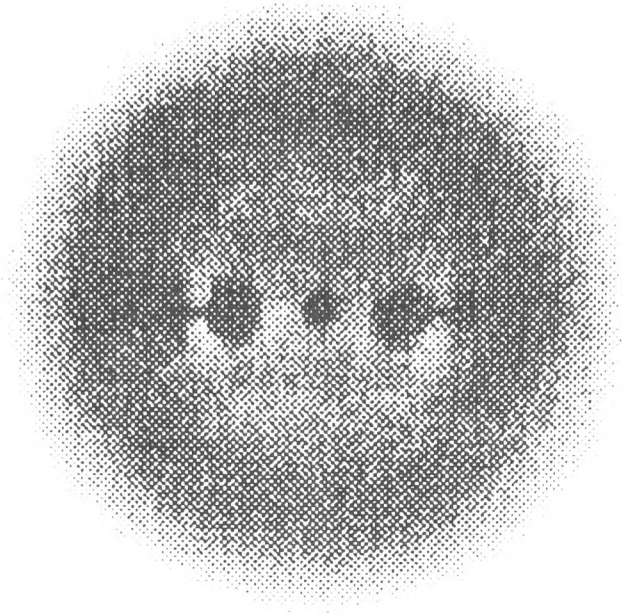


Figure 42. An image of the black calibration sphere. A virtual image of the photometric box is reflected by the sphere. The viewing hole is reflected as a dark spot at the centre of the sphere. The two baffles can also be seen as dark circular patches, one either side of the viewing hole. The lighter square area is due to the light source positions and the square shape of the photometric box.

We use a CCD camera for our images; this gives good geometric accuracy and a linear response to scene radiance. Due to defects in the silicon of CCD chips some sensor cells differ in their response to light. The normalisation of radiance values, *equation (45)*, removes this problem. However we use a non-linear scaling of radiance values, *equation (46)*, and as a result the problem is only partially diminished.



Figure 43. The reflectance contours of the black sphere viewed through a polarising filter. The two baffles and viewing hole distort the contours adjacent to the x-axis.

To ensure orthographic projection the camera should be 'far' from the object; the distance is ideally much greater than the dimensions of the object. In practice this may not be possible, in our case all images are captured from a fixed distance regardless of object size. The camera position is fixed some 1.8 metres from the object. The objects varied in size from the 4.8 cm. diameter calibration sphere, to the cylinder at 3.2 cm. diameter and 5 cm. in length.

10.2 Future Work.

The complex refractive index of many metal surfaces is typically $0.8-2.0i$. They are also highly absorbant, which indicates that the complex part of the refractive index is large. These two factors indicate that it may be possible to extend this method to some metal objects. We experimented with a pair of metal objects both highly specular with a mirror finish. Because so much light is reflected we had difficulty in observing any

polarisation effects. For this reason highly polished metal objects may well be impractical for Polarimetric Stereo.

Different types of surface absorb incident light in varying degrees. Surfaces will absorb more incident light if the light is in a particular area of the spectrum, i.e brass absorbs blue light more strongly than other colours.³⁶ It may be possible to utilise coloured light sources to extend the scope of this method.

The possible application of this technique is limited to surfaces that will produce polarised reflectance contours with sufficient variation to enable an accurate look-up table to be produced. Work needs to be carried out using our model to generate objects with known refractive indices and absorption factors. These will help us to determine combinations of absorption factor and refractive index that will mark the upper and lower limits of this method. At present our model is not accurate enough and we need to implement the refractive index as a complex number.

Currently, the look-up table is produced using frequency of occurrence as the criterion for allocating a triple of radiance to a particular sampling point. A more accurate look-up table may be possible by using a more formal statistical approach to the problem.

We only deal with a single object in the field of view in this application. The method is particularly suited to the bin of parts problem in robotics and as such must be able to deal with multiple objects. A further segmentation is required that will select a candidate from the pile of objects prior to the process described within. We feel that mutual illumination of one object by another will cause problems for our system, however it is a universal problem suffered by other methods such as Photometric Stereo.¹²

No range data is available from this method, and this is a pre-requisite for a bin of parts application.

CHAPTER 11.

Conclusions

We have shown that it is possible to recognise specular dielectric objects, and to determine their orientation, from the analysis of the degree of polarisation of the reflected light. Our method determines the object's orientation to an average uncertainty of less than seven degrees of arc.

Pre-computing a database of viewer-centred model orientation histograms is highly efficient because the process reduces to a simple search and comparison of viewed and model data. We have stored the database on spare image planes, and by embedding pointers to relevant model data in the database itself, the search for a suitable match is reduced to less than one percent of the total database.

References

1. **Ramachandran, V.S.** 'Perceiving Shape from Shading' *Scientific American*, August 1988, pp58-65.
2. **Horn, B.K.P.** 'Understanding Image Intensities' *Artificial Intelligence*, Vol.8, No.2, 1977, pp201-231.
3. **Horn, B.K.P. and Sjoberg, R.W.** 'Calculating the Reflectance Map' *Applied Optics*, Vol.18, No.11, June 1979, pp1770-1779.
4. **Woodham, R.J.** 'Photometric Method for Determining Surface Orientation from Multiple Images' *Optical Engineering*, Vol.19, No.1, Jan/Feb 1980, pp139-144.
5. **Rindfleisch, T.** 'Photometric Method for Lunar Topography' *Photogrammetric Engineering*, Vol.32, No.2, March 1966, pp262-276.
6. **Woodham, R.J.** 'Photometric Stereo: A Reflectance Map Technique for Determining Surface Orientation from Image Intensity' *Image Understanding Systems and Industrial Applications*, Proc. 22nd. SPIE, Vol.155, August 1978, pp136-143.
7. **Horn, B.K.P, Woodham, R.J. & Silver, W.** 'Determining Shape and Reflectance Using Multiple Images' *AI memo 490*, August 1978.
8. **Silver, W.M.** "Determining Shape and Reflectance Using Multiple Images' *MS Thesis*, MIT, June 1980.
9. **Coleman, E.N. and Jain, R.** 'Obtaining 3-Dimensional Shape of Textured and Specular Surfaces using Four-Source Photometry' *Computer Graphics and Image Processing*, Vol. 18, 1982, pp309-328.
10. **Ikeuchi, K.** 'Determining Surface Orientations of Specular Surfaces by Using the Photometric Stereo Method' *IEEE Trans PAMI*- 3, No.6, November 1981, pp661-669.
11. **Ray, R., Birk, J. and Kelley, R.B.** 'Error Analysis of Surface Normals Determined by Radiometry' *IEEE Trans PAMI*-5, No. 6, November 1983, pp631-645.
12. **Horn, B.K.P. and Ikeuchi, K.** 'Picking Parts out of a Bin' *AI 746*, MIT, October 1983.
13. **Brou, P.** 'Using the Gaussian Image to Find the Orientation of Objects' *The International Journal of Robotics Research*, Vol.3, No.4, Winter 1984, pp89-125.

14. **Minkowski, H.** *'Allgemeine Lehrsätze über die Konvexen Polyeder'* Nachrichten von der Koniglichen Gesellschaft der Wissenschaften, mathematisch-physikalische Klasse, Gottingen, pp198-219, 1897.
15. **Bajcsy, R.** *'Three-Dimensional Scene Analysis'* Proc. Pattern Recognition Conf., Miami, 1980, pp1064-1074.
16. **Horn, B.K.P.** *'Sequins and Quills - Representations for Surface Topography'* AI memo 536, MIT, 1979.
17. **Horn, B.K.P.** *'Extended Gaussian Images'* AI memo 740, MIT, July 1983.
18. **Ikeuchi, K.** *'Recognition of Objects Using the Extended Gaussian Image'* Proc. IJCAI-81 (Vancouver), 1981, pp595-600.
19. **Ikeuchi, K. & Shirai, Y.** *'A Model Based Vision System for Recognition of Machine Parts'* Proc. AAAI-82, Pittsburgh, 1982, pp18-21.
20. **Horn, B.K.P.** *Robot Vision*, The MIT Press, 1986.
21. **Konnen, G.P.** *Polarised Light in Nature*, Cambridge University Press, 1985.
22. **Koshikawa, K. & Shirai, Y.** *'A Model Based Recognition of Glossy Objects using their Polarimetrical Properties'* JRSJ, Vol.3, No.1, 1985, pp4-9.
23. **Terashi, Y., Shirai, Y. & Koshikawa, K.** *'Recognition of Surface Geometry of Glossy Objects Using Polarised Light'* JRSJ, Vol.3, No.1, December 1985, pp86-95. (In Japanese)
24. **Koshikawa, K.** *'A Polarimetric Approach to Shape Understanding of Glossy Objects'* Proc. IJCAI-79, 1979, pp493-495.
25. **Brooks, M.J.** *'Shape from Shading Discretely'* Ph.D Thesis, Essex University, September 1982.
26. **Jenkins, F.A. and White, H.E.** *Fundamentals of Optics*, McGraw- Hill, 1967.
27. **Clarke, D. and Grainger, J.F.** *Polarised Light and Optical Measurement*, Pergamon Press, 1971.
28. **Levi, S.** *Fundamentals of Computer Vision*, Edited by O.D. Faugeras, Cambridge University Press, 1983.
29. **Marr, D.** *Vision*, W.H. Freeman & Co., 1982.

30. **Ming-Kuei Hu.** '*Visual Pattern Recognition by Moment Invariants*' IRE trans. IF, IT-8, February 1962, pp179-187.
31. **McGhee, J.W.** *Introductory Statistics*, West Publishing Co., 1985.
32. **Fairney,P.T., Jones,B.F. and Wiltshire,R.J.** '*Use of Viewer- Centred Descriptions of Objects to Improve the Computational Efficiency and Accuracy of Photometric Stereo*' Eng. Appl. of AI, vol.1, June 1988, pp111-118.
33. **Porrill,J.,Pollard,S.B.,Pridmore,T.P.,Bowen,J.B.,Mayhew,J.E.W. & Frisby,J.P.** '*TINA: A 3D Vision System for Pick and Place*' Proc. AVC87, pp65-72, 1987.
34. **Porrill,J.,Pollard,S.B. & Mayhew,J.E.W.** '*The Optimal Combination of Multiple Sensors Including Stereo Vision*' Proc. AVC86, 1986.
35. **Blake,A., Zisserman,A. & Mclauchlan,P.** '*A Knowledge Source for Describing Stereoscopically Viewed Textured Surfaces*' Proc. AVC86, 1986.
36. **Kaye,G.W.C. & Laby,T.H.** *Tables of Physical and Chemical Constants and Some Mathematical Functions*, Eleventh Edition, Longmans, Green & Co., London, 1956.
37. **Ikeuchi,K.** '*Determining Attitude of Object from Needle Map using Extended Gaussian Image*' AI memo 714, MIT, April 1983.