Computational Vision
U. Minn. Psy 5036
Daniel Kersten
Lecture 17: Shape from shading cont'd

http://vision.psych.umn.edu/www/kersten-lab/courses/Psy5036/SyllabusF2000.html)

Initialize

Outline

Last time

- **Inference of shape from shading--set in context of other cues to shape,**

- **Perception of shape from shading**

- **Introduce models for the inverse problem "shape from shading":**

Inverting the generative model: L(pn,qn) =**n.e**= $\frac{\{pn,qn,-1\}.\{pe,qe,-1\}}{\sqrt{(pn^2+qn^2+1)(pe^2+qe^2+1)}}$

Classic Ikeuchi & Horn solution:

Assume light source direction is known. Constant reflectance. Known surface normals at the smooth occluding boundary.

Find p's and q's such that E_D+E_S is small.

Generative model -> data cost function E_D(p(x,y), q(x,y))

Prior assumption of surface smoothness

(i.e. spatial second derivatives of p(x,y) and q(x,y) should be small) -> data-independent cost function $E_S((\nabla^2 p)^2, \nabla^2 q^2)$.

Equivalent to Bayes.

Today

- **Perceptual ambiguities continued**

- **Formal ambiguities seen by analyzing the generative model, the bas-relief transform**

Solutions to the shape from shading problem for the linear case

Learning scene parameters from images

Last time I mentioned that if one has available a set of images $\{L_i\}$, together with a set of unit surface normals for each image $\{p_i, q_i\}$(e.g. derived from a set of range images), then a linear model suggests that one could simply do a linear regression to find a map: L -> $\{p, q\}$, that might be a workable approximation to shape from shading from an image L that was not in the training class $\{L_i\}$.

This was done by Knill and Kersten (1990). They used a neural network-like technique (Widrow-Hoff learning rule for synaptic modification) to do regression learning "on-line", rather than in "batch mode". This method was general and empirical in the sense, that one can imagine applying it to other scene-from-image problems (Kersten et al., 1988). How well it works depends on the linear approximation. Non-linear neural networks and learning techniques such as back-propagation can be used to find solutions for non-linear formulations (Lehky & Sejnowski, 1988).

Pentland's solution to the inverse problem

- **Recall from last time**

Last time we derived a linear approximation to the lambertian shading model, L= **n.e.** The image luminance is given by

L(pn,qn) =**n.e**= $\frac{\{pn,qn,-1\}.\{pe,qe,-1\}}{\sqrt{(pn^2+qn^2+1)(pe^2+qe^2+1)}}$,

where n={pn,qn,-1}/$\sqrt{(pn^2+qn^2+1)}$ and e={pe,qe,-1}/$\sqrt{pe^2+qe^2+1)}$ are the unit surface normal vector and illumination vector respectively.

Using a Taylor series expansion about {pn,qn}={0,0}, we were able to derive a linear approximation to the shading equation.

```
Lmodel[pn_,qn_,pe_,qe_]:= ({pn,qn,-1}/Sqrt[pn^2+qn^2 +
1]).({pe,qe,-1}/Sqrt[pe^2+qe^2 + 1]);
Series[Lmodel[pn,qn,pe,qe], {pn, 0, 1}, {qn, 0, 1}]
```

$$\left(\frac{1}{\sqrt{1+pe^2+qe^2}}+\frac{qe\,qn}{\sqrt{1+pe^2+qe^2}}+O[qn]^2\right)+\left(\frac{pe}{\sqrt{1+pe^2+qe^2}}+O[qn]^2\right)pn+O[pn]^2$$

```
Lapprxom[x_,y_] := (1+qe*qn[x,y] + pe*pn[x,y])/Sqrt[1+pe^2 + qe^2];
```

■ Using the fourier transform to estimate surface depth z, from image intensity L

A standard result from fourier transform theory is, given g(x) and its fourier transform Fg(fx), then it is easy to write down the fourier transform of the derivative of g(x). It is just the fourier transform of g(x) times 2πifx. So, FourierTransform[$\frac{dg}{dx}$] = 2πifx Fg(fx).(cf. Gaskill).

Let z(x,y) be any differentiable function. Let Fz[fx,fy] be the fourier transform of z, which can be written in terms of the (complex) amplitude and phases as:

a[fx,fy] $e^{i\phi(fx,fy)}$. Using the above result, we have:

Fp[fx,fy] = 2π i fx a(fx,fy) $e^{i(\phi(fx,fy))}$ and Fq[fx,fy] = 2π i fy a(fx,fy) $e^{i(\phi(fx,fy))}$.

Let's simplify L = (1+qe*qn[x,y] + pe*pn[x,y])/Sqrt[1+pe^2 + qe^2] by ignoring the constant term 1.

Let's express the direction of the light source in terms of tilt and slant. Then $\tau=\tan^{-1}(q_e/p_e)$, and $\sigma=\tan^{-1}\left(\sqrt{q_e^{\,2}+p_e^{\,2}}\right)$.

So we can write L = cosσ + p sinσ + q sinτ sin σ. Let the fourier transform of the image function L be: a_L[fx,fy] $e^{i\phi_L(fx,fy)}$. With a little algebra, one can show that:

$$\mathrm{Fz(fx,fy)} = \frac{a_L(\mathrm{fx,fy})\,e^{i(\phi_L(\mathrm{fx,fy})-\pi/2)}}{2\,\pi\,\sin\sigma\ \mathrm{fx}(\mathrm{cosfy}\cos\tau+\mathrm{sinfy}\sin\tau)}$$

The solution requires known point light source, constant reflectance. Further the linear approximation is good for small p's and q's --i.e. for shallow bas-relief like shapes.

The inverse fourier transform provides an estimate of the surface depth z. Note that if ignore the constant term in image luminance, we can replace z-> z + k1x + k2 y + k3, we obtain the same solution.

Shortly, we will look at recent results that show how general shape ambiguity is when the light source is unknown.

Shading & the bas-relief transform

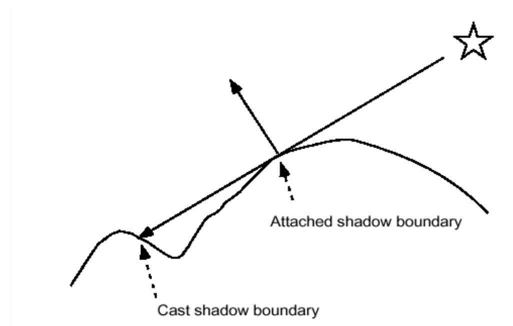
Generalizing the simple lambertian generative model

The lambertian model thus far is pretty limiting. It doesn't take into account *attached* or cast shadows or multiple light sources. An attached shadow occurs whenever the surface normal is at an angle greater than 90 degrees relative to the light source vector. The cosine of an oblique angle is negative, but we can't have negative light. If there is no ambient light, then an attached shadow region is black. So a better model would be:

L = Max[**n.e**,0]. We may have more than one light source, and since light is nice and linear, we can just add up all the contributions:

L = $\sum_i$ Max[$n.e_i$, 0]

A cast shadow boundary occurs on a receiving surface whenever another surface (or part of one) is between the receiving surface and the light source.



The bas-relief effect

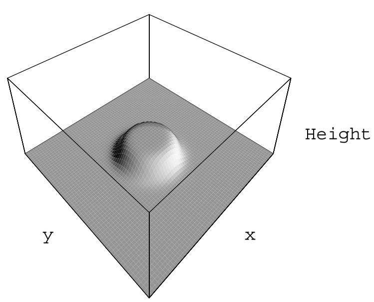
Recently, Belhumeur, P. N., Kriegman, D. J., & Yuille (1997) showed that there is a simple transformation of a surface (i.e. adding a slanted and tilted plane to a surface z(x,y) and a compression in depth) that induces (together with an albedo and illumination adjustment) an well-defined equivalence class of surfaces for a given image. For each image of a lambertian surface z(x,y), there is an identical image of a bas-relief produced by a transformed light source. Further this holds for both shaded and shadowed regions. And for the classical bas-relief in sculpture (no added slant, just compression), the image is insensitive to small motions and illumination changes.

Figure 1: Frontal and side views of a pair of marble bas-relief sculptures: Notice how the frontal vie appear to have full 3-dimensional depth, while the side views reveal the flattening the sculptu rise only 5 centimeters from the background plane. While subtle shading is apparent on the fac the shadows on the women's pleats are the dominant perceptual cue in the body.

First surface

■ Define First surface bump

```
bump[x_, y_] := (1 / 4) (1 - 1 / (1 + Exp[-10 (Sqrt[x^2 + y^2] - 1)]));
g1 = Plot3D[bump[x, y], {x, -3, 3}, {y, -3, 3},
   PlotPoints → 64, PlotRange → {0, 1}, Mesh → False,
   LightSources → {{{1., 0., 1.}, RGBColor[1, 0, 0]},
     {{1., 0., 1.}, RGBColor[0, 1, 0]}, {{1., 0., 1.}, RGBColor[0, 0, 1]}},
   ViewPoint → {1, 1, 1}, AxesLabel → {"x", "y", "Height"}, Ticks → False,
   AspectRatio → 1, PlotRange → {{-3, 3}, {-3, 3}, {0, 1}}];
```



■ Calculate surface normals of first surface bump

```
nx[x_, y_] := Evaluate[D[bump[x, y], y]];
ny[x_, y_] := Evaluate[D[bump[x, y], x]];
```

■ Rendering specification for normals, light, reflectance first surface bump

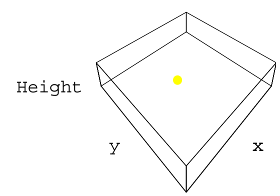
■ Unit surface normals

```
normbump[x_, y_] := {nx[x, y], ny[x, y], 1} / Sqrt[nx[x, y]^2 + ny[x, y]^2 + 1];
```

■ Point Light source

```
s = {1, 0, 1} / Sqrt[3];
```

```
g2 = ScatterPlot3D[{s}, ViewPoint → {1, 1, 1},
   AxesLabel → {"x", "y", "Height"}, Ticks → False,
   PlotStyle → {PointSize[.05], {RGBColor[1, 1, 0]}},
   AspectRatio → 1, PlotRange → {{-3, 3}, {-3, 3}, {0, 1}}];
```



■ Reflectance

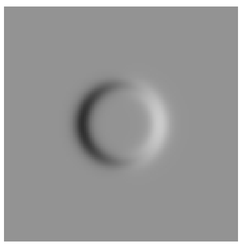
```
a[x_, y_] := 1;
```

- **Rendering equation for first surface bump**

```
imagebump[x_, y_] := a[x, y] * normbump[x, y].s;
```

- **Render first surface bump**

```
DensityPlot[imagebump[x, y], {x, -3, 3}, {y, -3, 3},
  Mesh → False, Frame → False, PlotPoints → 128, PlotRange → {0, 1}];
```



A second surface

- **Bas relief transform, G : Second surface**

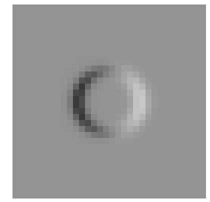
```
basreliefG[λ_, μ_, ν_] := {{λ, 0, -μ}, {0, λ, -ν}, {0, 0, 1}};
```

```
λ = .25; μ = .125; ν = 0;
G = basreliefG[ λ, μ, ν];
iGt = Inverse[Transpose[G]];
```

```
basreliefnormbump[x_, y_] := G.(a[x, y] * normbump[x, y]);
s2 = iGt.s;
a2[x_, y_] := Sqrt[basreliefnormbump[x, y].basreliefnormbump[x, y]];
normbump2[x_, y_] := basreliefnormbump[x, y] /
   Sqrt[basreliefnormbump[x, y].basreliefnormbump[x, y]];
```

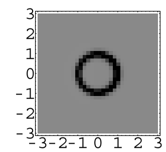


```
imagebump2[x_, y_] := a2[x, y] * normbump2[x, y].s2;
DensityPlot[imagebump2[x, y], {x, -3, 3}, {y, -3, 3},
  Mesh → False, Frame → False, PlotPoints → 32, PlotRange → {0, 1}];
```



- **Plot reflectance**

```
DensityPlot[a2[x, y], {x, -3, 3}, {y, -3, 3},
  PlotPoints → 32, Mesh → False, PlotRange → {.9, 1.1}];
```

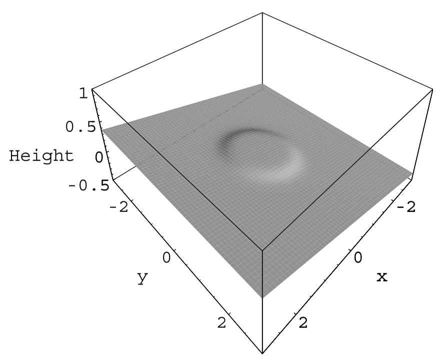


- **Plot new bas-relief surface height**

(*brpointbump[x_,y_]:=G.{x,y,bump[x,y]};*)

```
brbump[x_, y_] := λ * bump[x, y] + μ * x + ν * y;
```

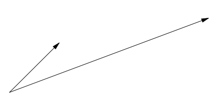
```
Plot3D[brbump[x, y], {x, -3, 3}, {y, -3, 3},
  PlotPoints → 64, PlotRange → {-.5, 1}, Mesh → False,
  LightSources → {{{1., 0., 1.}, RGBColor[1, 0, 0]},
    {{1., 0., 1.}, RGBColor[0, 1, 0]}, {{1., 0., 1.}, RGBColor[0, 0, 1]}},
  ViewPoint → {1, 1, 1}, AxesLabel → {"x", "y", "Height"},
  AspectRatio → 1, PlotRange → {{-3, 3}, {-3, 3}, {0, 1}}];
```



Compare first and second surfaces

■ Plot light source directions for new and old

```
units = Drop[s, {2}];(*units=units/Sqrt[units.units];*)
units2 = Drop[s2, {2}]; (*units2=units2/Sqrt[units2.units2];*)
glight1 = ListPlotVectorField[{ {{0, 0}, units2}, {{0, 0}, units}},
  Ticks → False, Axes → False, PlotRange → {{-1.5, 2.5}, {-.25, 1}}];
```

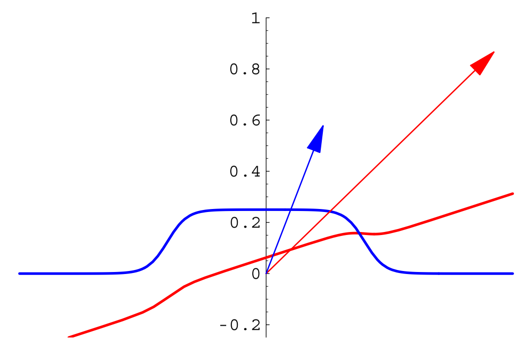



```
units2.units2
```

```
1.
```

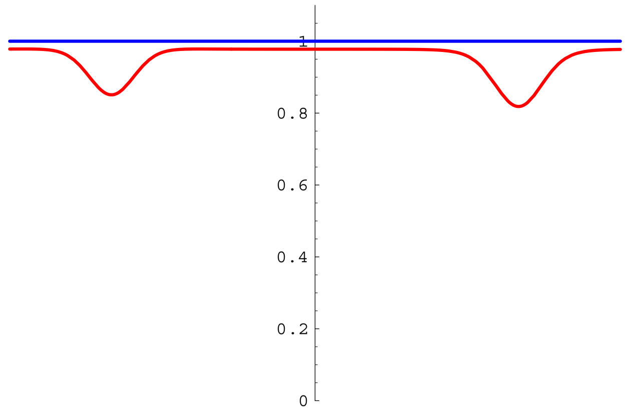
■ Plot surface height cross-sections of new and old

```
gheight = Plot[{brbump[x, 0], bump[x, 0]}, {x, -2.5, 2.5},
   PlotRange → {-.25, 1}, PlotStyle → {{Thickness[.005], RGBColor[1, 0, 0]},
     {Thickness[.005], RGBColor[0, 0, 1]}}, Ticks → True,
   Axes → {False, True}, Epilog → {{RGBColor[1, 0, 0], Arrow[{0, 0}, units2]},
     {RGBColor[0, 0, 1], Arrow[{0, 0}, units]}}];
```



■ **Plot reflectance cross-sections for new and old**

```
greflectance = Plot[{a2[x, 0] - 0.03, a[x, 0]}, {x, -1.5, 1.5},
  PlotRange → {0, 1.1}, PlotStyle → {{Thickness[.005], RGBColor[1, 0, 0]},
    {Thickness[.005], RGBColor[0, 0, 1]}}, Axes → {False, True}]
```



Task analysis: "Shape for X"

■ **Common shape representation vs. task-dependent?**

■ **Shape for object recognition**

■ **Shape for grasp**

■ **Essence of Bayesian approach to task analysis:**

Primary variables to estimate explicitly. Secondary (or generic) variables that contribute to the image data, but which should be discounted. In shape from shading, we considered surface normals to be the primary variables, and the illumination to be the secondary variable. Suppose we don't know the light source direction AND we don't want to estimate it explicitly. Somehow we need to "discount" the illumination.

"Discount" in Bayesian terms means to integrate out their contributions to the posterior probability. So suppose that Sprim = {surface normals}, and Ssec = {light source direction}. Further, suppose we have a model that prescribes p(Sprim,Sec | I), where I is the image measurement (e.g. I = luminance L). Then if we can do the following integral,

$$p(S_{prim} \mid I) = \int p(S_{prim}, S_{\sec} \mid I) dS_{\sec}$$

we could then base our decision on p(Sprim | I), where the Ssec is no longer explicit, but its effects have gotten folded into the posterior p(Sprim | I). Bayes rule can be used to express the joint posterior in terms of the likelihood and the prior:

$$p(S_{prim}, S_{\sec} \mid I) \propto p(I \mid S_{prim}, S_{\sec}) p(S_{prim}, S_{\sec})$$

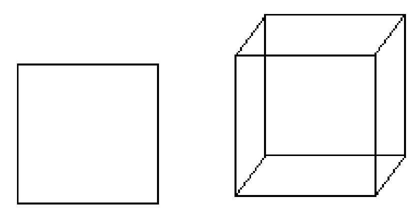
The lambertian shading equation is sufficient to determine the likelihood term. As we've seen, usually one assumes a known light source direction (Ssec), and then the prior is some measure that ranks the probability of surfaces based on how smooth or rought they are.

A few years ago, Bill Freeman at Mitsubishi Electric showed that this process of "integrating out" or "marginalization" can actually act to disambiguate the shape in the shape from shading problem. In other words, by assuming up front that illumination direction is a secondary variable, one can find solutions to the shape from shading problem with subtantially less reliance on a priori smoothness constraint.

Amibiguity reduction using task constraints: "genericity"

General viewpoint constraint (Lowe). Why is the figure on the left seen as a square rather than as a cube?

■ **Cube**



■ **Penrose triangle**

The following example uses figures from: http://www.klab.caltech.edu/~seckel/triangle.html

Generic view or "general viewpoint constraint" is so strong that the human visual system can sometimes prefer impossible to possible objects (e.g. Penrose triangle).

Is there a real object that corresponds to this image?

http://www.klab.caltech.edu/~seckel/triangle.html

■ General principle?

Perception's model of the explicit variables in a scene should be robust to variations in the generic or secondary variables

Freeman's solution to shape from shading: "General illumination direction constraint"

■ Shape & illumination direction ambiguity

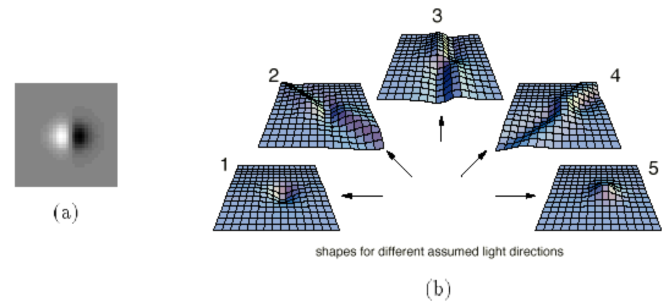


Figure 6: (a) Perceptually, this image has two possible interpretations. It could be a bump, lit from the left, or a dimple, lit from the right. (b) Mathematically, there are many possible interpretations. For a sufficiently shallow incident light angle, if we assume different light directions, we find different shapes, each of which could account for the observed image.

Imagine wiggling the light source--which interpretation gives the smallest image variation? Freeman showed that the circularly symmetric bump interpretation gave the least variation, and was thus the most probable interpretation.

■ Lambertian vs. shiny ambiguity

Here is another example from Freeman.

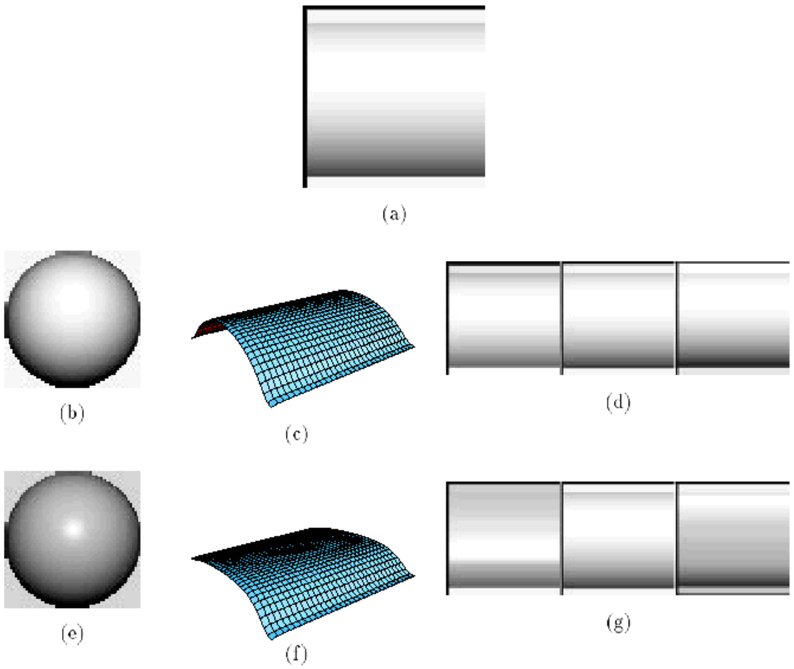


Figure 2: The image (a) appears to be a cylinder (c) painted with a Lambertian reflectance function (b) (shown on a hemisphere). However the flatter shape of (f) and a shiny reflectance function (e) also explain the data equally well. We can distinguish between the competing accounts for (a) by imagining rotating each shape. Images of each shape at three nearby orientations are shown in (d) and (g). We see that the image made assuming a Lambertian reflectance function (b) is more stable than that made assuming a shiny reflectance function (e). The reflectance function of (b) provides more angles over which the image looks nearly the same. If all viewpoints are equally likely, and the shapes and reflectances of (b-c) and (e-f) are equally likely to occur in the world, then (b-c) is a more probable interpretation than (e-f).

Now imagine wiggling the shape a little (or wiggle the light source). The biggest variations are shown in (g). Smaller variations are seen in the (d) images. Thus the more probable scene interpretation is (b, c), rather than (e,f). (Figure from: Mitsubishi Tech Report, TR93-15. http://www.merl.com/people/freeman/publications.html. See too: W. T. Freeman, Exploiting the generic viewpoint assumption, International Journal Computer Vision, 20 (3), 243-261, 1996..) (See too: Kersten (1999) for an example from depth from shadows).

Freeman showed for several other problems (motion disambiguation) that marginalizing out the generic variable (secondary) can peaken the prior probability for the explicit (primary) variable, thereby constraining the shape (or other) estimates of the explicit variable. Further, he showed that under certain conditions, this marginalization is equivalent to the robustness principle above: *Perception's model of the explicit variables in a scene should be robust to variations in the generic or secondary variables*.

More on local shape representation

■ **Metric. vs. qualitative**

■ **View-point vs. object-centered: extrinsic vs. intrinsic shape descriptors**

Curvature of a line.

Curvature of a surface. Principal curvatures. Gaussian curvature--elliptic (+), hyperbolic (-), cylindrical & flat points (0).

Next time

Introduction to motion analysis

Appendices

■ **(under construction) we can also plot the unit surface normals on the surface itself in 3D**

References

Belhumeur, P. N., Kriegman, D. J., & Yuille, A. (1997). *The Bas-Relief Ambiguity*. Paper presented at the IEEE Conf. on Computer Vision and Pattern Recognition.

Brady, M., & Yuille, A. (1983). An Extremum Principle for Shape from Contour. *Proceedings IJCAI*, 969-972.

Bülthoff, H. H., & Yuille, A. (1990). Shape-from-X: Psychophysics and computation. *Proceedings of the SPIE, 1383*, 165-172.

Dupuis, P., & Oliensis, J. (1994). An optimal control formulation and related numerical methods for a problem in shape reconstruction. *The Annals of Applied Probability, 4*(No, 2), 287-346.

Freeman, W. T. (1994). The generic viewpoint assumption in a framework for visual perception. *Nature, 368*(7 April 1994), 542-545.

Gray, Alfred. Modern Differential Geometry of Curves and Surfaces with Mathematica, Second Edition. (CRC Press, 1997) (http://store.wolfram.com/view/ISBN0849371643/?39FED89E-38F0)

Horn, B., & Brooks, M. J. (1989). *Shape from shading*. Cambridge, Mass.: MIT Press.

Ikeuchi, K., & Horn, B. K. P. (1981). Numerical shape from shading and occluding boundaries. *Art. Intel., 17,*, 141-184.

Kersten, D., O'Toole, A. J., Sereno, M. E., Knill, D. C., & Anderson, J. A. (1987). Associative learning of scene parameters from images. *Appl. Opt., 26*, 4999-5006.

Knill, D. C., & Kersten, D. (1990). Learning a near-optimal estimator for surface shape from shading. *CVGIP, 50*(1), 75-100.

Knill, D. C. (1992). Perception of surface contours and surface shape: from computation to psychophysics. *J Opt Soc Am [A], 9*(9), 1449-1464.

Koenderink, J. J. (1998). Pictorial relief. *Philosophical Transactions of the Royal Society of London Series a-Mathematical Physical and Engineering Sciences, 356*(1740), 1071-1085.

Koenderink, J. J. (1990). *Solid Shape*. Cambridge, MA: MIT Press.

Koenderink, J. J., van Doorn, A. J., & Kappers, A. M. (1992). Surface perception in pictures. *Perception & Psychophysics, 5*, 487-496.

Koenderink, J.-J., Kappers, A., & van Doorn, A. (1992). Local Operations: The Embodiment of Geometry. *Artificial and Biological Vision Systems*.

Langer, M. S., & Zucker, S. W. (1994). Shape from Shading on a Cloudy Day. *Journal of the Optical Society of America A, 11*(2), 467-478.

Lehky, S. R., & Sejnowski, T. J. (1988). Network model of shape-from-shading: neural function arises from both receptive and projective fields. Nature, 333, 452-454.

Mamassian, P., Kersten, D., & Knill, D. C. (1996). Categorical local-shape perception. *Perception, 25*(1), 95-107.

Mamassian P & Kersten D. (1996). Illumination, shading and the perception of local orientation. *Vision Research., 36*(15), 2351-2367.

Mingolla, E., & Todd, J. T. (1986). Perception of Solid Shape from Shading. *Biological Cybernetics, 53*, 137-151.

Oliensis, J. (1991). Uniqueness in Shape From Shading. *The International Journal of Computer Vision, 6*(2), 75-104.

Pentland, A. P. (1990). Linear shape from shading. *International Journal of Computer Vision, 4*, 153-162.

Pentland, A. P. (1988). *Shape Information from Shading: A Theory About Human Perception*.

TR-103. Paper presented at the M.I.T. Media Lab Vision Sciences.

Reichel, F. D., Todd, J. T., & Yilmaz, E. (1995). Visual-Discrimination of Local Surface Depth and Orientation. *Perception & Psychophysics, 57*(8), 1233-1240.

Todd, J. T., & Mingolla, E. (1983). Perception of Surface Curvature and Direction of Illumination from Patterns of Shading. *Journal of Experimental Psychology: Human Perception & Performance, 9*(4), 583-595.

Todd, J. T., Norman, J. F., Koenderink, J. J., & Kappers, A. M. L. (1997). Effects of texture, illumination, and surface reflectance on stereoscopic shape perception. *Perception, 26*(7), 807-822.