Chapter 07.00D

Physical Problem for Integration Computer Engineering

Problem

Human vision has the remarkable ability to infer 3D shapes from 2D images. When we look at 2D photographs or TV we do not see them as 2D shapes, rather as 3D entities with surfaces and volumes. Perception research has unraveled many of the cues that are used by us. The intriguing question is: can we replicate some of these abilities on a computer? To this end, in this assignment we are going to look at one way to engineer a solution to the 3D shape from 2D images problem. Apart from the pure fun of solving a new problem, there are many practical applications of this technology such as in automated inspection of machine parts, inference of obstructing surfaces for robot navigation, and even in robot-assisted surgery.

Image is a collection of gray level values at a set of predetermined sites known as pixels, arranged in an array. These gray level values are also known as image intensities. The registered intensity at an image pixel is dependent on a number of factors such as the lighting conditions, surface orientation, and surface material properties. The nature of lighting and its placement can drastically affect the appearance of a scene. In another module on simultaneous linear equations, we saw how to infer the surface normal vectors for each point in the scene, given three images of the scene taken with three different light sources. In Figure 1 we see vector field that we have inferred form the three images. In this module, we will see how we can *integrate this vector field to arrive at a surface.*

Physics of the Problem

To be able to reconstruct the shape of the underlying surface, we have to first understand the how the surface normal is related to the underlying surface. Figure 2 shows the schematic of the camera centered coordinate axis that we can use to formulate the problem. The z-direction is away from the camera towards the scene. Let the scene surface be parameterized by the function.

$$z = g(x, y)$$

The equation of the local surface normal can be related to this function as follows. If we find two tangents on the surface then the cross product of these tangents will give us the surface normal. Two tangents along the x and y-directions can easily be specified in terms of the derivatives along these directions. Figure 2 shows the underlying geometry that can be used to arrive at these equations.

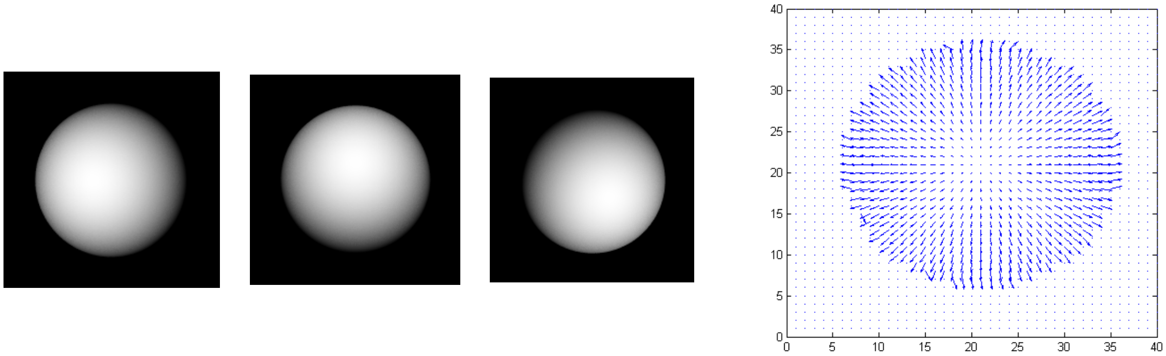

Figure 1 The first three images are of a sphere taken with three different light source positions. The right image is a vector field representation of the surface normal vectors estimated from these three images. Can you compute the underlying surface representation from this vector field?

$$\vec{t}_x = \begin{bmatrix} 1 \\ 0 \\ \dfrac{\partial g(x,y)}{\partial x} \end{bmatrix} \quad \text{and} \quad \vec{t}_y = \begin{bmatrix} 0 \\ 1 \\ \dfrac{\partial g(x,y)}{\partial y} \end{bmatrix}$$

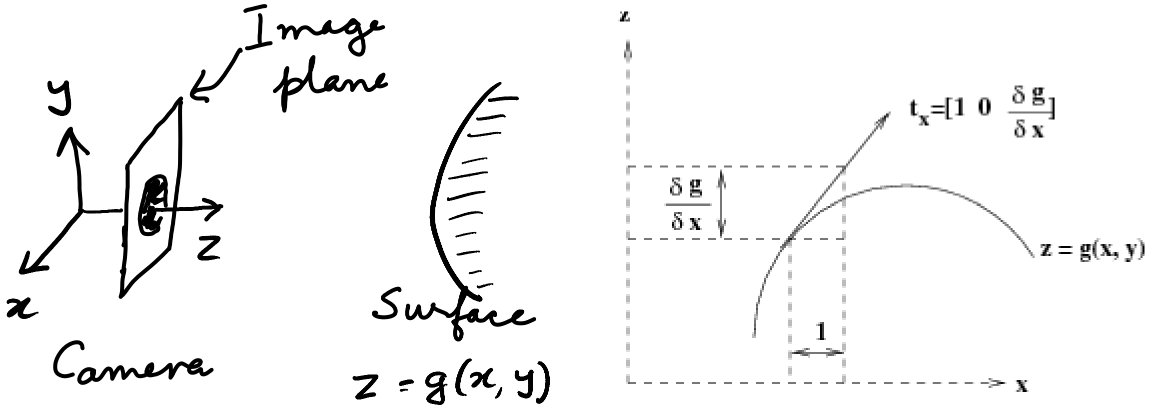

Figure 2 Relationship between surface tangent at a point to underlying surface equation. On the left is a simplified representation of the imaging geometry. The surface is assumed to be far away from the camera. On the right is a schematic of the local geometry on the scene surface.

The local surface normal, n, will be along the direction of the cross product of t_x and t_y.

$$c\vec{n} = -\vec{t}_x \times \vec{t}_y = \begin{bmatrix} \dfrac{\partial g(x,y)}{\partial x} \\ \dfrac{\partial g(x,y)}{\partial x} \\ -1 \end{bmatrix}$$

Note that there is 1800 ambiguity in the specification of the surface normal. We need the negative sign because we want the surface normal to be oriented towards the camera and the z-axis is pointed away from the camera. Also, note that the cross product gives as a vector along the surface normal. We have to normalize the vector to arrive at the surface normal. For notational ease let

$$p = \frac{\partial g(x,y)}{\partial x} \text{ and } q = \frac{\partial g(x,y)}{\partial y}$$

Then,

$$\vec{n} = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} = \frac{1}{\sqrt{p^2 + q^2 + 1}} \begin{bmatrix} p \\ q \\ -1 \end{bmatrix}$$

We see that the surface normal can be expressed in terms of the derivatives of the underlying surface. From this equation, we can express the local surface derivatives in terms of ratios of the surface normal components.

$$\frac{\partial g(x,y)}{\partial x} = \frac{n_x}{n_y} \qquad \text{and} \qquad \frac{\partial g(x,y)}{\partial y} = \frac{n_y}{n_z}$$

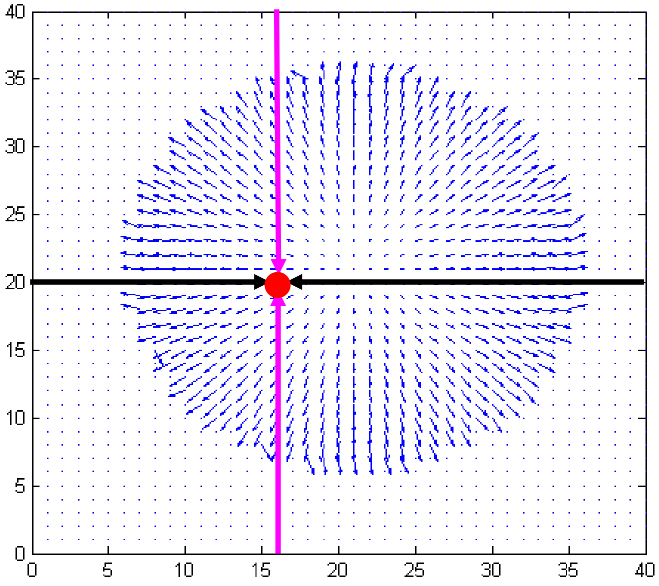

Figure 3 Integration paths to arrive at the depth surface value for the point marked by the red dot.

The black lines denote possible paths over which the x-partial derivative can be integrated. And, the purple lines denote possible paths to integrate the y-partial derivative. The paths are shown overlaid on the input vector field representing the partial derivatives of the surface along x and y directions.

Solution

We arrive at the final surface equation by integration these partial derivatives. Since we have a 2D field, we can perform the integration along many paths. The simplest paths are along the x (or y) axis.

$$g(u,v)=\int_{o}^{u}\frac{\partial g(x,v)}{\partial x}dx \qquad \text{or} \qquad g(u,v)=\int_{o}^{v}\frac{\partial g(u,y)}{\partial y}dy$$

One could also start from the right edge of the image and integrate the partial derivative with respect to x towards the left edge, or start from the bottom of the image and integrate the y-partial derivative towards the top. These directions are depicted in

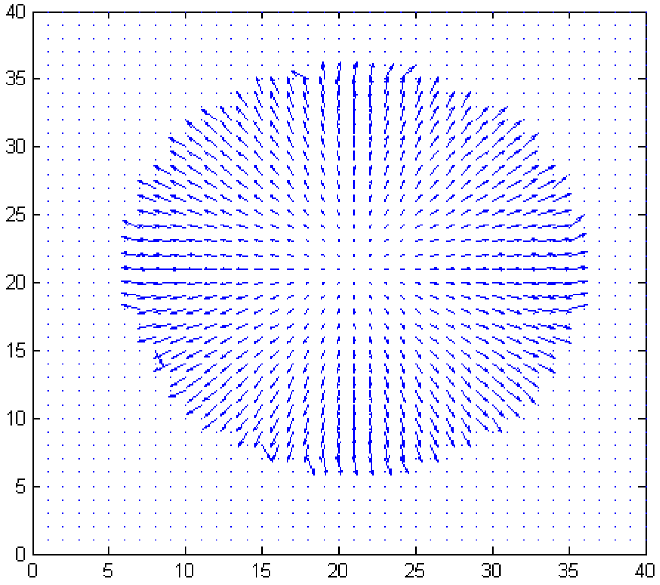

Figure 3 Integration paths to arrive at the depth surface value for the point marked by the red dot.

. The equivalent integrals are as follows. The image is assumed to N pixels by M pixels in size and the negative sign arises because of the direction of the integration.

$$g(u,v)=-\int_{N}^{u}\frac{\partial g(x,v)}{\partial x}dx \qquad \text{or} \qquad g(u,v)=-\int_{M}^{v}\frac{\partial g(u,y)}{\partial y}dy$$

Ideally, all the four integrals should result in the same value, however, for noisy, real world data they will never be the same. One solution is to take the average of these four integrals as the final value. Note that these integrals have to be evaluated for each location (u,v) in the image to arrive at the full surface.

Worked Out Example

Consider the problem of estimating the surface height along the line passing through the center of the sphere in Figure 1. Figure 4 (a) shows the input estimates for the partial derivatives along the x direction, $p = \frac{\partial g(x,y)}{\partial x}$ for the sphere along this line. The raw data for this plot is given below.

x	g
1.00000	0.00000
2.00000	0.00000
3.00000	0.00000
4.00000	0.00000
5.00000	0.00000
6.00000	0.00000
7.00000	0.00000
8.00000	0.00000
9.00000	0.00000
10.00000	0.00000
11.00000	0.00000
12.00000	0.00000
13.00000	0.00000
14.00000	0.00000
15.00000	0.00000
16.00000	0.00000
17.00000	0.00000
18.00000	0.00000
19.00000	0.00000
20.00000	0.00000
21.00000	0.00000
22.00000	0.00000
23.00000	0.00000
24.00000	0.00000
25.00000	0.00000
26.00000	0.00000
27.00000	0.00000
28.00000	0.00000
29.00000	0.00000
30.00000	5.42857
31.00000	4.40000
32.00000	3.82362
33.00000	3.23097
34.00000	2.74844
35.00000	2.43579
36.00000	2.19293
37.00000	2.04677

38.00000	1.91851
39.00000	1.88649
40.00000	1.62019
41.00000	1.55007
42.00000	1.47428
43.00000	1.44755
44.00000	1.36134
45.00000	1.28187
46.00000	1.14868
47.00000	1.15089
48.00000	1.14027
49.00000	1.05379
50.00000	0.98459
51.00000	1.00329
52.00000	0.96249
53.00000	0.90161
54.00000	0.88597
55.00000	0.83243
56.00000	0.78901
57.00000	0.80149
58.00000	0.74730
59.00000	0.73934
60.00000	0.70364
61.00000	0.67288
62.00000	0.66161
63.00000	0.62773
64.00000	0.61402
65.00000	0.60783
66.00000	0.58271
67.00000	0.55106
68.00000	0.52999
69.00000	0.50641
70.00000	0.50146
71.00000	0.46053
72.00000	0.45829
73.00000	0.43148
74.00000	0.41627
75.00000	0.39025
76.00000	0.38186
77.00000	0.35635
78.00000	0.34187
79.00000	0.32304
80.00000	0.31064
81.00000	0.27967
82.00000	0.27367

83.00000	0.26137
84.00000	0.24340
85.00000	0.23128
86.00000	0.20758
87.00000	0.19566
88.00000	0.19566
89.00000	0.17807
90.00000	0.16632
91.00000	0.14409
92.00000	0.12686
93.00000	0.11039
94.00000	0.10481
95.00000	0.08266
96.00000	0.06623
97.00000	0.04945
98.00000	0.04383
99.00000	0.03309
100.00000	0.00038
101.00000	-0.00516
102.00000	-0.02149
103.00000	-0.03232
104.00000	-0.04317
105.00000	-0.05421
106.00000	-0.06513
107.00000	-0.09250
108.00000	-0.09820
109.00000	-0.11468
110.00000	-0.12011
111.00000	-0.13697
112.00000	-0.14808
113.00000	-0.18176
114.00000	-0.18776
115.00000	-0.19904
116.00000	-0.20511
117.00000	-0.22784
118.00000	-0.25084
119.00000	-0.26862
120.00000	-0.26960
121.00000	-0.30463
122.00000	-0.30577
123.00000	-0.33591
124.00000	-0.33720
125.00000	-0.35061
126.00000	-0.37626
127.00000	-0.38823

128.00000	-0.40213
129.00000	-0.42330
130.00000	-0.44994
131.00000	-0.47173
132.00000	-0.47897
133.00000	-0.50129
134.00000	-0.54231
135.00000	-0.54480
136.00000	-0.57621
137.00000	-0.56790
138.00000	-0.61925
139.00000	-0.63884
140.00000	-0.65569
141.00000	-0.68145
142.00000	-0.73088
143.00000	-0.75263
144.00000	-0.76928
145.00000	-0.79030
146.00000	-0.80970
147.00000	-0.81065
148.00000	-0.88367
149.00000	-0.89782
150.00000	-0.92400
151.00000	-1.00067
152.00000	-1.03536
153.00000	-1.08947
154.00000	-1.09584
155.00000	-1.20465
156.00000	-1.22123
157.00000	-1.26090
158.00000	-1.30056
159.00000	-1.33557
160.00000	-1.49718
161.00000	-1.53483
162.00000	-1.70123
163.00000	-1.69167
164.00000	-1.89507
165.00000	-2.06525
166.00000	-2.24788
167.00000	-2.45874
168.00000	-2.80478
169.00000	-3.22446
170.00000	-3.86839
171.00000	-4.00000
172.00000	-4.00000

173.00000	0.00000
174.00000	0.00000
175.00000	0.00000
176.00000	0.00000
177.00000	0.00000
178.00000	0.00000
179.00000	0.00000
180.00000	0.00000
181.00000	0.00000
182.00000	0.00000
183.00000	0.00000
184.00000	0.00000
185.00000	0.00000
186.00000	0.00000
187.00000	0.00000
188.00000	0.00000
189.00000	0.00000
190.00000	0.00000
191.00000	0.00000
192.00000	0.00000
193.00000	0.00000
194.00000	0.00000
195.00000	0.00000
196.00000	0.00000
197.00000	0.00000
198.00000	0.00000
199.00000	0.00000
200.00000	0.00000

This data needs to be numerically integrated to arrive at height values.

Figure 4 (b) shows the plot of the surface height, as computed by integrating starting from left and from right. We used the trapezoidal integration method. Notice the small discrepancy, which is due to real world data noise. The overall shape does look circular, which should serve as a sanity check on the calculations.

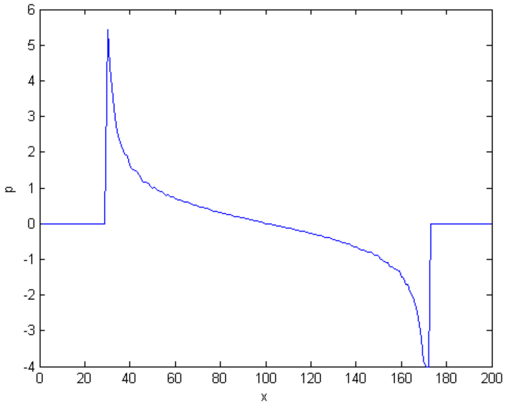

Figure 4(a)

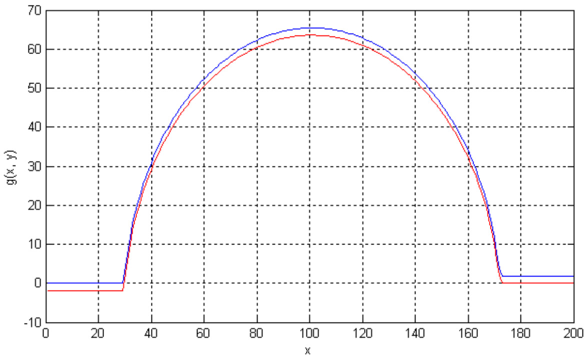

Figure 4(b)

Figure 4(a) The plot of the estimates for the partial derivatives along the x direction, $p = \dfrac{\partial g(x,y)}{\partial x}$ for the sphere along a horizontal line passing through the middle of the sphere.

Figure 4(b) Plot of the integrated value at each point along the horizontal. The blue plot corresponds to integrating from left to right and the red plot corresponds to integrating from right to left.

If we repeat the above process along the vertical direction (along image columns), then we will arrive at two more estimates for the middle point of the sphere, which can be averaged to arrive at one estimate. If we repeat this for each point on the sphere, not just the middle, then we will arrive at the surface representation for the full sphere. Figure 5 shows the averaged estimate of the sphere as estimated from the input vector field. Note the spherical nature of the final estimate.

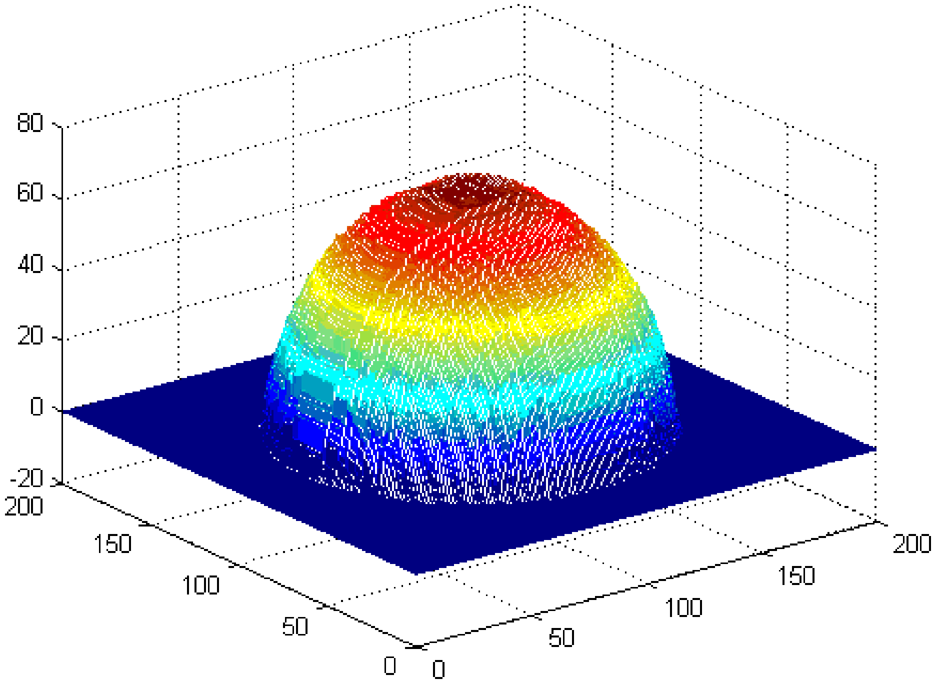

Figure 5 Estimated surface height for the vector field in Figure 1(c) as computed by averaging the estimates of the integrals along four directions, for each point.

QUESTIONS

1. Write code to recover the surface height at each point in the image given the vector field.
2. For any point on the sphere, specify 25 different paths along which you can integrate.
3. Study how the roughness of the estimated surface changes with number of path integrals that are averaged.

INTEGRATION	
Topic	Integration
Summary	To infer the surfaces from vector fields
Major	General Engineering
Authors	Sudeep Sarkar
Date	September 10, 2005
Web Site	http://numericalmethods.eng.usf.edu