Assignment 3 COMP 646 Instructor: Mike Langer

Posted Tues. April 1, 2008 Due: Thurs. April 24 (5 pm)

Old Instructions (same as A1, A2)

- Answers should be typeset using LaTeX or some other suitable software. (ASCII is fine, except when writing math formulas). Handwritten answers will not be accepted.
- All programming in this assignment is with Matlab. You will need to run OpenGL code, but you will not need to modify it or hand in new OpenGL code. Be sure to include a README file that explains the code you are handing in.
- You will be evaluated by the correctness of your answers, and by how clearly and concisely you present your results and arguments.
- There are a total of 20 points on this assignment.

New Instructions

- *If you make color figures, then you should print these on a color printer.* Cathy will mark the hard copy of what you hand in.
- *The due date is firm. Late assignments will not be accepted.*

Introduction

There are several purposes in this assignment:

- to give you some experience with steerable filters and how they are used for estimating oriented structures in images;
- to give you some experience in working with shaded images, and how shading depends on lighting, and how it interacts with other scene parameters such as overall surface slant;
- to understand the relationship between global properties of images such as overall image energy as a function of orientation, and local properties such as the dominant orientation at a point.

The tools we use in this assignment are not only applicable to shading analysis, but also to texture analysis, focus, motion, etc.

Code provided to you

Three programs are provided for you in this assignment:

- A Matlab program `mksteerable.m` that makes a steerable filter that can be used to estimate the local orientation and response amplitude of the image. You will use this filter to analyze a set of shaded images. The filter is defined using first and second derivatives of a Gaussian, similar to lecture 2.2 pages 4-5. The filter has arguments `N, SIGMA, THETA`.

- A Matlab program `A3base.m` that shows how to compute the response of the steerable filter by using the convolution theorem. You must use this method!

- An OpenGL program `render.c` renders a surface that is illuminated from a direction

  $$\texttt{lightposition}[\,] = (L_X, L_Y, L_Z).$$

  This light source direction is initialized to $(0, 0, 1)$. You can modify the light source direction using the left mouse button. Click left, and then drag the mouse across the image. Horizontal vs. vertical mouse motion changes the L_X vs. L_Y values, respectively.

  The surface is a "height field", defined by adding up a set of 2D sine and cosine functions over some range of frequencies from 0 to `KMAX`. See function `initHeightfieldFourier()` in `render.c`. The distribution of widths and the amplitude of the bumps is determined by the constants `KMAX` and `AMPLITUDE`, respectively. Do not modify these parameters for the experimental results that you hand in. The surface is initially given a slant of 0 degrees.

  You can modify the global surface slant and tilt by using the right mouse button. Click right, and drag the mouse. Horizontal vs. vertical mouse motion modifies the global surface normal vector. The surface is rotated so that it has this global normal vector.

Question 1 (10 points)

When you move the light source away from the line of sight, a bias is introduced in the distribution of image intensity gradients. See lecture 3.1 page 3. Here you will demonstrate this bias by filtering images with a steerable filter. You will consider two images rendered under different light source directions. One of these directions is the initial direction $(0, 0, 1)$. The other should differ from the original by about 45 degrees on the unit sphere. For each of these images, you will do the following:

1. For each pixel (x, y), compute θ_{max} which is the θ (`THETA`) that maximizes the response of the steerable filter:

   $$r(x, y, \theta) = \sqrt{(I(x, y) * G_\theta^2(x, y))^2 + (I(x, y) * G_\theta^1(x, y))^2}.$$

   Consider twelve θ values that are separated by 15 degrees from each other, which is a range of 180 degrees. We consider this range rather than the whole 360 since the responses will be identical up to a phase difference of 180 degrees (which don't care about here).

   For each image, show a histogram of the θ_{max} values computed over all pixels. (Hint: Use Matlab function such as `hist`, `rose`, `polar`.)

2. For each θ, give a plot of the average response $r(x, y, \theta)$, i.e. average over all pixels (x, y). Compare the result to (1).
3. Briefly discuss your findings in terms of the linear and quadratic components of the shading model presented in lecture 3.1.

NOTES:

- For this question, you should *not* alter the surface slant.
- Do not include points that are within a distance of say 4*`SIGMA` from the boundary of the image, since the responses at these points will be corrupted by the image boundaries. Specify which portion of the image you are using.
- Do not alter the parameters `SIGMA, AMP, KMAX`.
- Along with the histograms and plots, you should hand in the images you used, and specify the `LX, LY` values.

Question 2 (10 points)

When the surface has non-zero slant, perspective effects become evident, namely the surface is foreshortened in the tilt direction. This also can lead to a bias in the distribution of oriented structure in the image.

1. Define two images such that the surface slant is about 30 degrees, and the tilts are 0 degrees and 90 degrees. For each image, demonstrate a directional bias in the same way as in Question 1.2 namely give plots of the average response $r(x, y, \theta)$ over all pixels (x, y), as a function of θ. For both images, relate these average responses to the surface tilt direction.

   In this question, you should *not* vary the light source direction – leave it at $(0, 0, 1)$.

2. When you vary the slant and tilt of the surface *and* the light source direction, the orientation bias is determined by an interaction of these variables. Demonstrate this interaction by choosing one of the slant/tilt combinations from Q2.1, and varying the light source direction. Again, examine how average response varies with the filter orientation θ. In particular, consider cases in which the light source directions have:

   (a) the same tilt as the tilt of the plane,

   (b) a tilt that differs by 90 degrees from that of the plane.

   Note: the tilt of the light source is an angle τ such that (`LX, LY`) is parallel to $(\cos\tau, \sin\tau)$. Note it is undefined when (`LX, LY`) = (0,0).