Problem set 3

Assigned: 10/24/02
Due: 11/05/02 at 2:30pm

Problem 1 Lucas-Kanade tracker (Matlab)

Lucas & Kanade's optic flow technique proposes to solve the brightness constant constraint equation (BCCE) by assuming constant flow over a patch:

$$\begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = -\begin{pmatrix} \sum I_x I_t \\ \sum I_y I_t \end{pmatrix} \tag{1}$$

a) Implement a single-iteration Lucas-Kanade tracker. Your function should have the following syntax:

```
function [dx,dy] = lucaskanade(I1, I2, xl, yt, xr, yb, wsize)
```

where `I1` and `I2` are the images, (`xl,yt`) and (`xr`,`yb`) are the top-left and bottom-right corners of the region where you want the optical flow computed, and `wsize` is the size of the squared image patch used to solve equation (1).

Test your implementation using `LK-0001.bmp` and `LK-0002.bmp` with different window sizes: 5, 9, 13 and 17.

- For each window size, plot the optical flow for the region (`xl,yt`);(`xr`,`yb`)=(80,170);(150,220) using the Matlab function `quiver`.
- Is the estimated optical flow constant over the whole region? Should the true optical flow differ from your estimate? Why or why not?
- Does the window size change the estimated optical flow? Why or why not?

You should submit your file `lucaskanade_yourlastname.m` and print a copy of your code.

Hint: To compute the image gradient, you should first blur image I1 and then use the Matlab function gradient. *The temporal gradient can be approximated by the difference between blurred versions of I1 and I2.*

b) Derive the Lucas-Kanade solution (Eq. 1) from the BCCE by minimizing the following objective function:

$$E(u,v) = \sum_{x,y\in\Omega} \left(I_x(x,y)u + I_y(x,y)v + I_t\right)^2 \tag{2}$$

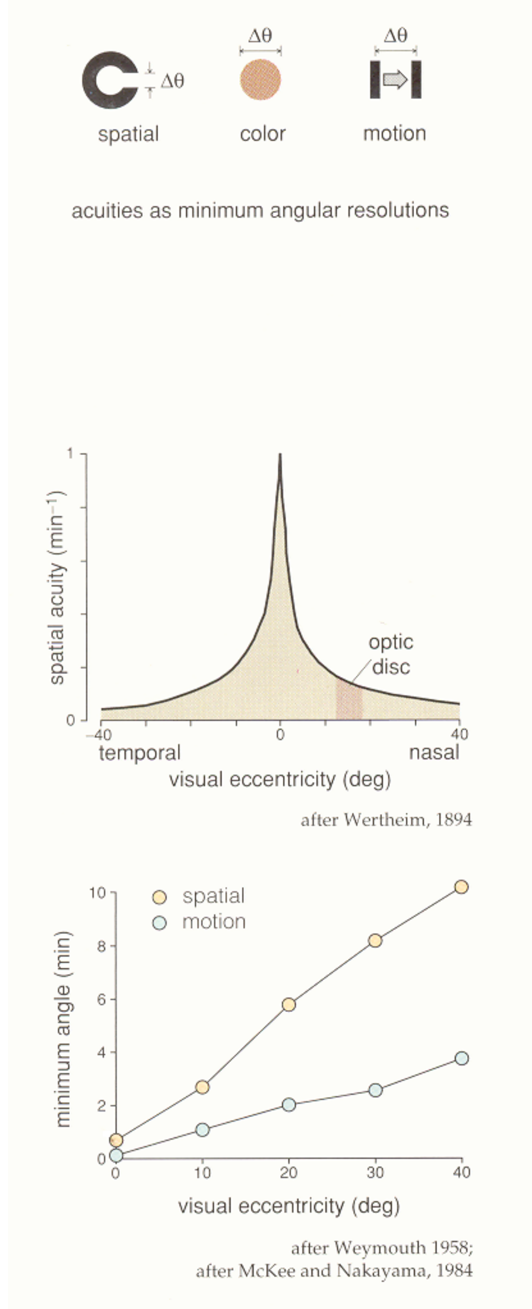
Hint: Rewrite the objective function as the least-squares solution of a linear system of equations in matrix form.

Problem 2 Spatially Variant Filtering (Matlab)

The next two problems use pyramid image processing. Download the matlabPyrTools from http://www.cns.nyu.edu/~eero/software.html. When forming pyramid decompositions for these problems, you may always use the default decomposition filters. For both problems, you should submit your Matlab code and include a printout.

Subjectively, our visual world appears to us to be high resolution everywhere. However, we have much higher spatial resolution in the center of our field of view than in the periphery. In this problem, we will synthesize an image approximating our visual resolution as a function of eccentricity.

The figure shows a plot of the minimum angle resolvable as a function of the visual eccentricity. The visual eccentricity is measured in degrees away from the center of fixation. (From Rodieck, "The First Steps in Seeing", Sinauer, 1998).



Approximate acuity, a, in minutes of arc (60 minutes to a degree) as a function of eccentricity, e, in degrees, by the expression,

$$a = 0.23\,e + 0.7$$

We will create an image with the effective spacing of the pixels equal to the angular size of the acuity limit. In the figure, that limit is defined as the white space between two ends of a circle. Adjacent black, white, black pixels could approximately represent that circle opening if the pixel spacing were equal to the angular size of the acuity limit.

Assume that the image (or monitor) is square, and that you view it from a distance of three times the length of one side of the image. Where convenient, you may assume angles are small enough so that $\tan(\theta) \approx \theta$.

a) How many evenly spaced pixels per side does the image need to have in order that the highest resolution part of the image has one pixel per length of finest acuity? Assume that some image point lies at zero eccentricity.
b) Let the upper left corner of the image be (0,0), and the right and bottom edges of the picture be at a distance 1 from this corner. Assume that the upper left corner is the center of fixation. What effective pixel spacing, as a function of these units, causes the pixel spacing to equal the spatial acuity for the corresponding eccentricity?
c) We can approximate images of this resolution by using a Gaussian pyramid, which generates images at different numbers of pixel samples, dividing the number of pixels by two at each level of the Gaussian pyramid. Start from an image at the full resolution of part (a). Each pyramid level increases the effective size of its pixels by a factor of two in each dimension. As a function of the coordinate system used in (b), by how many factors of two should the resolution of the original image be reduced as a function of position in the image in order to

simulate the human visual acuity, assuming the viewer stares at the upper left corner of the image?

d) The expression in (c) involves fractional pyramid levels. We can visually approximate images at those intermediate resolution levels by linearly interpolating between our Gaussian pyramid levels. On the class web site is a 2000x2000 image, which should be more than enough pixels for you. Crop that image to the desired resolution such that the upper left corner will be at the maximum visual acuity, when viewed from 3 picture lengths away. Use the Gaussian pyramid to create an image that simulates the fall-off in visual acuity, assume the fixation point is at the upper left corner. At any given pixel, determine the coefficients for interpolating between images by linearly interpolating the corresponding pixel dimensions.

   Hint: You will want to use the upBlur function to transform the Gaussian pyramid levels to all have the same number of pixels. Assume that a pyramid level after upBlur has effectively the same number of pixels (in terms of picture content) as the original pyramid band before the upBlur operation. That is a reasonable approximation (take 6.341 for the details that we're glossing over here).

Problem 3 Noise Removal (Matlab)

We study noise removal using a wavelet/QMF pyramid.

a) Download the three "training set" images from the web site. Convert each to a QMF pyramid. For the subband selecting vertical oriented structure, at the second highest resolution level, plot the histogram of the coefficient values, for each of the 3 training set images. Do these images share the subband histograms described in class?

b) Add independent Gaussian random noise to each pixel of each of the training images (mean zero, standard deviation 10). Reform the QMF pyramids, and plot the histograms for the same subbands used in (a). Discuss any changes.

c) For each image, make a scatter plot plotting noisy versus clean pixel values. Make similar scatter plots of noisy versus clean subband coefficients, for the single subband used in part (a). Discuss how these plots show that one representation is better than the other for noise removal.

The last two parts of this problem are ***optional*** *(and potentially time-consuming). They demonstrate how the statistical observations above could be transformed into an image denoising algorithm.*

d) Find a parametric form which estimates the clean subband coefficient value as a function of noisy input value. The values of the free parameters could in general depend on the original image statistics, the noise level, and the pyramid band. The original image statistics and the noise level may be estimated from the noisy image. However, for this exercise, we'll assume the fit is only a function of the pyramid band.

   Matlab's fminsearch function may be used to find the parameters which minimize the squared error from the observed relationships in the noisy and clean training images (such as the scatter plots of (c)). Combining the data for the 3 images, determine the parameters of your parametric form for each of the 3 subbands (vertical, horizontal, and diagonal) over the 4 finest pyramid levels.

e) Download the noisy test image, which is made from a clean image corrupted by noise at the same level as (b). Use the parametric forms from (d) to estimate a denoised image.

Problem 4 Focus of expansion

Exercise **17-2** from *Robot Vision* (Horn).

Problem 5 Center of rotation

Exercise **17-3** from *Robot Vision* (Horn).

Problem 6 Stereo and Motion estimation

a) Is the Stereo problem a special case of Motion estimation? Why or why not?
b) The Brightness Constant Constraint Equation is a classic technique for Motion estimation; is it possible to use it as a closed-form, least squares solution to the stereo estimation problem? Why or why not? If yes, which technique is better? If no, what extensions would be needed to make BCCE work for stereo?

Problem 7 Bayes' Theorem

Suppose we have a box containing 8 apples and 4 oranges, and we have a second box containing 10 apples and 2 oranges. One of the boxes is chosen at random (with equal probability) and an item is selected from the box and found to be an apple. Use Bayes' theorem to find the probability that the apple came from the first box.

Problem 8 Minimum-risk decision criterion

Verify that the minimum-risk decision criterion (Bishop, eq. 1.39) reduces to the decision rule (Bishop, eq. 1.26) for minimizing the probability of the misclassification when the loss matrix is given by $L_{kj} = 1 - \delta_{kj}$.