

COMP 558 Assignment 3

Prepared by Prof. Michael Langer and T.A.s Chatavut and Florian

Posted: Friday, Nov. 12, 2010

Due: Wed. Nov. 24, 2010 (11:59 PM) – no extensions

The purpose of this assignment is to give you a hands-on experience with scale spaces, namely a Gaussian scale space and a Laplacian of a Gaussian scale space. It will also give you some experience relating the structure of a scale spaces to the image information about a 3D scene that is available from focus and texture cues.

Question 1 (8 points)

You are given images of two scenes that consist of planar surfaces with a texture pattern (a shirt, a newspaper). For each scene, the images have different f-numbers and shutter speeds which have been chosen such that the exposures of focussed points are roughly the same within each pair. Let $I(x,y)$ and $J(x,y)$ be the images with the larger and smaller f-numbers, respectively, so $I(x,y)$ has a larger depth of field and $J(x,y)$ has a smaller depth of field.

- a) When shooting a pair of images as just described, a tripod is typically used. However, the camera may still shift slightly between shots. For this reason, the images must be re-aligned. Fill in the missing code from the function **globalImageRegistration.m** which returns a global shift (hx,hy) that best aligns two input images.

The rounded values of the shifts will be used in question (b). The correct rounded solution for the shirt example is $(hx,hy) = (0,0)$ and for the newspaper example $(hx,hy) = (1,0)$. These values are provided so that you can do question (a) without doing question (b). Your code image registration code will be tested on other pairs of images to ensure it is correct.

- b) Consider a Gaussian scale space, $I(x,y)*G(x,y,\sigma)$ that is based on the large f-number image. Suppose the optical blur in the small f-number image $J(x,y)$ can be approximated by digital blur with a 2D Gaussian. Then, for each (x,y) , we expect the mean square error,

$$\sum_{(x,y) \in \mathcal{N}_{gd}(x_0,y_0)} (J(x,y) - I(x,y) * G(x,y,\sigma))^2$$

to be smallest for a particular σ in scale space, namely a σ that corresponds roughly to the optical blur in the neighborhood of that pixel.

Fill in the missing code in **depthFromDefocus.m** so that it computes the **bestSigma** value for each pixel (x_0,y_0) , namely it computes the σ that minimizes the above expression. The code displays **bestSigma** in a figure, along with a colormap indicating the values. (See slides from lecture 17 for an example result).

Hints:

- You will need to choose a sufficiently large summation neighborhood to get your solution to work (see Ngd in expression above). The reasons include the presence of image noise, sub-pixel misregistration errors, and small natural lighting changes between the shots. A larger summation neighborhood reduces the sensitivity to these effects.
- The original images are too big for testing your code. Use the provided function **subSample.m** that reads an image, blurs it with a Gaussian $G(x,y,\sigma)$ and then subsamples it to produce a smaller image. Also, do not store the entire scale space at once since that would use too much memory.

c) Briefly discuss (max. 100 words) whether your results for the two cases, i.e. shirt and newspaper, could be used to infer depth from focus, and what the limitations are.

Question 2 (7 points)

- a) Take the large f-number image **shirt1.jpg**. Detect keypoints by computing the Laplacian of the Gaussian-blurred image at a range of scales, and finding local maxima and minima. Specifically, fill in the missing code from **boxDetect.m**.
- b) Explain the plot that is produced by the **scatter3** instruction. Use the Rotate 3D option in the Figure Toolbar to view the data in this plot. Briefly describe how one could use the keypoint distribution to compute shape from texture, and what the limitations might be. Specify any information e.g. threshold choice, image region choice, or tests that you applied. (Maximum 200 words please.)
- c) Run **boxDetect.m** on the **newspaper1.jpg** image. Again, does the keypoint distribution contain useful information about shape from texture ? (Maximum 200 words.)

General Instructions:

Submit **depthFromDefocus.m**, **globalImageRegistration.m**, **boxDetect.m**, along with a text file or PDF that contains answers to the questions above.

In order to receive full points, your solution code must be properly commented. It is not enough to produce correct answers