

CSE190B
Computer Vision

Machine Problem 3

Stereo Vision

Due: Tuesday, May 27, 2003

Part I: Background Reading

To get started read the Chapter 7 on Binocular Stereo in Trucco and Verri.

Part II: Written Problem

1. Do problem 7.8 from the textbook.
2. Consider two perspective cameras whose optical axes are parallel (say in direction $(0, 0, 1)$), and whose centers differ by a translation in orthogonal to the optical axis, (say $T = (T_x, T_y, 0)$).
 - (a) What is the Essential matrix?
 - (b) Show algebraically (not geometrically) that the epipolar lines are parallel.

Part III: Programming a Simple Binocular Stereo System

Your assignment is to program a binocular stereo system for recovering depth. Your data will be pair of stereo images – *left.bmp* and *right.bmp* – available by download from the course web page. The images have been rectified so that the scan lines are epipolar lines. Your output will be an image of disparity values - where dark denotes far and light denotes near. To get the disparity value at each point, you are to implement the following correlation (not normalized) stereo algorithm:

$$\hat{d} = \arg \max_d \sum_{k=-w}^w \sum_{l=-w}^w I_l(i+k, j+l) \times I_r(i+k, j+l-d)$$

where $I_l(i, j)$ is the image intensity in the left image at point (i, j) , $I_r(i, j)$ is the image intensity in the right image, and w is a parameter indicating the size of the window over which the optimization is performed. Record the \hat{d} values in a corresponding disparity image.

Do this process for $w = 0, 5$, and 10 . Restrict the possible disparity values to the integers and choose a reasonable value for d_{max} from an examination of the images.

Part IV: Improving the System

Modify the above algorithm in at least one significant way – as long as the modification improves the accuracy. Here are examples of three modifications that you might consider (1) Use normalized correlation (or sum of squared distances, SSD) rather than simple correlation. (2) Allow half integer (half pixel) values for the disparity. (3) Enforce the ordering constraint. There are many, many other ways to improve the method, but your choices shouldn't be just threshold tweaking. If your choices are too trivial, then you won't get full credit.

Part V: The Report

Write up a page or two description of how you implemented the algorithm in Part II and include any insights you may have had along the way. Describe your two modifications to the algorithm in Part III, discussing any possible trade-offs that come to mind. Attach **clearly labelled pictures** of the disparity maps for each value of the window size parameter w , for the original algorithm and your variant from Part III. Also, for each algorithm, showing mapping of 10 arbitrary points (see figure 7.5 in the text for example). Conclude the report with suggestions for additional ways to improve the algorithm, pointing out places in the disparity maps which might be corrected by your suggestions.

Hand in

1. Turn in the written problems and the report in class. NO EXCEPTION. Read part V above carefully to know what you need to include in the report.
2. Send the code to TA any time during the day (code only, no picture).