**HW#5 (Stereo matching)**

* Implement correlation-based matching of rectified stereo images.  The resulting disparity map should be the same size as the two input images, although the values at the left edge will be erroneous.  Match from left to right (i.e., for each window in the left image, search in the right image), so that the disparity map is with respect to the left image.  Recall that a (left) disparity map D(x,y) between a left image L and a right image R that have been rectified is an array such that the pixel corresponding to L(x,y) is R(x-D(x,y), y).
  + Implement the left-to-right consistency check, retaining a value in the left disparity map only if the corresponding point in the right disparity map yields the negative of that disparity.  The resulting disparity map should be valid only at the pixels that pass the consistency check; set other pixels to zero.
  + Your code should be efficient as possible, on the order of several frames per second.  (Hint:  First compute the dissimilarities of all the pixels for each disparity, storing the results in an array of images; then convolve each image with a summing kernel (all ones) in both directions.  Further speedup can be obtained using mmx\_diff and xmm\_diff in Blepo, but this is not required.)
  + Suggestion:  use SAD (sum of absolute differences) to match raw intensities and use a window size of 5x5.
  + Run your code on [tsukuba\_left.pgm](https://cecas.clemson.edu/~stb/ece847/fall2012/tsukuba_left.pgm) and [tsukuba\_right.pgm](https://cecas.clemson.edu/~stb/ece847/fall2012/tsukuba_right.pgm). Show the results both with and without the consistency check.  What kind of errors do you notice?  Now run the algorithm on[lamp\_left.pgm](https://cecas.clemson.edu/~stb/ece847/fall2012/lamp_left.pgm) and [lamp\_right.pgm](https://cecas.clemson.edu/~stb/ece847/fall2012/lamp_right.pgm). What happens? Why is this image difficult?
  + \*Your code should output a PLY file that can be read by [MeshLab](http://meshlab.sourceforge.net/).  This will enable you to visualize the matching results in 3D.  Here is an [example PLY file](https://cecas.clemson.edu/~stb/ece847/fall2012/kermit.ply) created from a set of [Kermit](https://cecas.clemson.edu/~stb/ece847/fall2012/kermit.jpg)images.  PLY files are ASCII files with a simple format:  In the header you specify the number of vertices, along with the properties stored for each vertex (e.g., x y z nx ny nz r g b); then after the header there is one line per vertex.  For your assignment, you should just output six columns (x y z r g b) for each matched pixel, ignoring the normal components.  Manually set the focal length and baseline to nominal values in order to achieve visually plausible results. You can use either perspective or orthographic projection to get your x,y,z coordinates.  Orthographic is simpler and will lead to a more aesthetically pleasing point cloud, but it is less accurate mathematically. Your stereo matching code does not have to work in color space, but you should definitely use RGB color to make your PLY file more pleasant to look at:  [tsukuba\_left.ppm](https://cecas.clemson.edu/~stb/ece847/fall2012/tsukuba_left.ppm) and [tsukuba\_right.ppm](https://cecas.clemson.edu/~stb/ece847/fall2012/tsukuba_right.ppm) .  (*Note: When you click the "Import Mesh" button (the one that looks like File.Open), MeshLab merges the file that you select with the current mesh -- it****does not replace****the current mesh with the new one. To fix this problem, either be sure to click the "Reload" button (next to the "Import Mesh" button) or close MeshLab altogether and start it up again.*)
* In separate figure windows, display the two input images, the disparity map resulting from left-to-right matching, and disparity map after applying the left-right disparity check. Optionally, you may display other intermediate results if you want.
* Take a look at the results of the latest stereo research at <http://vision.middlebury.edu/stereo> (click on the "Evaluation" tab). Look only at the column (nonocc) under the column Tsukuba. What errors do you see in the best algorithm (the one with minimum error in this column)?  What does this tell you about the difficulty of the problem?
* No report is due for this assignment.