CS 63500
Spring 2018
Voicu Popescu
Due: Friday March 9th at 11:59pm

Assignment 4—Depth from stereo and rendering from a depth image

In a nutshell

Implement a basic application that takes as input a pair of overlapping images (digital photos) acquired with two cameras in a known standard stereo configuration, enhances the left image with per pixel depth, and then renders the scene interactively from the depth image.

Details

1. Take a pair of overlapping images with cameras that are in the standard stereo configuration, i.e. their views is the same except for a known horizontal direction.
    a. OK to use same camera to take left image and then to translate the camera to take the right image.
    b. The scene should contain
        i. View-dependent effects, e.g. some surfaces should look differently in the two depth images.
        ii. Occlusion-disocclusion effects, i.e. some surfaces visible in the left image should not be visible in the right image, and vice-versa.
    c. The scene can be static, i.e. geometry and lighting doesn't change between the time the left and the right images are taken.
2. Find approximate depth per pixel in the left image by searching for correspondences for the left image pixels in the right image, along their respective epipolar segments.
    a. Have a maximum error threshold above which a correspondence is considered invalid and the pixel remains without depth.
3. Allow the user to render the scene interactively using the depth image
    a. Use point-based rendering, where each pixel with depth from the left image is reprojected to the output image, and drawn as a square point of an application chosen size, with z-buffering.
4. Make a white (close) to black (far) visualization of your depth image, with red for pixels that do not have valid depth.
    a. Save the visualization for three different values of the correspondence error threshold discussed above at 2.a.
5. Make a 10s video that shows the scene rendered from the depth image.
    a. The output camera should have the same intrinsics as the left image.
    b. Start from the view of the left image, and then translate right, and pan left, to show the scene from a different viewpoint that illustrates the depth you have computed from stereo.
6. Extra credit
    a. Render the scene from a triangle mesh obtained by 2D Delaunay triangulation of the valid depth samples in the left image (3%)

b. A post-processing (i.e. post depth from stereo) mechanism for filling in the holes in the depth map (2%)
c. Anything else that creates a compelling visual experience (negotiable%)

Turn in via blackboard

An archive that contains:

- Your source code and binaries
- Your output video
- Your depth visualizations (i.e. 3 images, one for each good depth threshold)
