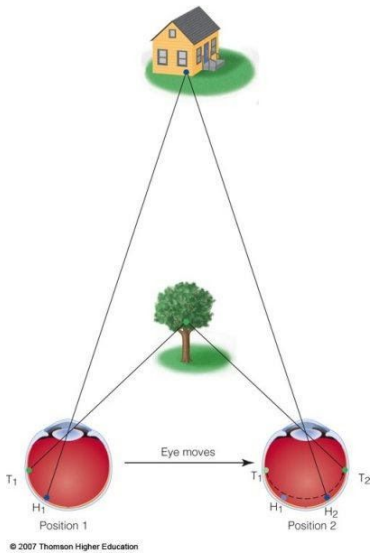
Stereo Depth Estimation

EE6130 – Computational Imaging & Displays

Assignment I (Deadline March 1st 2021)

Total Marks - 10



When looking out of the side window of a moving car, the distant scenery seems to move slowly while the lamp posts flash by at a high speed. This effect is called parallax, and it can be exploited to extract geometrical information from a scene. From multiple captures of the same scene from different viewpoints, it is possible to estimate the distance of the objects, i.e. the *depth* of the scene. By tracking the displacement of points between the alternate images, the distance of those points from the camera can be determined. Different disparities between the points in the two images of a stereo pair are the result of *parallax*. When a point in the scene is projected onto the image planes of two horizontally displaced cameras, the closer the point is to the camera baseline, the more difference is observed in its relative location on the image planes. Stereo matching aims to identify the corresponding points and retrieve their displacement to reconstruct the geometry of the scene as a depth map.

More recently, stereo matching has been adapted to produce and transmit content for 3D TV, especially for multiview displays where it can save significant bandwidth compared to sending all required views separately. Depth manipulation via view synthesis is the fundamental concepts in 3D displays and VR HMDs. It also allows scaling of 3D content for different sizes and types of displays. However, stereo matching and depth estimation is an ill-defined problem and have its own drawbacks. Understanding of which is a part of this assignment.

Middlebury Stereo View Database

http://vision.middlebury.edu/stereo/data/

The Middlebury stereo database has a collection of stereo pairs for use in development of stereo matching algorithms. Some of the Middlebury images are the de facto standard in comparing the results when new algorithms are proposed. They also assess rank matching algorithms on the quality of the produced depth map.

Step I: Select stereo images from 2001, 2003, 2005, 2006 datasets. Download **Art, Books, Dolls, Reindeer, Cones, Teddy data** for your experiments. You may pick pairs from other years. To limit processing time, use the link labeled ***"2 views: ThirdSize"*** to get subsampled images. **Note values in ground truth disparity maps of this size should be divided by 3 to correspond to actual disparity**. Unzip the files to your working directory and familiarize yourself with the contents of the resulting folders. Data structure is like that

File	view1.png	view5.png	disp1.png	disp5.png
Content	Left image	Right image	Left disparity	Right disparity

Task I: Pick any of the high ranking matching algorithms in the top 20 or so and check the publication behind it. What kind of similarity metric (i.e. how does the algorithm determine, which points correspond) the algorithm uses? Cite the publication and summarize the paper idea in your own words.

http://vision.middlebury.edu/stereo/eval3/

Disparity map quality

The quality of estimated disparity is commonly measured in bad pixels – the percentage of pixels in the estimated disparity that are different from the ground truth. Small errors can be allowed by requiring the estimate to be within a given threshold.

Task II: Implement a function that counts those pixels in the estimated disparity that are different from the ground truth and returns a percentage of the different pixels in the whole image. Name that function as **'dataname_badpixelcount.m'**, e.g. '**Art_badpixelcount.m**'.

Disparity estimation

The whole concept of stereo matching is based on finding correspondences between input images. The correspondence between two points is determined by inspecting the pixel neighborhood N around both points. The pairing that has the lowest *sum of absolute differences*, is selected as a corresponding point pair. In practice, a matching block is located for each pixel in an image. The relative difference in the location of the points on the image planes is the *disparity* of that point. Due to the assumption of being constrained into a 1-dimensional search space, these disparities can be represented as a 2D disparity map which is the same size as the image. Disparity of a point is closely related to the depth of the point. This is essentially a block matching scheme. In this application, the search space can be constrained to be horizontal only (with certain assumptions). The matching block size is one of the most important parameters that affect the outcome of the estimation. Smaller blocks can match finer detail, but are more prone to errors, while large blocks are more robust, but destroy detail. In this assignment, a symmetric, square block of radius r has pixels.

Task III: Implement a Matlab function that estimates disparity from the Middlebury stereo images with different block radii using stereo matching algorithm SAD (sum of absolute difference). Measure the quality of the reconstructed map with the bad pixels -function you implemented at different radii. Make a single plot that contains a graph for each stereo pair depicting the quality metric as a function of block radius.

Analysis: Which block size is the best for each image? Is there a correlation between the type of content and the best block size you observed?. Attach the best estimate of each image to the report. Take a look at the output parameter `cost` from the depth estimation function. What kind of problem areas could you identify based on that information?. On the other hand, what problem areas cannot be determined from it?.

Real data sets

The Middlebury data sets have high quality images captured in a controlled setting and with carefully calibrated equipment. In practice, the task becomes more difficult. Camera alignment even on physically connected cameras is never perfect, and there may be issues with different properties of the cameras. We will provide stereo images captured using ZED camera. Check our IC3D paper:

https://ieeexplore.ieee.org/document/8975903

https://drive.google.com/file/d/1FxMfV8OaSvcZHZI3hiILhxCL5kAPjyYW/view?usp=sharing

Rectification

For efficient stereo matching, the concept of *epipolar lines* is essential. The horizontal translation in the camera positions between the stereo pair should only create differences in the horizontal direction. This allows for the matching to happen only in one direction, along an epipolar line, greatly reducing the search space. However, this is rarely the case in a realistic capture setting. Camera misalignment makes it impossible to reliably search only on the epipolar lines. This is why a software rectification step is performed after capture. In short, the misalignment of the image planes can be corrected by rotating them in a 3-dimensional space. The rotation matrices doing this can be computed from the images themselves.

Task IV: Implement stereo rectification and attach the best depth map you are able to generate to the report.

Analysis: How does the subjective quality of the depth maps compare to the ones estimated from Middlebury data?. Are there any distinctive problems with some type of content?. If (and when) any especially problematic images or sections appear, also include an example of that.

Summary of tasks

1. Download images From Middlebury.
2. Make a function for evaluating the quality (% of bad pixels).
3. Estimate depth from the Middlebury images using different radii.
4. Collect stereo images captured using ZED Camera.
5. Apply rectification to the captured ZED camera images.
6. Estimate depth from the rectified ZED camera images.
7. Make a report including the plots, images and answers specified in the detailed instructions.
8. Submit Matlab codes you implement.