INF6804 Computer Vision
H2020 – Practical Assignment 2
Description and comparison of regions of interest

Objectives:

- Allow the student to learn about image description based on histograms of oriented gradients (HOG).
- Allow the student to learn about image description based on BRIEF.
- Learn the basics of disparity estimation in stereoscopy.

Submission:

- *All your source code* (we should be able to run your tests)
- A report (*.pdf format* of 8 to 15 pages with font size of 10)
- Submit before March 13th, 5:00 PM, on Moodle –*late submissions will not be accepted*
- *You must also submit your report on TurnItIn*
  - Register at www.turnitin.com using the info available on Moodle!

References:

- See course notes on Moodle (Chapter 2 and chapter 4)

Other directives:

- The assignments must be made in teams of two, submit only one version of your work!

Presentation

In this assignment, you will have to characterize two methods used for the description of regions of interest in images, and determine which method is better, and under which circumstances. A description of your work, your experiments, and answers to the questions outlined in this document must be included in your report.

In this assignment, you will have to compare two region description methods, namely a method based on description by HOG, and a method based on BRIEF. These methods have been briefly presented in class — you can use your course notes as a reference to understand their basic working principles. For more details, go look online! Your goal here is to determine which approach works best when trying to match image regions from two different images under varying conditions (e.g. using different region sizes, under illumination variation, under affine transformations, etc.). To compare both methods, you will have to solve the task of disparity estimation. You will have to describe each image with the descriptors, and then use a stereo matching algorithm, such as semi-global matching (SGM), to get the disparity estimation.

In your report, you have to include the following elements (marked on 20 pts):

1. Presentation of the two methods (4 pts) :

   In your own words, give the general description and principles of your two methods.

2. Performance hypotheses in specific use cases (3 pts) :

   Identify, based on your theoretical understanding of the two methods, which one should be the best of the two in at least THREE specific use cases. For example, which is the best method to use if the size of the compared regions is very large (e.g. 200 pixels by 200 pixels)? Why? And if their content is relatively uniform ?

3. Description of experiments, datasets and evaluation criteria (2 pts) :

   Describe in detail the experiments realized to test the hypotheses of the previous point. Which dataset did you use? What are the difficulties in this datasets? Which evaluation criteria did you use?

4. Description of the implementations used (2 pts) :

   Describe the implementation of the two studied methods as well as the algorithm chosen to perform the disparity estimation. If you did not write all the code yourself, where does it come from? Did it require modifications? Otherwise, from which papers or websites did you inspire yourself to write it? In all cases, what are the primary parameters of your methods? How did you set their values?

5. Experimentation results (3 pts) :

   Provide the evaluation results from your experiments related to the hypotheses of the first point. Use a proper format for their presentation — tables, figures, ...

6. Discussion on results and prior hypotheses (3 pts) :

   Discuss the results of the fourth point in relation with the hypotheses of the first point. Which hypotheses are supported by these results? Which are not? Which test resulted in a lack of conclusion? How could you improve these tests?

7. Readability and completeness (3 pts) :

   In addition to the content, the format must be clean and complete.

We propose three datasets (that you can find in the resources section) to do your experiments. You can also use any other stereo dataset of your choice.

During the lab periods, do not hesitate to ask questions to the TAs — they can help you with any technical issue if you are working on Windows/Linux, or if you are coding in C/C++, Python or Matlab.

You will be penalized by 50% of the total grade if you do not hand in your code. Also, if your report is not submitted to TurnItIn, it will not be graded. The order of presentation for the topics listed above does not matter, as long as they are all present.

Disparity estimation

We reference a lecture on the basics of stereo matching in the resources section. Simply put, the task of disparity estimation is, for given pixel p at location (x, y) in a reference image, to find the exact same pixel q at location $(x + d, y)$ in another image. Therefore, d, the disparity, is the distance between the same pixel of both images. In order to predict d, you will have to first describe both left and right images with the descriptors. Then, you will form a cost volume, a tensor of dimension $H \times W \times D$ (H = height of image, W = width of image and D = number of disparity), where the cost of an element corresponds to a distance (L1, L2 or any other distance) between the elements of the descriptors. Figure 1 shows a small example of how to form the cost volume where both descriptors are 4×4 pixels, and the maximum disparity is 4. You can view the process as sliding one descriptor over the other, and at each possible offset, computing the distance between each elements of the descriptors. You repeat this process for every possible disparity value. Finally, in figure 2, we show an example of the left and right stereo images with the corresponding left disparity image. The lighter the color, the bigger the disparity value.

After computing the cost volume, you need to apply a matching algorithm

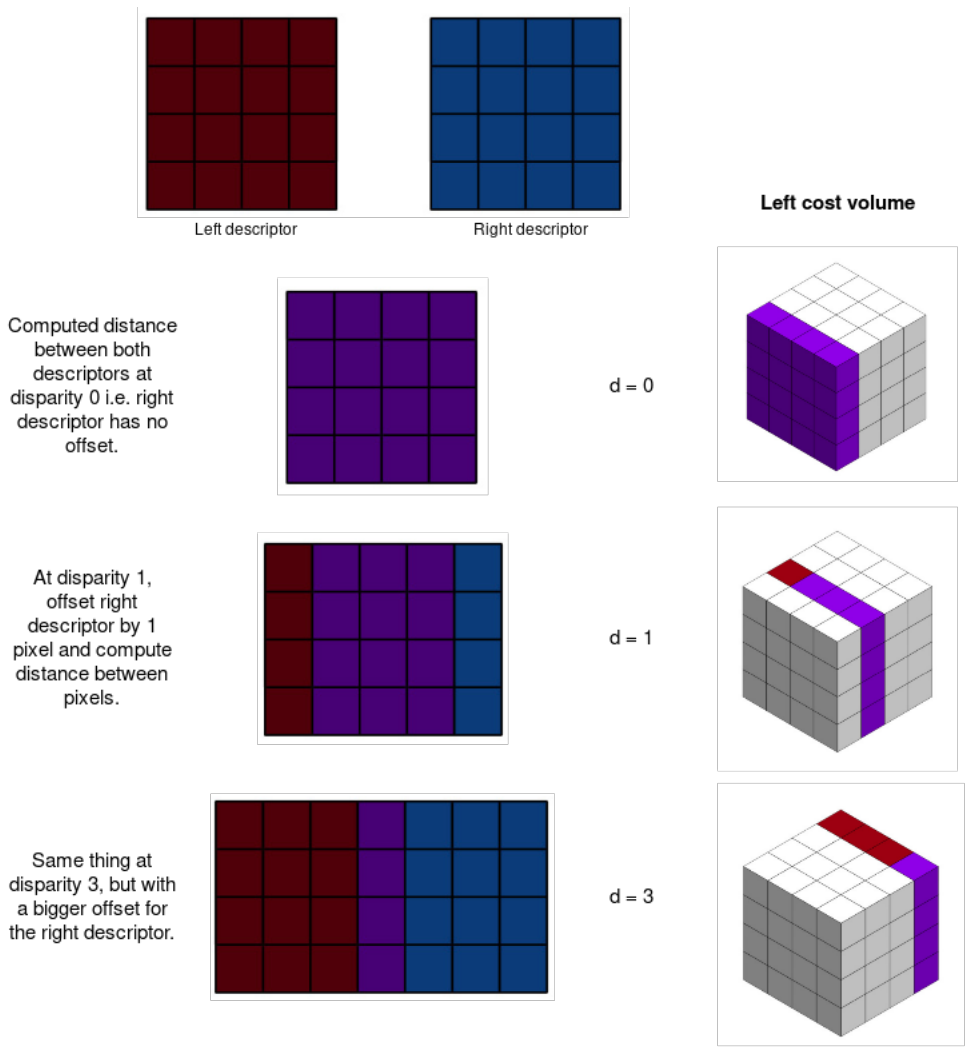

Figure 1: Diagram showing how to compute the cost volume from left and right descriptors. Red: left descriptor, Blue: right descriptor and Purple: computed distance between left and right descriptor.

to refine the volume. For this part, you can use any stereo algorithm of your choice. However, we suggest that you look at SGM since it is one of the best classical method and there are a lot of implementations that can be found online (Github or OpenCV). We put the reference to the paper in the resources section. Each matching algorithm will have its own way of producing the final disparity image.

Figure 2: Stereo image pair with the computed disparity map.

Resources

Datasets:

- KITTI dataset (http://www.cvlibs.net/datasets/kitti/eval_stereo.php)
- Middlebury dataset (http://vision.middlebury.edu/stereo/data/)
- CATS dataset (http://bigdatavision.org/cats/data.html)

Stereo/Semi-global matching:

- Lecture on stereo from University of Toronto (http://www.cs.toronto.edu/~fidler/slides/2015/CSC420/lecture12_hres.pdf)
- SGM paper (https://core.ac.uk/download/pdf/11134866.pdf)

Vision libraries:

- OpenCV (https://docs.opencv.org/4.0.0/d9/df8/tutorial_root.html)
- scikit-image (https://scikit-image.org/docs/stable/auto_examples/index.html)

Deep learning frameworks:

- PyTorch (https://pytorch.org/tutorials/)
- Tensorflow (https://www.tensorflow.org/tutorials)

Python:

- Guide (https://wiki.python.org/moin/BeginnersGuide/Programmers)
- NumPy (https://docs.scipy.org/doc/numpy/user/quickstart.html)

Matlab:

- Guide (http://www.mathworks.com/help/pdf_doc/matlab/getstart.pdf)
- Cheatsheet (http://web.mit.edu/18.06/www/Spring09/matlab-cheatsheet.pdf)