TIVA TP1: Canny edge detector and Bilateral filter

Mathieu Aubry mathieu.aubry@imagine.enpc.fr

October 7, 2014

This TP must be finished and returned on `https://educnet.enpc.fr` before October 13. Return a single .zip or .tar file containing your code and a report in pdf including the answers to the questions, your results and their analysis.

Note: no answer in the report are necessary for the first part, but results and discussion are expected for all the points of parts 2 to 4.

1 Beginning with Matlab

1. Subscribe to the course and download the initial program and data on `https://educnet.enpc.fr`.
2. Create a directory 'Bureau/TIVA_TP1'.
3. Extract the initial program in 'Bureau/TIVA_TP1'.
4. Open Matlab and change the work directory to 'Bureau/TIVA_TP1'.
5. Open *intro.m*, read, run and understand the code inside it. Ask questions if something is not clear.
6. Open the script *tp1.m.* Read, run and understand the first section of the code. Ask questions if something is not clear.

2 Basic image processing

1. The function *gaussian_convolution* computes the convolution of an image with a Gaussian kernel. Similar to what we have done in the first lecture, complete the function *gaussian_filter_2d* to make it work properly. Visualize its result. What are the important points to check?
2. Visualize the image blurred with Gaussian kernels of several variances. Comment.
3. Why are the borders of the image darker? Can you suggest strategies to solve this problem? (you do not have to implement them)
4. Complete the function *compute_gradient* to compute the gradient of an image by finite difference, its norm and its orientation. Visualize its norm.
5. Adapt the *tp1* script to compute the thresholded gradient with several thresholds. Comment on the difference between the results obtain with the different thresholds.

3 Canny edge detector

1. Complete the function *quantify_gradient.* Visualize its output.
2. Complete the function *non_max_suppression* l. 21-26. Visualize its output and comment. (note: the contours obtained at this point should already be meaningful, if they are not satisfying, you need to figure out where there is a problem in your code before trying to finish this section)
3. Comment on the effect of noise in the input.

4. Comment on the effect of the size of the blur kernel.

5. Understand and complete the function *canny_edges* to compute hysteresis thresholding (l. 21).

6. Visualize the Canny edges. Try several images and parameters. Comment the results.

4 Bilateral Grid

The cell Bilateral Filter of the script *tp1.m* compute the bilateral filter of an image using the bilateral grid algorithm implemented in *BF.m*.

1. Justify the default choice of the *range_sampling* and *spatial_sampling* parameters. Experiment with those parameter and discuss the results.

2. Experiment different choices of *sigma_spatial* and *sigma_range* and comment the results, in particular extreme cases.

3. Use the results of the bilateral filter to perform details enhancement. Compute the smoothed image with a Gaussian blur instead of the Bilateral Filter and compare.

5 Optional

Try to apply the Canny edge detector after applying a bilateral filter on the images. Comment the results.