

HOMework ASSIGNMENT 2

Edge Detection and Geometrical Modification

Due Date: 11:59 p.m. on Apr. 2, 2024

Please read the **submission guideline** carefully before getting started. All images in this homework are in PNG format and can be downloaded from our NTU COOL website. Details of all files offered are listed in the appendix. You are **NOT** allowed to use other functions except I/O, plotting and basic functions.

Problem 1: EDGE DETECTION

Given two different images named **sample1.png** and **sample2.png**, please follow the instructions below to perform different kinds of edge detection algorithm.



(a) sample1.png



(b) sample2.png

Figure 1: Images for edge detection.

- (10 pt) Apply Sobel edge detection to **sample1.png**. Output the gradient image as **result1.png** and its corresponding edge map as **result2.png**. Additionally, describe the threshold selection process and its impact on the result, and provide your analysis.
- (20 pt) Perform Canny edge detection on **sample1.png** and save the resulting edge map as **result3.png**. Also, provide an explanation of your parameter selection process and how it impacts the outcome.
- (10 pt) Using Laplacian of Gaussian edge detection to generate the edge map of **sample1.png** and save it as **result4.png**. Compare **result2.png**, **result3.png**, and **result4.png**, and discuss the differences among these three results.
- (10 pt) Perform edge crispening on **sample2.png** and save the result as **result5.png**. Describe the differences between **sample2.png** and **result5.png**. Please also specify the parameters you used and explain how they influenced the outcome.
- (Bonus) Perform Canny edge detection on **result5.png** and save the edge map as **result6.png**. Then, apply the Hough transform to **result6.png** and save the resultant image as **result7.png**. What lines can you detect using this method?

Problem 2: GEOMETRICAL MODIFICATION

Please design several geometrical modification algorithms to meet the following requirements. **Your results may not be exactly the same as the sample images. Just try to create the effects as closely as possible.**

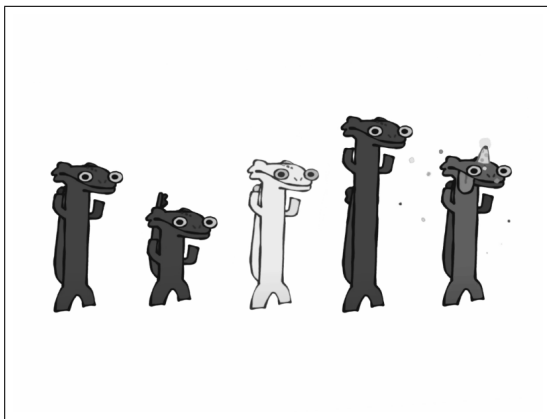
- (a) (25 pt) Toothless wants to become stronger. Please design an algorithm to convert **sample3.png** into **sample4.png**. The output results should be saved in **result8.png**, and the output image size is required to be the same as **sample3.png**. Please describe your approach and implementation details clearly. (hint: you may perform rotation, scaling, translation, etc.)
- (b) (25 pt) Toothless and his friends are practicing their new dance moves. Please design an algorithm to convert **sample5.png** into **sample6.png**. The output results should be saved as **result9.png**, and the output image size is required to be the same as **sample5.png**. Please describe the details of your method and provide some discussion on the design approach, results, and differences between **result9.png** and **sample6.png**.



(a) sample3.png



(b) sample4.png



(c) sample5.png



(d) sample6.png

Figure 2: Images for geometrical modification.

Appendix

Problem 1: EDGE DETECTION

sample1.png:	600×600	gray-scale
sample2.png:	500×500	gray-scale

Problem 2: GEOMETRICAL MODIFICATION

sample3.png:	600×600	gray-scale
sample5.png:	600×800	gray-scale