Final assignment A50: image comparison.

Module: Introduction to Programming for Architecture and Design, BENVGACH
Student number: SN12013268
Date: 02/01/2018
Word count: 939/1000

1 Introduction

The final assignment for the module Introduction to Programming for Architecture and Design consists of comparing several images in order to detect the most similar to a target image of London. This has been achieved through a piece of code written in the "Processing" language and included within this final submission.

Three methods have been implemented for this assignment: Euclidean Distance; Chamfer Distance and Brightness Average. All applied for both large and reduced images. The reduced images were obtained by averaging the pixels of the larger image.

All methods showed different results and is the opinion of the author that the Euclidean Distance (HSB variant) method provided the most similar images to the target.

2 Objectives

To compare a given set of images against a target image of London and select the most similar one to the target using the Euclidean Distance method, the Chamfer Distance method and the Brightness Average method for both large and reduced images.

All the pre-processing, processing and visualisation must be written in "Processing".

To produce a comparison table of speed between all methods implemented.

3 Methods

For this assignment four methods were used: Euclidean Distance with two variants; Chamfer Distance and Brightness Average. Before comparison, the code loads all images, check that they have the same sizes, and normalizes the hue or saturation or brightness (HSB) of every pixel.

The HSB model was chosen over the red, green, blue (RGB) model as it the former has advantages over the later regarding image comparison (Chmelar and Benkrid 2014) and (Schwarz, Cowan, and Beatty 1987).

Normalization is required to avoid that images of similar places with different illumination or colour are classified as different. The following is the formula used for normalization of each colour attribute.

$$x_{new} = x - x_{min} / x_{max} - x_{min}$$

3.1 Dimension reduction

All the methods can be applied to the original size images or for a set of copies with reduced dimension with the goal of comparing speed and results of both methods.

The reduced images were computed by overlapping a matrix over the original image and getting the average brightness of each cell. Only the brigthness was used to ease the reduction because hue and saturation can vary greatly for the same object in different conditions.

3.2 Euclidean distance

This method is very popular and simple but it is sensible to small deformations (Wang, Zhang, and Feng 2005). The euclidean distance is given by the formula below were x and y are the pixel values of the images being compared and h, s, b the colour attributes of the pixels.

$$d_{Euclidean}(x,y)=\sqrt{\sum_{K=1}^{N\ pixels}\sum_{c=h,s,b}(x^k-y^k)^2}$$

This method compares the color attribute of every pixel in one image against the attribute of the corresponding pixel in the target image and aggregates the individual pixel comparisons for the whole image (pixel to pixel comparison or vector approach).

This methods has two variants, the first compares the hue, saturation and brightness (HSB) and the second one only compares the brightness.

The second variant was created to test whether if, by using less channels, the result would vary.

3.3 Chamfer distance

This method requires first to detect the edges within every image. The edges were computed by comparing the brightness value of each pixel with its immediate neighbours. As the comparison is based in a mathemathical substraction, the values can vary greatly. To simplify, the edge values were normalized again from 0 to 1 and then transformed into binary using a threshold of 0.15.

For every edge pixel, the distance to the closest edge pixel in the target image was calculated. Both *x* and *y* axis were used for this search. The search was limited to a radius of 10 pixels for reduced images and 100 pixels for large images.

At the end, the average distance for every edge pixel is computed. This is preferred to a total sum as the amount of edges pixels vary for every image (Thayananthan et al. 2003).

4 Code

All the methods are implemented by a code written in Processing and included with this submission. The user is able to choose the method to use, compare reduced or original images, see the winning image, the results for every image and compare the speed between the methods.

The code first checks that all images are of the same size as it performs a pixel to pixel comparison.

5 Results

For all the methods, the computing time for the reduced image is significantly low (see Figure 1). For the larger sizes, the chamfer distance method takes longer to compute, followed by the all other methods. However, these results are shown in miliseconds and therefore any methods used for both original and reduced images take little time to produce a result.

	Euclidean distance		Euclidean (b only)		Chamfer distance		Brightness average	
	Reduced	Original	Reduced	Original	Reduced	Original	Reduced	Original
Time (ms)	0	4	0	3	0	8	0	4
First 10 most similar	34	32	34	36	44	8	32	32
	10	5	10	9	4	50	48	41
	11	37	11	6	13	4	24	48
	32	21	32	11	24	16	41	24
	9	13	9	18	43	0	34	34
	49	17	49	14	30	23	10	10
	5	39	5	23	21	6	15	15
	21	7	21	47	16	31	9	9
	36	4	36	45	42	14	12	12
	12	49	12	5	27	36	8	8

Figure 1: Table of results

Image 32 shows up as a winner in at least two of the methods used: Euclidean distance method (HSB) and Brightness Average method. There is no clear second winner when comparing all the methods used.

A visual comparison suggests that the Euclidean distance method gives more score to “urbanized” areas similar to the target image than any other method which suggest that it is a better choice for comparison. The Brighness average seems to give high score to many rural areas not similar to the target image perhaps because it doesn´t consider the pixel to pixel differences. The Chamfer method seems to produce the least “accurate” results but it might be due to the simple implementation in this assignment.

The results for same methods but different size of image tend to be different except for the Brightness average method. This can suggest that is better to use large images for comparison as the processing speed is similar.

6 References

Chmelar, P., and A. Benkrid. “Efficiency of HSV over RGB Gaussian Mixture Model for Fire Detection.” In 2014 24th International Conference Radioelektronika, 1–4, 2014. https://doi.org/10.1109/Radioelek.2014.6828426.

Schwarz, Michael W., William B. Cowan, and John C. Beatty. “An Experimental Comparison of RGB, YIQ, LAB, HSV, and Opponent Color Models.” ACM Trans. Graph. 6, no. 2 (April 1987): 123–158. https://doi.org/10.1145/31336.31338.

Wang, Liwei, Yan Zhang, and Jufu Feng. “On the Euclidean Distance of Images.” IEEE Transactions on Pattern Analysis and Machine Intelligence 27, no. 8 (August 2005): 1334–39. https://doi.org/10.1109/TPAMI.2005.165.

Thayananthan, A., B. Stenger, P. H. S. Torr, and R. Cipolla. 2003. “Shape Context and Chamfer Matching in Cluttered Scenes.” In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, 1:I-127-I-133 vol.1. https://doi.org/10.1109/CVPR.2003.1211346.