Recognition of Panorama Parts Using OpenCV

Ekaterina Polishchuk
Saint Petersburg Academic University –
Nanotechnology Research and Education Centre RAS
St. Petersburg, Russia
katepol14@gmail.com

Abstract

The more data you store, the harder it becomes to manage it. This project is dedicated to classification of photos.

The data is 6000 image series sourced from panoramic images. Each series contains matching fragments of a panoramic image with a partial overlap (possibly in the wrong order). Some series may contain one or two images from other series. The goal is to figure out the odd ones.

Index Terms: OpenCV, panorama recognition, template matching, SURF.

I. Introduction

The task was taken from Yandex contest Internet Mathematics 2011 (http://imat2011.yandex.ru/).

Image data sets are sourced from panoramic images. The data set consists of 6000 series, 5 images each. The images are in the JPEG format with resolution 300 * 300 pixels. Each series contains matching fragments of a panoramic image with a partial overlap (possibly in the wrong order). Some series may contain one or two images from other series.

The task is to find these odd fragments using fully automated methods.

II. Main part

To solve the problem were used two methods: template matching and image stitching. For image stitching was used SURF algorithm. The source code was written on C++.

Data processing scheme:

1) *Evaluation of M := matrix of similarity coefficients for each series.*

A. M evaluation using template matching

For each pair of images: the first image is called source and the second one is called target. Then the segments of target are searched in the source using OpenCV::matchTemplate method. Target to segments split is implemented recursively.

m_{ij} = total matching percentage for all target segments.

Method OpenCV::matchTemplate sometimes gives false positive responses.

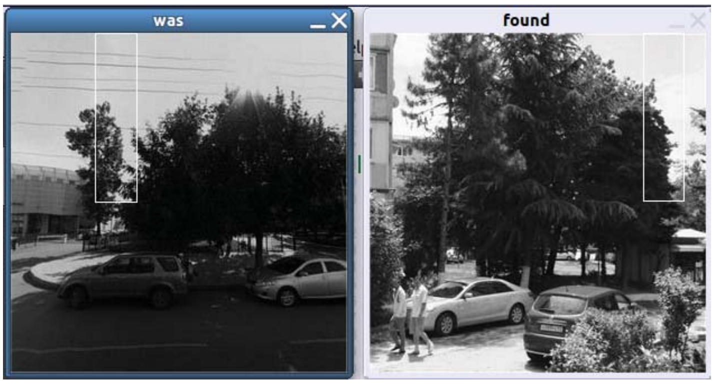

Fig. 1. False positive result of OpenCV::matchTemplate method. Fragments similarity K = 80.36%.

To cut off mismatches of this kind, was added verification of similarity of segments to the right and to the left. If both right and left segments exist (there is no overrun of both source and target images), the evaluated match is taken into consideration only if one of similarity coefficients (left or right) is greater than the pass value.

If one of right and left segments, or both, do not exist, the evaluated match is taken into consideration anyway.

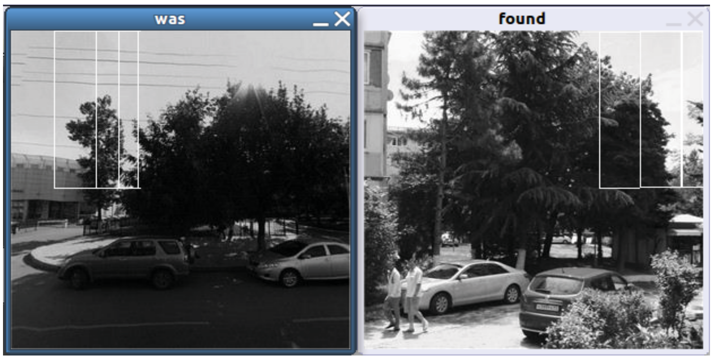

Fig. 2. Similarity verification. K lefts = 50.31%, K rights = 49.86%.

This behavior is dictated by the following reason: if found match is positioned close to the bounds of overlap, it should not be cut off in spite of left or right mismatch.

B. M evaluation using SURF

For each pair of images: find matching key points for current pair using OpenCV::cvExtractSURF method.

m_{ij} = number of matching key points for images i,j.

To defy whether two key points are matching or not, was developed naive nearest neighbor algorithm. This algorithm operates with key points' features which are extracted by OpenCV::cvExtractSURF.

2) *Search for groups of images that belong to the same panorama by evaluated matrix.* The classification is performed the following way:

- For each image make a vector of images that probably belong to the same panorama. The decision is made by similarity coefficient K value;

- The image which has maximal K is considered the initiator of first panorama;
- The rest images are added to this panorama or initiate a new one.

According to task conditions, there has to be one panorama containing at least 3 images. The images that are not in the largest group are odds.

To improve the performance

- the application was made multithreaded;
- Despite the images are colour, they are uploaded as greyscale. The comparison tests for template matching (OpenCV::cvExtractSURF takes only greyscale) have shown no changes in classification quality and the processing time has decreased by 60%.

III. Conclusion

Using template matching approach, the presented algorithm classifies the images with 88.1% correctness. And image stitching approach gives 92.6% correctness. The application has been tested on two data arrays of 6000 image sets and has shown a stable result: the marks for the same approach vary by 0.1.

Shortcomings

Each method has its own special shortcomings.

A. Shortcomings of template matching approach

1) Asymmetry of evaluating matrix of similarity coefficients.

This defect is caused by the big number of false positive OpenCV::matchTemplate responses. The examined pictures contain quite resembling areas.

2) High sensitivity to noise objects.

The introduced method is unable to ignore noise objects completely. It is unreasonable to divide the target image into smaller segments: the smaller segments are taken, the greater probability of false positive response becomes.

3) High sensitivity to perspective transformation.

The images which contain the same object rotated more than by 15-20 degrees are considered as mismatches.

B. Shortcomings of image stitching approach

1) Different objects on images – different types of key points.

The parameter which determines, which feature is a key point, is common for every image. For some images it may be unsuitable, so less than 10 key points per image are extracted. It does not allow classifying adequately.

2) High sensitivity to perspective transformation.

The overlap may be small and contain several objects which key points may radically change with object rotation.

3) High sensitivity to illumination change.

Possible ways to improve the results

1) develop a metrics which would evaluate and compare a kind of color fingerprint for some areas (this areas should be determined experimentally);
2) use a combination of methods;
3) optimize multi-threading for better performance.

Source code

https://github.com/katepol/PanoramaRecognition

ACKNOWLEDGMENT

The author would like to thank Ekaterina Tuzova and Kirill Krinkin. Without them this paper would have never been completed.

Also the author would like to thank the Finnish-Russian University Cooperation in Telecommunications which supported this activity (see: http://fruct.org/node/31388).

REFERENCES

[1] Nikhil Gupta, Rahul Gupta, Amardeep Singh, Matt Wytock, "Object Recognition using Template Matching," *URL:* www.stanford.edu/~nikgupta/reports/cs229-report.pdf

[2] Luo Juan, Oubong Gwun, "A Comparison of SIFT, PCA-SIFT and SURF," URL: www.cscjournals.org/csc/manuscript/Journals/IJIP/volume3/.../IJIP-51.pdf

[3] "OpenCV Reference Documentation," URL: http://opencv.willowgarage.com/documentation/