

## FEATURE BASED PANORAMIC IMAGE CREATION

### I. INTRODUCTION

In Computer Vision lectures, we have learned how to manipulate and infer meaningful data from images. We expected implement a term project based on the methods that are taught in this class. In *Recognizing Panoramas* [1], Brown and Lowe explains how to match images and create a panoramic image based on features. Several of the methods that are taught in the class are utilized for this implementation. In this report implementation of simpler version of this paper will be examined.

### II. BACKGROUND

We learned all concepts that are necessary for this project such as SIFT and projective transformation.

In this project, I decided to use OpenCV 3.2 on Python 3 since I wanted to use a new framework. OpenCV provides elegant functions for developers that facilitates their works.

### III. PROJECT SPECIFICATION

Similar to Brown's approach, the application first matches SIFT features between two given images. Then using those SIFT feature pairs, a homography matrix is calculated via RANSAC. This homography matrix is then used to warp one of the images onto other one.

### IV. PROBLEM ANALYSIS

Problem is partitioned into sub problems listed below:

- Finding interest points
- Feature description from interest points
- Feature matching
- Homography matrix computation
- Transforming images based on homography matrix
- Stitching images

All these sub problems can be solved with OpenCV methods.

## V. SOLUTION DESIGN

We will well-known methods that are mentioned in both lectures and the Lowe's article [1] to solve the problems established in the previous section. Solutions are listed below:

- Scale Invariant Feature Transform (SIFT) [2] for finding interest points and feature descriptions
- Distance function and ratio test for feature matching
- Random Sample Consensus (RANSAC) [3] for homography matrix calculation.
- Projective transform for images
- Simple stitching

## VI. IMPLEMENTATION

In this section, we will examine OpenCV functions that is used for implementation of solutions proposed in the previous section.

*xfeatures2d.SIFT\_create()*: This method creates an object that can be used for both interest point detection and feature description. *detect()* and *compute()* methods are used for interest point detection and feature description, respectively.

*DescriptorMatcher create()*: This method creates a descriptor object. Its *knnMatch()* method allows us to find k best matches in given feature set. Set of matches is returned from the method.

For Lowe's ratio test, we simple go through every match in the set and do elimination.

*findHomography()*: This is used for homography matrix based on selected interest points. It allows us to use different methods for computation, we chose to use RANSAC.

*warpPerspective()*: This is used for generation of projective transformations of the images corresponding to homography matrices.

For stitching, we do simple matrix manipulations.

There are other OpenCV methods that are utilized for image reading, showing, writing and resizing. But how they work is not in the scope of this paper.

### Requirements and Limitations of Implementation

- All images have same focal length.
- Input images are given in the correct order.

## VII. EVALUATION

This implementation can be refined with a bundle adjustment algorithm which would allow it to have input images in random order. Moreover, we can add a feature to find optimum width and height of the panoramic image for the sake of beauty. This implementation can be a useful tool for creation of panoramic pictures with addition of those features.

## VIII. CONCLUSION

This is a good project to utilize and demonstrate many of the concepts that are taught in Computer Vision course. Thanks to OpenCV, many complex tasks are done easily.

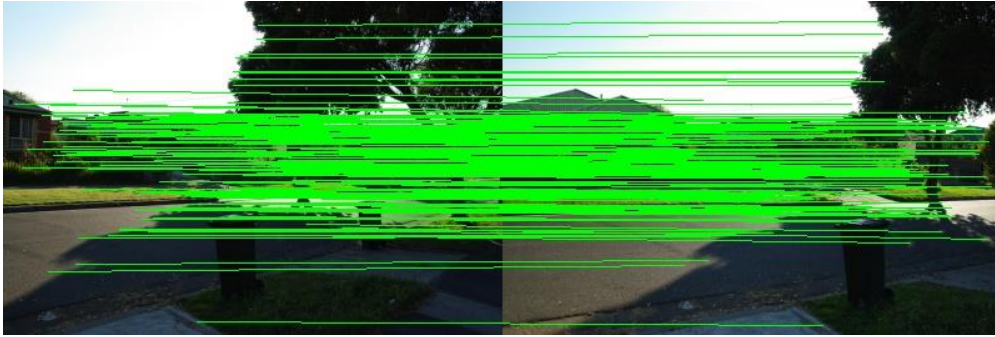
## IMAGES

### INPUT



### FOUND INTEREST POINTS



**MATCHED FEATURES****RESULT****IX. REFERENCES**

- [1] D. G. L. .. V. 3. 2. Matthew Brown, "Recognising Panoramas," *ICCV*, vol. 3, 2003.
- [2] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, p. 91–110, 2004.
- [3] R. C. B. Martin A. Fischler, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381-395, 2981.