**Course Project Report for Undergraduate Students**

**Course Type：** Subject Elective

**Course Name：** Computer Vision

计算机视觉(全英)

**Course Code：**

**《Creating a panoramic image from multiple photographs》**

**Name**

**Student ID**

**College**

**Major**

**Email**

Submit Date：/2022

Contents

[1. Abstract 3](#_Toc104921137)

[2. Methods 3](#_Toc104921138)

[2.1. Automatic Matching Feature Points 3](#_Toc104921139)

[2.2. DLT (Direct Linear Transformation) Algorithm 3](#_Toc104921140)

[2.2.1. Normalize Coordinates 3](#_Toc104921141)

[2.2.2. Estimate Projection Matrix 4](#_Toc104921142)

[2.2.3. SVD (Singular Value Decomposition) 5](#_Toc104921143)

[2.3. Symmetric Transfer Error 6](#_Toc104921144)

[2.4. RANSAC (Random Sample Consensus) Robust Estimation 6](#_Toc104921145)

[2.5. Gold Standard Algorithm 6](#_Toc104921146)

[3. Procedure 7](#_Toc104921147)

[3.1. Automatic Method 7](#_Toc104921148)

[3.1.1. Detect & Match Feature Points 7](#_Toc104921149)

[3.1.2. Compute Projection Matrix 8](#_Toc104921150)

[3.1.3. Results 11](#_Toc104921151)

[3.2. Manual Method 12](#_Toc104921152)

[3.2.1. Corresponding Matching Points 12](#_Toc104921153)

[3.2.2. Compute Projection Matrix 12](#_Toc104921154)

[3.2.3. Results 15](#_Toc104921155)

[3.3. Analysis 16](#_Toc104921156)

# Abstract

This project aims to combine two or more images into a panoramic view. And it mainly composed of three parts. The first part is to find the corresponding points in the pair of images. To accomplish the task, manual point selection method and automatic matching points method are used. The second part is the calculation of projection matrix, the normalized DLT (Direct Linear Transformation) algorithm is first used with the matching point pairs for the initial estimation. Specially, for automatic matching points method, since there are amounts of matching points, RANSAC (Random Sample Consensus) algorithm is used to find a precise projection matrix. Then the Gold Standard algorithm is used to minimize the geometric error for find a more precise projection matrix. The third part is to transform and combine the pictures into a panorama based on the projection matrices.

# Methods

## Automatic Matching Feature Points

To accomplish this task, the feature points should be found first. First, the image should be converted into a gray scale image, then the *detectSURFFeatures()* function is used to find out the points that containing information about SURF (Speeded Up Robust Features) features. And *extractFeatures()* function is used to obtain the feature vectors.

Second, for the two images, the feature points should be matched. In this case, *matchFeatures()* function is used to find out the index pair of the feature points whose features are matched. Specially, in this project, since there are too many feature points and there exists several points with similar feature vectors. The argument *‘MatchThreshold’* is set to *0.1* for selecting the strongest matches. The argument *‘Unique’* is set as *‘true’* to guarantee the only unique matches between the feature sets of two images.

## DLT (Direct Linear Transformation) Algorithm

After obtaining the matching points of the two images, normalized DLT algorithm is used to calculate the projection matrix between the images.

Objective: Given 2D to 2D point correspondences , determine the 2D projection matrix such that

Algorithm:

1. For each correspondence compute . Usually only two first rows needed.
2. Assemble n matrices into a single matrix .
3. Obtain of . Solution for is last column of .
4. Determine from .

### Normalize Coordinates

Suppose there are feature points selected for each image, the formulas to normalized each point in the corresponding image are shown as follows.

After normalization, for each the centroid is at the original point. And the average distance to the origin is . The normalization operation can be as the matrix format as follows.

Where,

, ;

,

### Estimate Projection Matrix

Since

Then

Which can be transformed into

In matrix format,

i.e.

Then, assemble n matrices into a single matrix . Obtain of . Solution for is last column of . After reshaping the vector into the normalized projection matrix .

To obtain the projection matrix , since

, ;

Then,

The projection matrix .

### SVD (Singular Value Decomposition)

Given a square matrix , the SVD is a factorization of as, where and are orthogonal matrices, and is a diagonal matrix that with all 0s except the main diagonal.

Since for a real symmetric matrix, its eigenvalues are real, and its eigenvectors are perpendicular to each other, they can be transformed into orthogonal matrices by orthogonalization. Suppose is a real symmetric matrix, is the orthogonal matrix after is orthogonalized. And is a diagonal matrix. Then,

Since is a matrix, and  are symmetric matrix, then

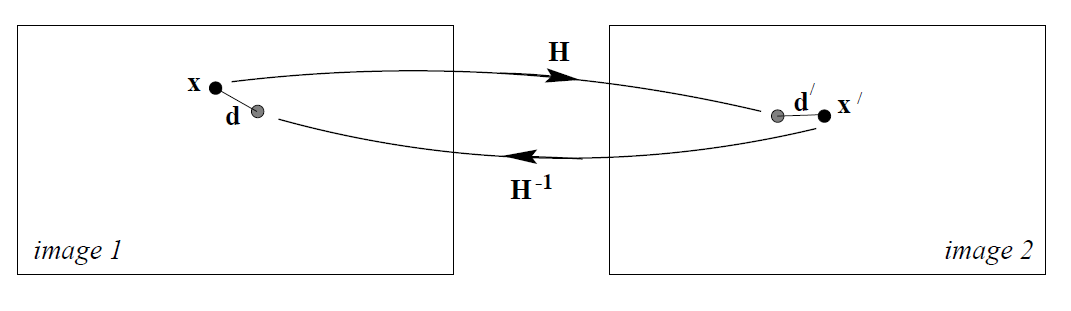
Where the column vectors of is the standard orthogonal eigenvectors of , and the column vectors of is the standard orthogonal eigenvectors of .

When and , can be transformed into , Since  is a matrix, the last column of is , and the last column of the product of is also , which indicates that the last column of multiply with matrix A, will equal to . That means the last column of is the solution of .

## Symmetric Transfer Error

To evaluate the geometric error of the normalized projection matrix , symmetric transfer error is used as the geometric error, where the symmetric transfer error is

Which is shown in the following figure.



**Figure 1.** symmetric transfer error

## RANSAC (Random Sample Consensus) Robust Estimation

Repeat for samples

1. Select a random sample of 8 correspondences and compute the projection matrix by DLT algorithm.
2. Instead of compute the distance for each putative correspondence, in this project, since the symmetric transfer error is used, the symmetric transfer error for the computed is used to replace the distance and the number of inliers.

Choose the H with the smallest symmetric transfer error.

## Gold Standard Algorithm

Geometric minimization of symmetric transfer error:

* 1. Compute the initial estimation of projection matrix with normalized DLT algorithm and RANSAC algorithm if the correspondences are chosen automatically.
  2. Minimized the symmetric transfer error over . The error is minimized using Levenberg-Marquardt algorithm for finding a more precise projection matrix.

In this project, to minimize the symmetric transfer error, the function *lsqnonlin()* with the option Levenberg-Marquart algorithm is used to find a more precise projection matrix.

# Procedure

In this project, three examples are used. One is about the distant buildings, one is about the buildings up close, and one is about indoor close-up. Based on the above various scenes, to make a comparison of the performances of the program.

1. **Buildings up close**

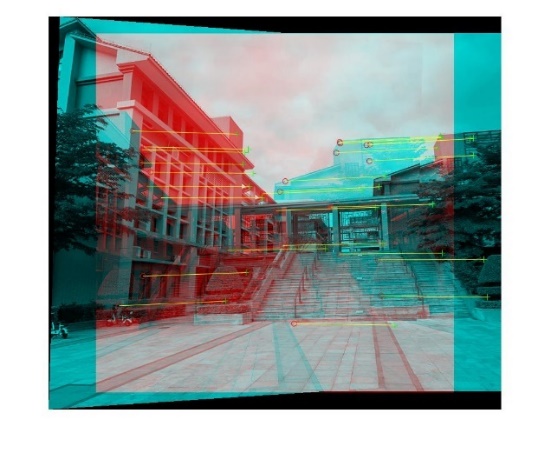
1. **Distant Buildings**

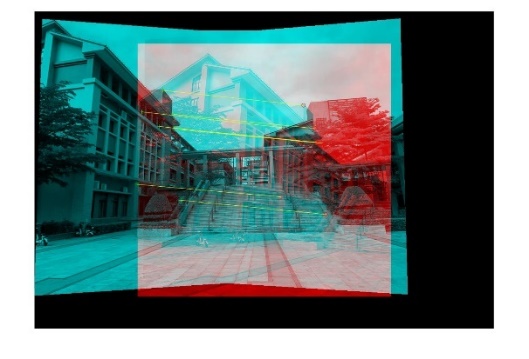
  

## Automatic Method

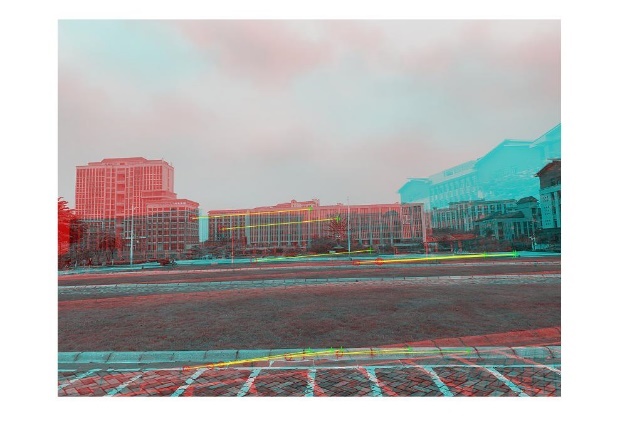
### Detect & Match Feature Points

1. **Buildings up close**





1. **Distant buildings**



### Compute Projection Matrix

In automatic method, since there are amounts of matching points, RANSAC algorithm is used to compute the projection matrix. Each time it takes 8 correspondences to compute the projection matrix by DLT algorithm, after 500 iterations, the projection matrix that with the best performance (that is with the smallest symmetric transfer error) is chosen and minimized the geometric error by the gold standard algorithm, after that the estimated projection matrix is used for transformations.

Since sampling correspondences randomly, the results are slightly different at each time. For the gold standard algorithm, the iteration stops when the function tolerance is smaller than 1e-10. The following figures are the error descending trend when using the Gold Standard algorithm for each calculation of the projection matrices.

1. **Buildings up close**
2. **Distant Buildings**

As the above figures show, the symmetric transfer errors have descended slightly after minimizing the geometric distance. And the following tables illustrate the error and the projection matrix for the two images.

1. **Buildings up close**

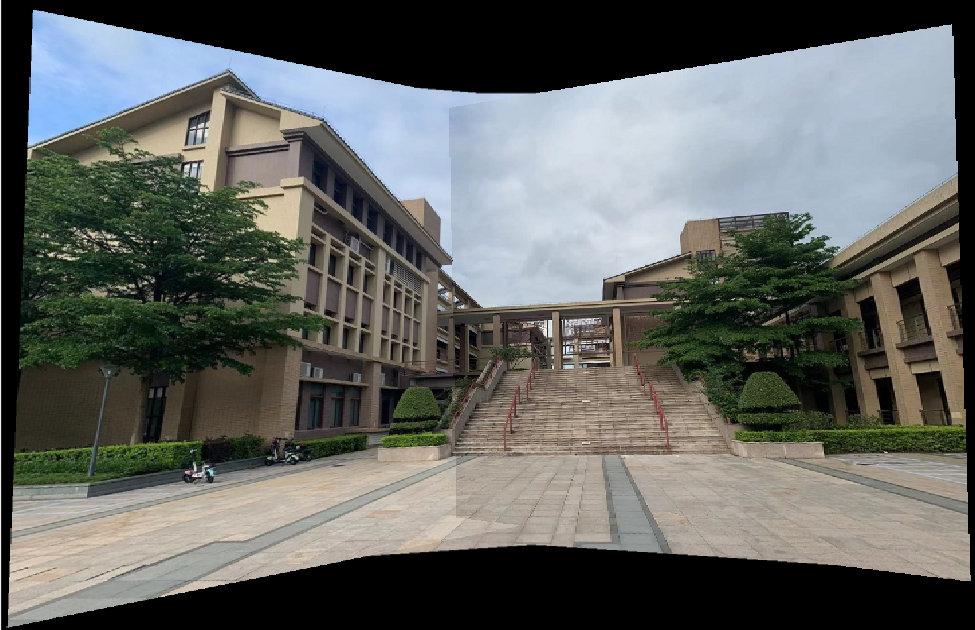
|  |  |  |
| --- | --- | --- |
| **Image Index (1 to n)** | **Error** |  |
| 3 with 2 | 0.0085 |  |
| 4 with (3 and 2) |  |  |
| 1 with (4, 3 and 2) | 0.0130 |  |
| 5 with (1, 4, 3 and 2) |  |  |

1. **Distant Buildings**

|  |  |  |
| --- | --- | --- |
| **Image Index (1 to n)** | **Error** |  |
| 2 with 1 |  |  |
| 3 with (2 and 1) |  |  |

### Results

1. **Buildings up close**



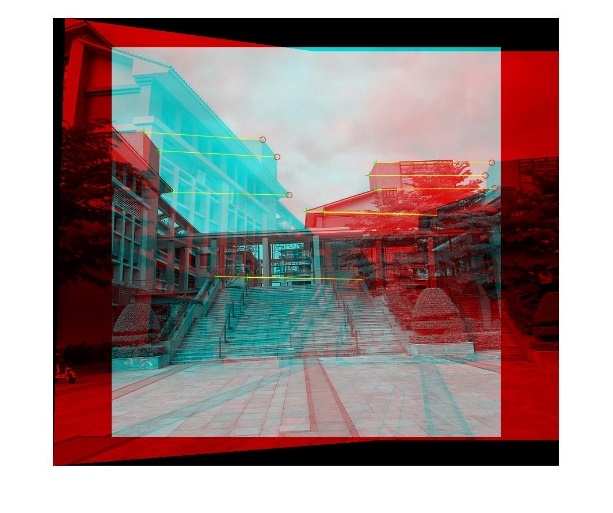
1. **Distant buildings**

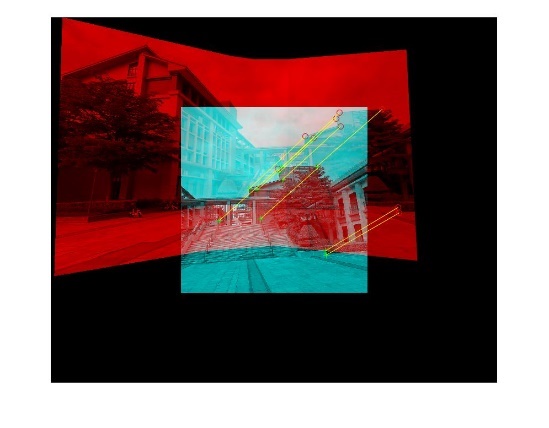
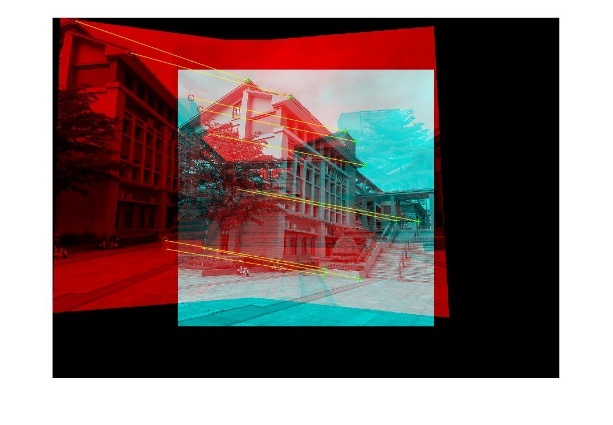


## Manual Method

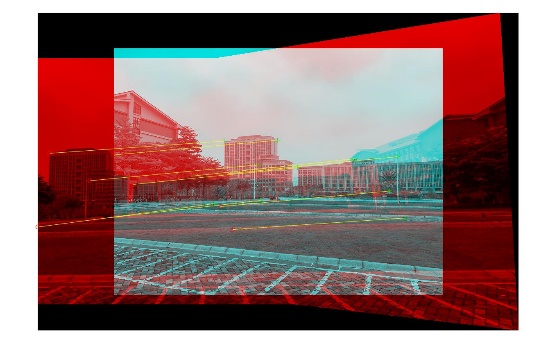
### Corresponding Matching Points

1. **Buildings up close**





1. **Distant buildings**



### Compute Projection Matrix

In mutual method, each time it takes 10 correspondences to estimate the projection matrix by DLT algorithm, then minimized the geometric error by the gold standard algorithm, after that the estimated projection matrix is used for transformations.

Since sampling correspondences randomly, the results are slightly different at each time. For the gold standard algorithm, the iteration stops when the function tolerance is smaller than 1e-10. The following figures are the error descending trend when using the Gold Standard algorithm for each calculation of the projection matrices.

1. **Buildings up close**
2. **Distant Buildings**

As the above figures show, the symmetric transfer errors have descended rapider than in the automatic method but still slightly after minimizing the geometric distance. And the following tables illustrate the error and the projection matrix for the two images.

1. **Buildings up close**

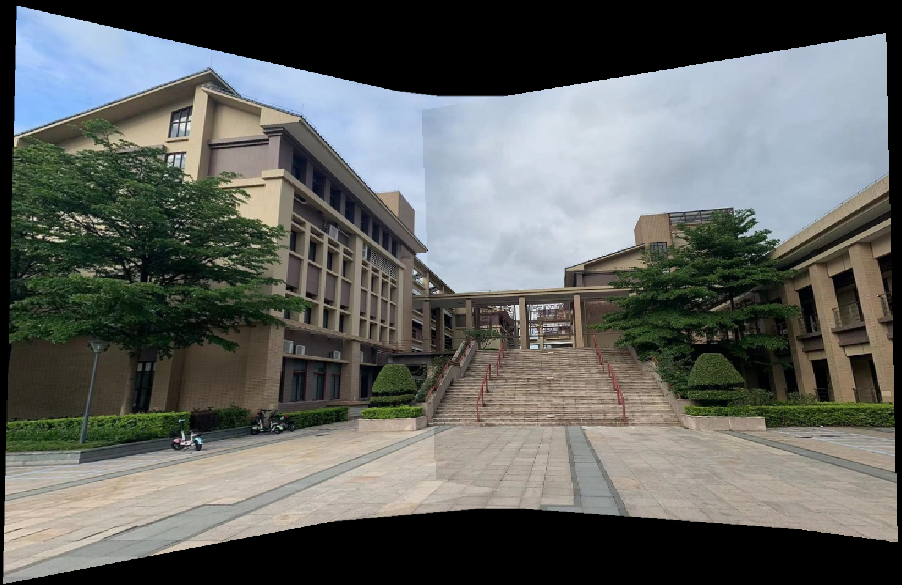
|  |  |  |
| --- | --- | --- |
| **Image Index (1 to n)** | **Error** |  |
| 3 with 2 |  |  |
| 4 with (3 and 2) |  |  |
| 1 with (4, 3 and 2) |  |  |
| 5 with (1, 4, 3 and 2) |  |  |

1. **Distant Buildings**

|  |  |  |
| --- | --- | --- |
| **Image Index (1 to n)** | **Error** |  |
| 2 with 1 | 0.0028 |  |
| 3 with (2 and 1) | 0.0011 |  |

### Results

1. Buildings up close



1. Distant buildings



## Analysis

As the above results show, it is noticeable that there is an apparent dividing line on every panorama. The reason is that the different levels of exposure intensity caused by the changing camera angle and the chromatic aberration caused by the adaptive setting of the camera.

From the above panoramas and symmetric transfer errors, it seems that for automatic method, if the angle between two images is larger, the error of the projection

matrix between them will be larger. It is possible that these photos are token by my bare hand without any professional photography equipment. Hence, if the angle between two images is larger, the movement and the deviation of the camera are larger simultaneously.

The comparison of the above two panoramas indicates that it is not so great for distant building since the matching points of the image apt to concentrated on a small specific area, which may cause a large error. Meanwhile, distant building photo is not friendly for manual method, since the features appear much smaller, which may cause a large deviation during picking the correspondences.