

# IN4393-16 Computer Vision Final Project Report

## Structure From Motion

Ramesh Konatala  
(4749146)

Alan van Rossum  
(4293932)

### 1. Introduction

Structure from motion (SfM) is a photogrammetric range imaging technique for estimating three-dimensional structures from two-dimensional image sequences. A typical 3D reconstruction procedure involves finding correspondence between images through feature detection, matching the detected features from one image to another, filtering correspondences, reconstructing 3D positions by tracking feature trajectories over time. However estimating SfM is still a challenging task owing to several limitations in tracking of features across sequences images. This assignment provides a method to estimate SfM of sequence of 19 model castle images taken from different viewing positions. One of the sample image among 19 sequences is shown in figure 1



Figure 1. Model castle

### 2. Components of Project

The project is divided into following components : Section 2.1 Feature point extraction, de-

criptor determination and finding matches, Section 2.2 Normalized 8-point RANSAC algorithm to find best matches, Section 2.3 Creating point-view matrix to represent point correspondences for different camera views, Section 2.4 Tomasi Kanade factorization of point view matrix block, Section 2.5 Eliminating affine ambiguity, Section 2.6 Plotting 3D point cloud of castle, Section 3 Finally results of the project and further improvements are discussed.

#### 2.1. Feature point detection and the extraction of SIFT descriptors

For each image, we obtain scale and affine invariant feature detectors using Harris/Hessian Affine implementation. Each feature points is described by a 128 bit descriptor which can be used to detect potential matches in image sequences.

Once feature points and descriptors are obtained for all the image sequences correspondence matches are obtained for image sequences 1-2, 2-3, 3-4 ... 19-1. Correspondence matches for one of the sequences is visualized in figure 2.

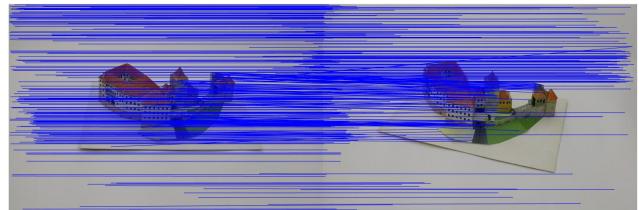


Figure 2. Matches for image sequence 1-2

It is observed that by increasing the matching threshold for potential matches across image pairs leads to removing certain percentage of outliers(matching threshold is a hyper parameter for vlubcmatch matlab vlfeat library function). Also

the feature points that lie outside the model castle zone are eliminated for stable results.

## 2.2. Normalized 8-point RANSAC

The given feature points still suffer from the problem of outliers and thus we use Normalized 8-point RANSAC method to get best inliers. Firstly the given 2D coordinates of the image sequence correspondences are converted into homogeneous coordinates. Then the coordinates are then normalized as it can lead to better conditioning of the problem and stability of the results. Next using iterative RANSAC algorithm we will first randomly select 8 matches and fundamental matrix is estimated using these correspondences. The 8 random matches are selected such that the points are well distributed across the image. For this we divide the image into 8 zones as shown in figure 3 and each of the 8 random points are selected such that each point belongs to only one zone. Further singularity of the fundamental matrix constraint is enforced by adjusting the entries of the estimated fundamental matrix.

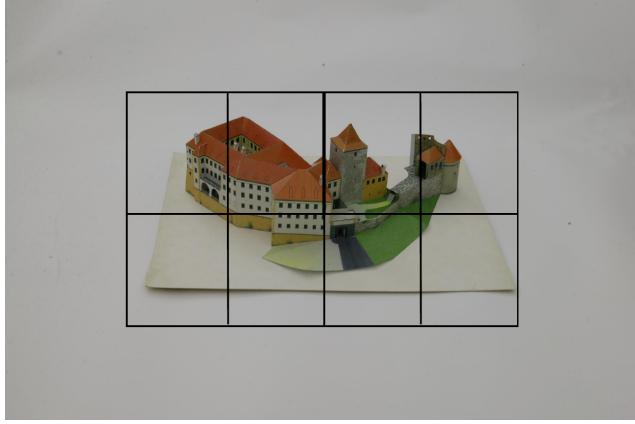


Figure 3. 8 different zones

Using fundamental matrix we will estimate the transformed coordinates of feature points from first image onto the second images. Then the distances of the estimated coordinates to that of the real feature coordinates on the second images are compared and number inliers are estimated. We used Sampson distance to estimate distances and all those points above certain distance threshold are eliminated. It is observed that by decreasing the distance threshold certain outliers can be elim-

inated. This is performed for certain predefined number of iterations(10000 in this case) such that best 8 points that lead to most number of inliers are identified. The result of RANSAC that maximized the inliers with best 8-points can be visualized as follows. The figure 4 shows best 8-point combination for the image sequence 8-9 that maximizes the number of inliers. The figure 5 shows the obtained inliers and their correspondences.

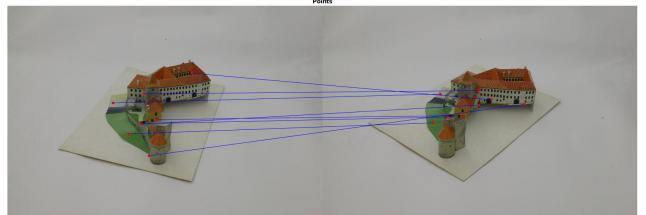


Figure 4. 8-points for image sequence 8-9

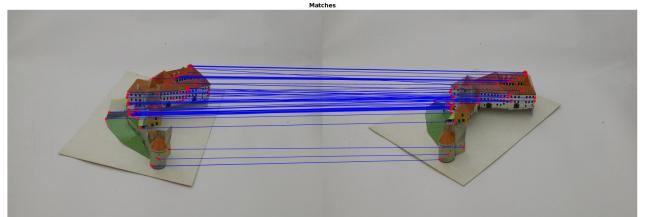


Figure 5. Inliers for image sequence 8-9

## 2.3. Chaining

After obtaining robust inliers across the sequences of the images we construct the point view matrix of size  $2m * n$  where m is number of views and n is the number of points observed over all the views. Thus each column in the point view matrix represents the coordinates of the observed points across different views which is indicated rows wise. If a point is observed across several views we can consider that the observed point is a robust inlier which can be used to estimate shape and motion matrices through factorization.

## 2.4. Stitching

### 2.4.1 Point View Matrix Segmentation

Point view matrix is segmented such that it includes observed in points in three consecutive views. Estimation of 3D point reconstruction would be better if the point view matrix segment

is denser. A segment of the constructed point view matrix is shown in figure 6. The white dots represent points observed across different views.



Figure 6. A segment of point view matrix

#### 2.4.2 Tomasi-Kanade factorization

The obtained segmented point view matrix is used to estimate 3D points using Tomasi-Kanade factorization. The matrix is decomposed using svd decomposition from which the shape and motion matrices for the observed points are constructed. The following figure 7 shows 3D plot of the obtained correspondences after Tomasi kanade factorization.

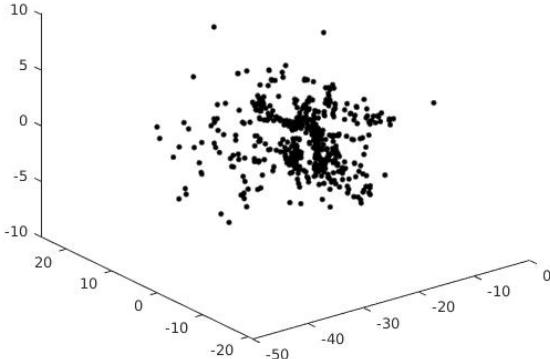


Figure 7. observed 3D points after factorization

#### 2.4.3 Procrustes analysis

After obtaining the 3D point sets for different segments the optimal transformation between different shared points is obtained using MATLAB Procrustes function. The following figure 8 shows 3D plot of the obtained correspondences after procrustes analysis.

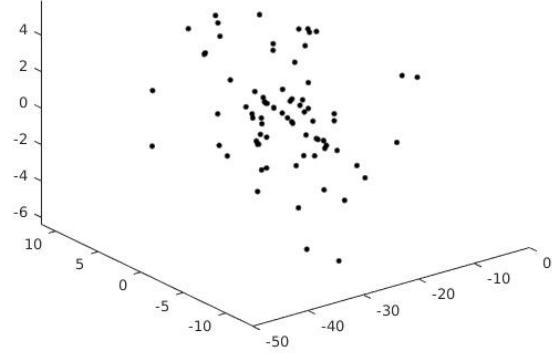


Figure 8. observed 3D points after Procruster analysis

#### 2.5. Affine ambiguity

The obtained 3D points still suffer from affine ambiguity problem. This is mitigated by enforcing the euclidean constraints where we have enforced the image axes to be perpendicular. This is done by using cholesky decomposition method.

#### 2.6. 3D Model plotting

The observed 3D coordinates are plotted using MATLAB trisurf function and can be visualized in the figure 9.

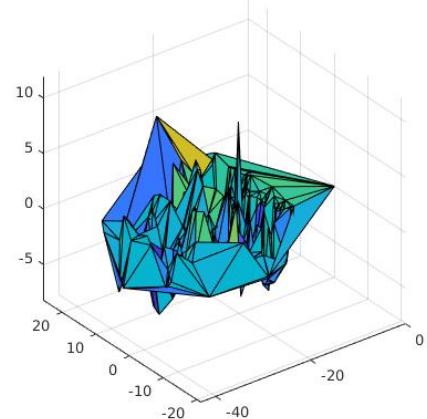


Figure 9. 3D plot

### **3. Conclusion and scope of further improvements**

After performing several experiments it is observed that identifying robust inliers is most important to create a dense point view matrix which can lead to better estimation of 3D point coordinates.

We can improve the density of the point view matrix by considering all possible image pairs and updating corresponding columns like considering 1-2, 1-3, 1-4 .... 1-19; 2-3, 2-4 .... 2-19; .....; 17-19 image pairs. This method can be used to determine next best view as suggested in paper [1] and then construct the point view matrix.

## **References**

- [1] Tomasi, Carlo and Kanade, Takeo: Shape and motion from image streams under orthography: a factorization method
- [2] Schonberger, Johannes L and Frahm, Jan-Michael: Structure-from-motion revisited