3D Reconstruction from 2D images Using Structure from Motion Algorithm

Muhammad Osama Nusrat
MS AI
i212169@nu.edu.pk

Abstract

This Project aims towards development of 3D model of an object based on the methods used in the paper "Building Rome in a day". In this paper, the collection of images of Rome were used from the massive library and a new distributed matching system was developed which had the capability of matching images very quickly using a bundle adjustment software. This software can calculate solutions to complex non-linear least squares problems. We reproduced this piece of work on a small scale, since such large data set handling and computation was beyond the limit of available resources. The main motive is to perform a 3D reconstruction from 2D camera images so that some sense of depth of the object can be inculcated (fix). Motion can be with respect to an observer or the observed objects moving or both. In our case the observer which was the camera had the freedom to move while the scene was static. The scene consisted of a lamp and the main purpose was to perceive the 3 dimensions of that lamp up to a certain scale. With the help of a simple mobile phone camera, we made it possible to visualize the lamp in 3D. This was done by capturing images, calibrating cameras, processing those images using computer vision technique of incremental structure from motion.

1. Problem statement

The projects aim to reconstruct a 3D model of an object using multiple 2D images. The paper I took is building a Rome in a day. There was an issue in implementing the data set used in the project. I had to use a camera with the same calibration settings, which means I need the same calibration settings from which the data set photos were taken. That's why I used my own data set.

2. Introduction

- Structure from motion is a technique used in computer vision that uses a collection of overlapping 2D images to develop a 3D model of the scene captured by the images.
- It is used in many disciplines where 3D information is vital for further processing like autonomous navigation, trajectory tracking, and surface conditioning.
- The best thing about this algorithm is that it is applicable on low cost cameras, hence it is an efficient technique for the 3D problems. For our case, we used a single mobile camera since it seemed the most suitable option.
- Recently, SFM has been integrated with technologies such as stereo and lasers to produce an absolute reconstruction of the scene, correct to a margin of few millimetres.
- There are some preliminary steps required to develop an SFM pipeline. The first of them is to determine the intrinsic and extrinsic matrix (optional) of the camera.
- This can be done by using the MATLAB camera calibration app. Intrinsic matrix contains different parameters like focal length (in pixels), resolution of image captured by camera etc.collectively used with camera extrinsic matrix which contains distance information with respect to the real world coordinates.

- The second step in the pipeline is to capture the images and sequentially process them to detect matching keypoints in the images.
- With the help of the matching keypoints and assuming the first camera pose to be a world coordinate (0,0,0) we can easily predict the new camera poses which represents their orientation and location with respect to the first camera position.
- The algorithms used for matching keypoints in images were SIFT/SURF.
- To run a reconstruction, we used non linear optimization techniques that involved concepts such as fundamental matrix, essential matrix, projection matrix, and bundle adjustment.
- These are discussed later in detail in the methodology section. Thus, the end goal of mapping the 2D image pixels to their respective 3D world position was obtained.
- Another objective was to compute a dense reconstruction if the key points in images were sparse(dispersed or scattered).

3. Literature Review

3.1 Structure From Motion

- Structure from motion is basically used to predict the geometry of a material.
- Multiple images from different camera angles are taken of same object and then we match the features of 2 images and reconstruct the 3d model of the object.

3.2 STEPS OF SFM

- First of all take multiple images of your desired object from different camera positions.
- Then we take one image at a time and undistort that image.

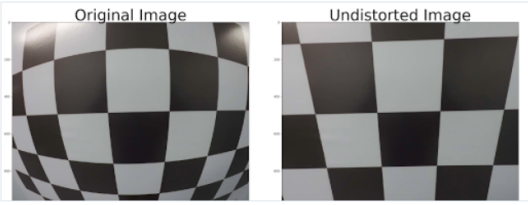

Figure 1. Original Undistorted Image

- In figure below we have a checkers board original image is distorted so we have to undistort it.
- The image taken by first camera position is then taken as a refernce and we set its coordinate as (0,0,0).
- After that the image background is changed from rgb to gray because SFM works at gray scale images.
- In next step SFM detect key points in our image. Keypoints are also called features of an image.
- Region of interest is found in our image and we crop it.
- From keypoint we can make a feature vector
- Feature vector are used to find the matching points in the two images.
- Same step is repeated from second image which is taken from different camera position. SFM detects key points from that image also.
- Note both images are of same object. The difference is that they are taken at different camera positions.
- Now these key features of both images will help to generate projection matrix
- Camera has 2 parameters intrinsic and extrinsic parameters. In intrinsic parameter include focal length and image size and extrinsic parameters include rotation and translation.
- when intrinsic and extrinsic matrix are combined we get projection matrix. With the help of projection matrix we can find a 3d points of an image.
- At this stage we have found matching points in two images and we know position of camera 1 which is origin.

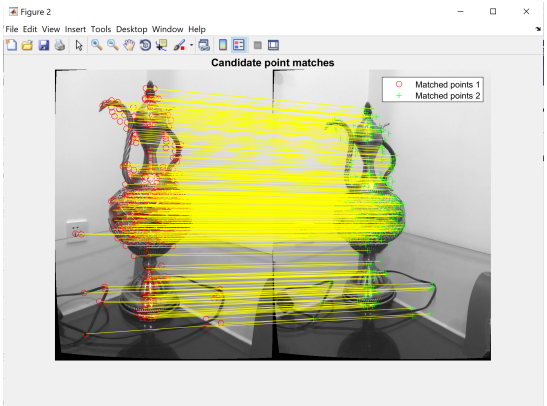

Figure 2. Feature Matching of Images

- Now we need to find position of camera 2.

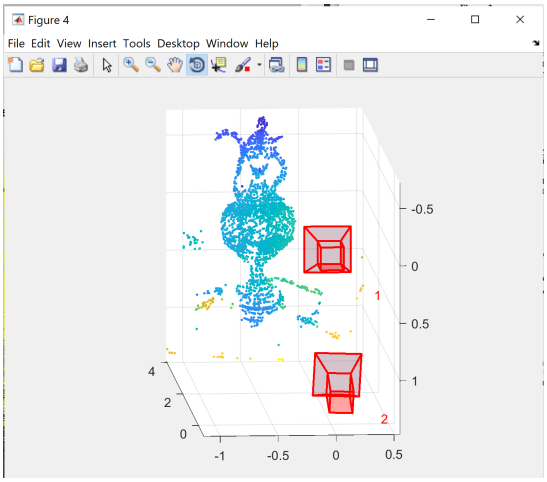

Figure 3. 3D reconstruction

- After finding position of camera 2 we can find 3d points.
- Bundle adjustment is used to reduce error incase our 3d points are not accurate.

3.3 What is SIFT/SURF

- Sift detects features in an image which are also called key points.
- We can make 3d model from a set of 2d images using SIFT, a scale-invariant feature transform. What it does is that it takes two images of the same object at different camera orientations and then find the key points in an image and match the key point.
- It is called scale-invariant because we take images at different camera orientations but the point is that by changing orientation, features of image doesn't change.

4. Methodology

4.1 Camera Model

- A camera is represented by a matrix PP which transforms a point X in the real world to a pixel x in the image.
- This relationship is given by the following equation

$$x = PX$$

To be able to find X, we need to do the following

$$X = P^{-1}x$$

- In order to do the 3D reconstruction, we need to find the camera pose represented by the above camera matrix P for each of the images (the 2D snapshots or views).

4.2 Incremental SfM

So, finding the camera matrix or the camera pose for each of the images (the 2D snapshots or views) is a part of the process of reconstructing 3D coordinates.

SfM, thus, involves estimating the 3D points along with the camera pose from a sequence of images.

The 3D points are recovered by a procedure known as triangulation that uses camera poses to accurately locate the points in space.

Incremental SfM chooses two images as the baseline views, obtains an initial reconstruction, and incrementally adds new images.

The first camera position is assigned location of world coordinate (0,0,0) and the rest is computed as the images come along.

4.3 Two view geometry

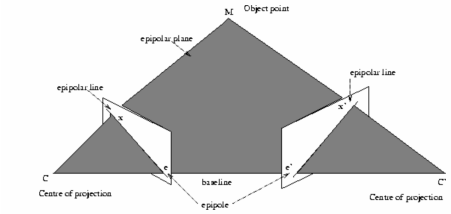

Figure 1: *Two-view geometry.* ***Credits.***

Figure 4.

- M is the point being photographed while x and x' are its projections in the two camera images. C and C' is the position of the two cameras. e' and e are the epipoles. x'e' and xe are the epipolar lines and all these points lie on the epipolar plane
- To find the coordinate in the second view, we have to look across the epipolar line. This relationship is described by the fundamental matrix F, which can be calculated from the camera matrix P.
- The F matrix can be used to triangulate matching points for their corresponding 3D point. The F matrix is replaced by the E matrix in case of calibrated cameras.
- The projection matrix P for calibrated cameras is shown below. It converts a 3D world point to a 2D

image pixel. Coordinates are represented as homogeneous.

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

2D Image Coordinates | Intrinsic properties (Optical Centre, scaling) | Extrinsic properties (Camera Rotation and translation) | 3D World Coordinates

Figure 2: *The camera matrix or projection matrix.* Credits.

Figure 5.

4.4 Point Correspondences

- Once we have a set of corresponding points using techniques like SIFT/SURF, we refine these matches and filter the wrong matches using RANSAC.
- The 8 point algorithm is used to compute the fundamental matrix.
- It requires a minimum of 8 points, however the more good matching points the better.

$$\begin{bmatrix} u_1u'_1 & v_1u'_1 & u'_1 & u_1v'_1 & v_1v'_1 & v'_1 & u_1 & v_1 & 1 \\ u_2u'_2 & v_2u'_2 & u'_2 & u_2v'_2 & v_2v'_2 & v'_2 & u_2 & v_2 & 1 \\ u_3u'_3 & v_3u'_3 & u'_3 & u_3v'_3 & v_3v'_3 & v'_3 & u_3 & v_3 & 1 \\ u_4u'_4 & v_4u'_4 & u'_4 & u_4v'_4 & v_4v'_4 & v'_4 & u_4 & v_4 & 1 \\ u_5u'_5 & v_5u'_5 & u'_5 & u_5v'_5 & v_5v'_5 & v'_5 & u_5 & v_5 & 1 \\ u_6u'_6 & v_6u'_6 & u'_6 & u_6v'_6 & v_6v'_6 & v'_6 & u_6 & v_6 & 1 \\ u_7u'_7 & v_7u'_7 & u'_7 & u_7v'_7 & v_7v'_7 & v'_7 & u_7 & v_7 & 1 \\ u_8u'_8 & v_8u'_8 & u'_8 & u_8v'_8 & v_8v'_8 & v'_8 & u_8 & v_8 & 1 \end{bmatrix} \begin{bmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{bmatrix} = 0$$

Figure 3: *System of equations for the 8-point algorithm.* Credits.

Figure 6.

4.5 View

- From the fundamental matrix, we calculate the new view which represents the position of the incoming camera.
- Thus, now we have the camera locations of the two cameras and the matching points in images taken by those cameras
- We can triangulate them for the corresponding 3D world points.
- The 3D points can be kept track of using a point tracker.
- Thus, with every incoming camera, we can update the point cloud with the new points and thus our reconstruction gets better
- Another algorithm to refine the reconstruction is bundle adjustment which is merely a non linear least square optimization algorithm that can help refine camera poses by minimising re-projection errors.
- The refined point cloud can be used to extract valuable depth information about the object, or the shape of the object, or even create a map of the surroundings.

5. Evaluation and Experiments

5.1 Camera Calibration

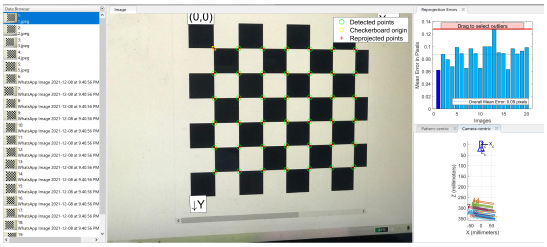

Figure 7. Camera Calibration

- In order to transform from 2D world coordinates (Camera Coordinates) to 3D World Coordinates, there are some transformations which needs to be done using some certain camera characteristics (Intrinsic + Extrinsic).
- Intrinsic parameters deal with the camera's internal characteristics, such as, its focal length, skew, distortion, and image center
- Extrinsic parameters describe its position and orientation in the world
- Knowing intrinsic parameters is an essential first step for 3D computer vision, as it allows you to estimate the scene's structure in Euclidean space and removes lens distortion, which degrades accuracy.
- Camera calibration contains both Intrinsic and extrinsic camera calibration.

5.2 Camera Intrinsic

- Intrinsic matrix of camera can be given as:

$$M = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Figure 8.

- Where (fx, fy) are focal point and (cx, cy) are principal point.
- Generally, the camera calibration process uses images of a 3D object with a geometrical pattern (e.g. checker board).
- The pattern is called the calibration grid. The 3D coordinates of the pattern are matched to 2D image points.

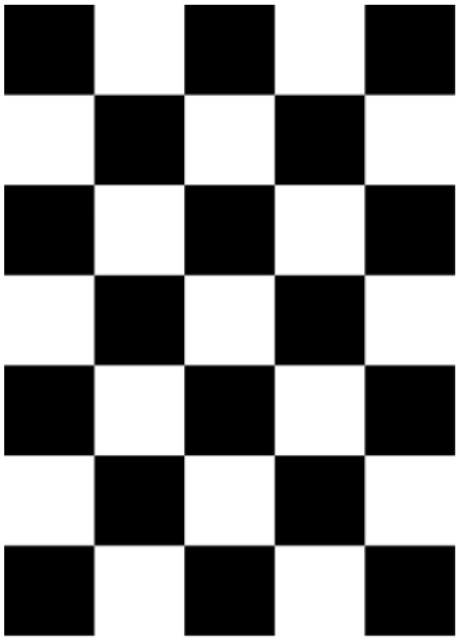

Figure 9. Calibration Grid

In order to measure the intrinsic parameters of camera, we have followed the following steps:

- Take 15-20 pictures of checkerboard patterns from different angles.
- Open the Matlab camera calibrator app from Apps –¿ Image processing and Computer Vision –¿ Camera Calibrator.
- Load all the images into calibrator app

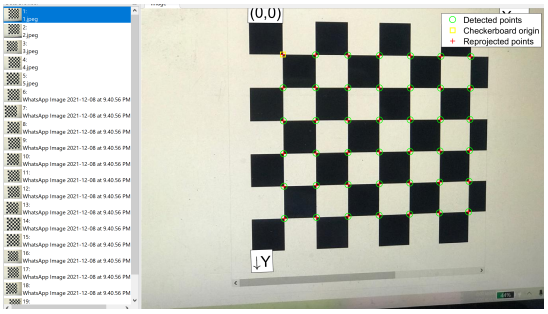

Figure 10. Images Loaded for Calibration

- App will automatically detect the checkerboard patterns from all the images
- Click on calibrate button
- After processing, Camera intrinsic parameters will be calibrated. One can import the parameters to workspace and can also generate the file for auto calibration for future needs.

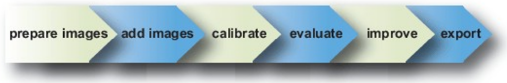

Figure 11. Flow Chart of Calibration of Images

5.3 Camera Extrinsic

- Camera Extrinsic describes the camera position and orientation with respect to certain world coordinates.
- They are calibrated with respect to certain reference world coordinates.
- Matlab perceive top left corner of image as (x, y) = (0,0). x increases horizontally, from left to right, while y increases vertically, from top to bottom.

6. Conclusion

- This project was a great source of learning for me. I had certain challenges in implementing it especially in understanding the theory and mathematical terms.

References

[1] Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S. M., Szeliski, R. (2011). Building Rome in a day.Communications of the ACM,54(10), 105-112.