Modelling Project Work: 3D reconstruction using Stereo vision

Yeraly Kalel
Supervisor: Hajder Levente

December 8, 2020

Introduction

Study of the computer vision allows us to understand the interaction between digital images and the physical environment and use this knowledge to automate the processes of humankind. One of the challenging fields in computer vision is a 3D reconstruction, which is the process of capturing the shape and appearance of real objects. Using 3D reconstruction one can determine an object's 3D profile, as well as knowing the 3D coordinate of any point on the profile. 3D reconstruction has plenty of applications in different fields such as computer graphics, computer animation, computer vision, medical imaging, computational science, virtual reality, digital media, etc. The 3D object or scene can be constructed from its point cloud, thus finding the point cloud of the object or scene is an essential task. This work is going to overview and implement the methods of estimating point cloud from known point correspondences and intrinsic parameters of the camera or/and extrinsic parameters as well.

Theoretical background

Point representation

The model of the image is the 2D projective plane $\mathbb{P}^2$ (see Figure 1).

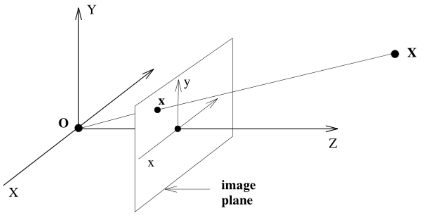


Figure 1. Point projection into the image plane [1]

A point $(x, y)^T$ in the image is transferred into homogeneous coordinates by 3D vector $(X, Y, W)^T$, where $x = X/W$ and $y = Y/W$. This modification allows representing any transformation between points by matrix. In this same way, one can define any 3D point $(x, y, z)^T$ as $(X, Y, Z, W)^T$, where $x = X/W$, $y = Y/W$ and $z = Z/W$. Most of the time W is 1 for simplification of calculations.

Connection between 3D world point and 2D image point

In pinhole camera model, a mapping from world coordinates into pixel coordinates is given by:

$$\mathbf{p} = \mathbf{P}\mathbf{X} \tag{1}$$

where $\mathbf{p} \in \mathbb{R}^{3\text{x}1}$ presents point in image, and it is represented by $\begin{bmatrix} u & v & 1 \end{bmatrix}^T$ without loss of generality; $\mathbf{X} \in \mathbb{R}^{4\text{x}1}$ and it is represented by $\begin{bmatrix} X & Y & Z & 1 \end{bmatrix}^T$ without loss of generality; $\mathbf{P}$ denotes 3x4 projection matrix. Moreover, projection matrix is decomposed by:

$$\mathbf{P} = \mathbf{K}[\mathbf{R}|\mathbf{t}], \tag{2}$$

where **K** is camera matrix; **R** and **t** are rotation matrix and translation vector between world and camera coordinates, respectively.

Estimating 3D point from 2D point using stereo vision

Eq. 1 can be rewritten as follows:

$$\mathbf{p} \times \mathbf{P}\mathbf{X} = \mathbf{0} \tag{3}$$

By writing all three resultant equations:

$$\begin{aligned} u\mathbf{p_3}^T\mathbf{X} - \mathbf{p_1}^T\mathbf{X} &= 0 \\ v\mathbf{p_3}^T\mathbf{X} - \mathbf{p_2}^T\mathbf{X} &= 0 \\ u\mathbf{p_2}^T\mathbf{X} - v\mathbf{p_1}^T\mathbf{X} &= 0, \end{aligned} \tag{4}$$

where $\mathbf{p}_i$ is i-th row of projection matrix **P**. Since third equation can be expressed by first two, it will be dropped out. Since stereo vision is used following system of linear equation is formed:

$$\begin{bmatrix} u^{(1)}(\mathbf{p_3}^T)^{(1)} - (\mathbf{p_1}^T)^{(1)} \\ v^{(1)}(\mathbf{p_3}^T)^{(1)} - (\mathbf{p_2}^T)^{(1)} \\ u^{(2)}(\mathbf{p_3}^T)^{(2)} - (\mathbf{p_1}^T)^{(2)} \\ v^{(2)}(\mathbf{p_3}^T)^{(2)} - (\mathbf{p_2}^T)^{(2)} \end{bmatrix} \mathbf{X} = \mathbf{0} \tag{5}$$

Here, coefficient matrix is $\mathbf{A} \in \mathbb{R}^{4\times4}$. During solving this homogeneous system of linear equations, scale of **X** will be lost.

Epipolar geometry

According to Hartley and Zisserman [2], “The epipolar geometry is the intrinsic projective geometry between two views. It is independent of scene structure, and only depends on the cameras’ internal parameters and relative pose.”
One of its main concepts is fundamental matrix, which encapsulates intrinsic projective geometry in stereo vision. Besides, each PC must satisfy the following relation:

$$(\mathbf{p}^{(2)})^T\mathbf{F}\mathbf{p}^{(1)} = 0, \tag{6}$$

where **F** is a singular 3x3 fundamental matrix. In the matrix format:

$$\begin{bmatrix} u^{(2)} & v^{(2)} & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} u^{(1)} \\ v^{(1)} \\ 1 \end{bmatrix} = 0 \tag{7}$$

For arbitrary n correspondences (7) can be rearranged to the following form:

$$\begin{gathered}\begin{bmatrix} u_1^{(2)}u_1^{(1)} & u_1^{(2)}v_1^{(1)} & u_1^{(2)} & v_1^{(2)}u_1^{(1)} & v_1^{(2)}v_1^{(1)} & v_1^{(2)} & u_1^{(1)} & v_1^{(1)} & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_n^{(2)}u_n^{(1)} & u_n^{(2)}v_n^{(1)} & u_n^{(2)} & v_n^{(2)}u_n^{(1)} & v_n^{(2)}v_n^{(1)} & v_n^{(2)} & u_n^{(1)} & v_n^{(1)} & 1 \end{bmatrix} \\ \begin{bmatrix} f_{11} & f_{12} & f_{13} & f_{21} & f_{22} & f_{23} & f_{31} & f_{32} & f_{33} \end{bmatrix}^T = \mathbf{0} \end{gathered} \tag{8}$$

The matrix with known variables has a size of $n \times 9$, and one can choose the value of the norm of the unknown vector arbitrarily. Thus, eight or more correspondences are required to solve this system. Also, Figure 2 portrays geometric relations between pairs of projected points.

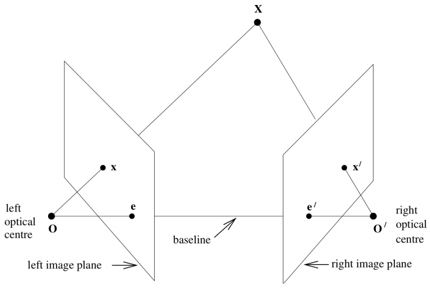


Figure 2. Representation of fundamental matrix [1]

Another concept in epipolar geometry is an epipole or an epipolar point. It is the point created from the intersection between the image plane and line joining the camera centers, which is the baseline. Both of the images have epipolar points, and they have the following structure: $\mathbf{e} = \begin{bmatrix} e_u & e_v & 1 \end{bmatrix}^T$ and $\mathbf{e}' = \begin{bmatrix} e'_u & e'_v & 1 \end{bmatrix}^T$, where $\mathbf{e}$ and $\mathbf{e}'$ are the epipoles for the first and second images, respectively. The relationship between epipolar points and the fundamental matrix has the following form:

$$\begin{aligned} \mathbf{Fe} &= \mathbf{0} \\ \mathbf{F}^T\mathbf{e}' &= \mathbf{0} \end{aligned} \tag{9}$$

The point that has corresponding point on the second image, must locate on the line specified by the Fundamental matrix and that corresponding point. The lines are called epipolar lines and they can be formulated as:

$$\begin{aligned} \mathbf{l}^{(2)} &= \begin{bmatrix} a' \\ y' \\ c' \end{bmatrix} = \mathbf{Fp}^{(1)} \\ \mathbf{l}^{(1)} &= \begin{bmatrix} a \\ y \\ c \end{bmatrix} = \mathbf{F}^T\mathbf{p}^{(2)} \end{aligned} \tag{10}$$

Set of $\mathbf{l}^{(2)}$ lines coming from all $\mathbf{p}^{(1)}$ of the first image has to be intersected at the epipole of the second image. The same is true for the other image.
Essential matrix is the special case of the fundamental matrix when image coordinates normalized by camera:

$$(\hat{\mathbf{p}}^{(2)})^T\mathbf{E}\hat{\mathbf{p}}^{(1)} = 0, \tag{11}$$

where $\hat{\mathbf{p}}^{(i)}$ is $(\mathbf{K}^{-1})^{(i)}\hat{\mathbf{p}}^{(i)}$ for $i = 1, 2$ and $\mathbf{E}$ is an 3x3 essential matrix. Essential matrix can be estimated as fundamental matrix, but the only difference is that it needs minimum five points.

Essential matrix decomposition

The essential matrix defines rotation and translation variables between two cameras:

$$\mathbf{E} = [\mathbf{t}]_\times \mathbf{R} \tag{12}$$

It is worth noting that $[\mathbf{t}]_\times$ is skew symmetric matrix. The decomposition of $\mathbf{E}$ defines four possible values of projection matrix and only one of them is correct (see Figure 3). One of the properties of the essential matrix is that two of its singular values are equal and third one is zero. Suppose that $\mathbf{P}^{(1)} = \mathbf{K}^{(1)}[\mathbf{I}|\mathbf{0}]$ and SVD of $\mathbf{E}$ is $\mathbf{U}diag(1,1,0)\mathbf{V}^T$, then four solutions are:

1. $\mathbf{P}^{(2)} = \mathbf{K}^{(2)}[\mathbf{UWV^T}|{+\mathbf{u_3}}]$
2. $\mathbf{P}^{(2)} = \mathbf{K}^{(2)}[\mathbf{UWV^T}|{-\mathbf{u_3}}]$
3. $\mathbf{P}^{(2)} = \mathbf{K}^{(2)}[\mathbf{UW^TV^T}|{+\mathbf{u_3}}]$
4. $\mathbf{P}^{(2)} = \mathbf{K}^{(2)}[\mathbf{UW^TV^T}|{-\mathbf{u_3}}]$

$+\mathbf{u_3}$ is third column of matrix $\mathbf{U}$ and $\mathbf{W}$ is orthogonal matrix:

$$\mathbf{W} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{13}$$

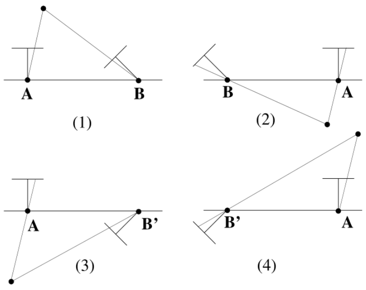


Figure 3. The four possible solutions of the projections [1]

Homogeneous system of linear equations

Any system of linear equations in the form of $\mathbf{Ax} = \mathbf{0}$, where $\mathbf{A} \in \mathbb{R}^{k \times n}$ is matrix of known variables, while $\mathbf{x} \in \mathbb{R}^{nx1}$ is a vector of independent unknown variables; can be solved using lagrange-multipliers if $k \geqslant n-1$ by constraining the norm of $\mathbf{x}$ to be any arbitrary value, usually it is 1. Thus, solution of $\mathbf{x}$ is the (one dimensional) kernel of $\mathbf{A}$ and it is an eigenvector with at least eigenvalue of $\mathbf{A}^T\mathbf{A}$ subjected to $\|\mathbf{x}\| = 1$. If $k = n-1$, the solution is solved exactly and if $k > n-1$, the system is overdetermined.

Data normalization

Since linear system of equations are solved here, point normalization have a positive effect on increase of condition number of the the coefficient matrix of the fundamental and essential matrix, which ensures to estimate the inverse of those safely. Point normalization is done in the following manner for each image independently:

- Translate the points such that centroid is at the origin:

$$translated(\mathbf{p}_i) = \mathbf{p}'_i = \mathbf{p}_i - average(p) \qquad \forall i \tag{14}$$

- Scale points so that the average distance from origin is $\sqrt{2}$:

$$normalized(\mathbf{p}_i) = \bar{\mathbf{p}}_i = \frac{\sqrt{2}n}{\sum_{j=1}^{n}\sqrt{p_{j_x}^2 + p_{j_y}^2}}\mathbf{p}'_i \qquad \forall i \tag{15}$$

  , where n is number of points, $\mathbf{p} = \left\{ \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}, \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix}, ..., \begin{bmatrix} x_n \\ y_n \\ 1 \end{bmatrix} \right\}$, p_{j_x} is x_j and p_{j_y} is y_j.

Structure of this manipulation in the matrix format looks as following:

$$\begin{gathered} \bar{\mathbf{p}}_i = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & t'_x \\ 0 & 1 & t'_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \\ \begin{bmatrix} s_x & 0 & s_x t'_x \\ 0 & s_y & s_y t'_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & t_x \\ 0 & s_y & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \mathbf{T}\mathbf{p}_i \end{gathered} \tag{16}$$

Clustering algorithm

Unfortunately, processing data may contain outliers, which are elements of the dataset that do not support the prevailing model, and they have to be detected to make sure that result is correct. One of the clustering techniques is the random sample consensus (RANSAC), which segments data into inliers and outliers, which solves this problem. It is an iterative non-deterministic algorithm producing acceptable results with a certain probability. Each iteration has the following steps.

- Randomly selects a subset from the dataset, with size n, which is a minimum number of elements to describe the model, i.e., the DoF of the model.
- Define the model using those hypothetical inliers.
- Test dataset using the defined model according to some loss functions. If an element has a loss value under the specified threshold, then it is considered as an inlier; otherwise, it is an outlier.
- The model is identified as a better one among previously defined models if it fits more data than them.

It is noteworthy that the user specifies the number of iterations and threshold for the loss value. It might be the case that the user may specify a higher number of iterations than it should be to estimate the predominant model. The maximum number of iterations can be predicted under some confidence level by:

$$k = \frac{\log(1-p)}{\log(1-w^n)}, \tag{17}$$

where k is the predicted maximum number of iterations, p is the confidence level, w is the inlier ratio, and n is the minimum number of elements needed to construct the model. Furthermore, RANSAC can be fastened by adding local optimization at the iterations where a better model is found [3]. This extra work might increase the time of RANSAC; however, they proved controversial by the experiment. The least-square method is implemented in this work as local optimization. After finding each better model among previous ones, a hypothetical model is computed using all inliers found by the minimum number of elements. Then, the better model is chosen among the two.

Sometimes, data consists of multiple models, and simple RANSAC will find only the predominant one. Usage of Sequential RANSAC can be beneficial since it can overcome this issue. It executes RANSAC iteratively, and after each increment, it removes inliers of the predominant model from the data and seeks to find another model using the resultant outliers.

Testing

Only two cases will be considered, when projection matrix is prior known and camera matrix is given. It may be noticed that third case, which is dealing with unknown projection matrix and camera matrix is not considered. As it was overviewed before, the rotation and translation variables are estimable from the essential matrix. RANSAC is used for estimating the essential matrix and fundamental matrix. Technically, we don't require the fundamental matrix for our computation, but we can check the result of essential by visualizing the fundamental matrix.

The testing of the feasibility of the case with known camera parameters demanded the construction of the synthetic scene. Specifically, three mutually perpendicular chessboard planes are constructed and used as the synthetic scene. Later, at two different angles images are photographed without losing any square on the planes (see Figure 4). Moreover, only the inner vertices are selected to be point correspondences. Overall, each chessboard plane has 48 appealing points.

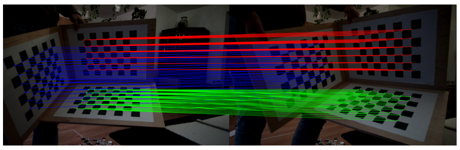
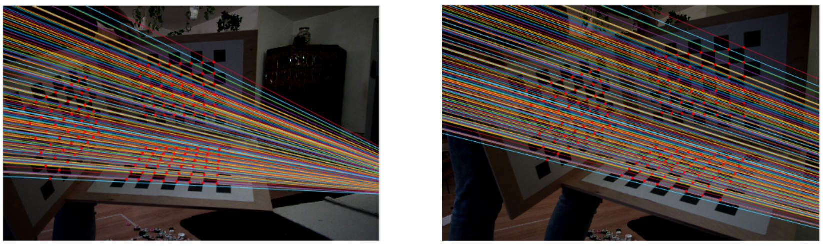


Figure 4. Synthetic data and correspondences in them

Fundamental matrix then is estimated using those 48 pair points and epipolar lines are presented in Figure 5.



(a) First image

(b) Second image

Figure 5. Epipolar lines for the first and second image for chessboard case

This result looks good and the essential matrix now can be calculated from the fundamental matrix. After decomposing the essential matrix, four possible solutions are obtained and only one of them is good (see resultant good 3D point cloud in Figure 6).

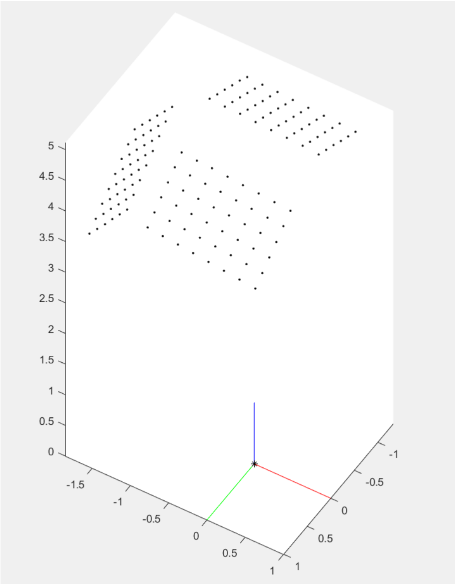


Figure 6. 3D point cloud of the 3 planes of the chessboard. Red, greed and blue lines represent x, y and z axes, respectively

For testing the case when projection matrix is already known, a dinosaur from dataset from Visual Geometry Group Of Oxford University is used [4]. Since stereo vision is used and there are 36 different images, all possible image pairs are used which is $\binom{36}{2}$. Obtained 3D point clouds using stereo vision is represented in Figure 7.

(a) One view

(b) Second view

Figure 7. 3D point clouds of dinosaur

The result is almost perfect, but there are some points far from the dinosaur. It can be explained by the fact that coordinates of some point correspondences are not exactly true and they are located at the edge of the dinosaur. This small error may the solver think that point is behind the dinosaur.

Bibliography

[1] Andrew Zisserman. "Geometric Framework for Vision I: Single View and Two-View Geometry". In: *Lecture Notes, Robotics Research Group, University of Oxford* (1997).

[2] Richard Hartley and Andrew Zisserman. "Multiple View Geometry in Computer Vision (Second Edition)". In: *Cambridge University Press* (2003).

[3] Ondrej Chum, Jiri Matas, and Josef Kittler. "Locally optimized RANSAC". In: *Joint Pattern Recognition Symposium*. Springer. 2003, pp. 236–243.

[4] *Multi-view and Oxford Colleges building reconstruction*. `https://www.robots.ox.ac.uk/~vgg/data/mview/`. Accessed: 2020-12-07. 2009.