Automatic 3D Model Construction for Turn-Table Sequences - a simplification

Fredrik Larsson
larsson@isy.liu.se

April 11, 2011

1 Background

This report introduces some simplifications to the method by Fitzgibbon et al. [1] that allows for 3D model construction from turn-table sequences. It is assumed that the reader has previously read [1] in order to fully understand this report.

Fitzgibbon et al. presents a method for 3D model construction that utilizes the extra constraints imposed by turn-table sequences. Restricting the scenario to a turn-table sequence with a single camera with fixed settings produces these extra constraints:

C1. The internal parameters for the camera are the same for all images

C2. The motion of the camera can be described by a rotation around a single axis

It is shown that in the uncalibrated case the number of parameters to estimate is $m + 8$ where m is the number of images.

We further simplify the problem by using extra constrains given from the fact that we know:

C3. The internal parameters of the camera, i.e the K matrix

C4. That the angle between each pair of consecutive cameras is the same

Using these extra simplifications makes it possible to create a 3D model from realistic data without using Bundle Adjustment.

2 Method

The effects of the additional constraints C3-C4 are presented in this section. For full details regarding the original method see [1].

2.1 Original

The world coordinate system is chosen such that the rotation axis coincides with the z-axis. Using the notation in [1], the first camera may be written as

$$\mathbf{P}_0 = \mathbf{H}[\mathbf{I}|\mathbf{t}] \tag{1}$$

where $\mathbf{H}$ is a homography representing the camera internal parameters and a rotation about the camera center, and $\mathbf{t} = (t, 0, 0)^T$. The subsequent cameras are then given by post-multiplying $\mathbf{P}_0$ with

$$\begin{bmatrix} \mathbf{R}_z(\theta_i) & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix} \tag{2}$$

yielding the cameras

$$\mathbf{P}_{\theta_i} = \mathbf{H}[\mathbf{R}_z(\theta_i)|\mathbf{t}] \tag{3}$$

The paper [1] describes a method for estimating $\mathbf{H}$ and the different θ_i.

2.2 Simplified

Simplification C3 allows us to write (3) as

$$\mathbf{P}_{\theta_i} = \mathbf{KR}[\mathbf{R}_z(\theta_i)|\mathbf{t}] \ . \tag{4}$$

Simplification C4 allows us to further write (4) as

$$\mathbf{P}_{\theta_i} = \mathbf{KR}[\mathbf{R}_z(i\theta)|\mathbf{t}] \tag{5}$$

where θ is the known angle between each camera. These simplifications effectively reduce the number of unknowns from $m + 8$ to just the 3 parameters describing the rotation $\mathbf{R}$ since we assume the rotation θ to be known.

2.3 Modifications to H-estimation

In order to estimate $\mathbf{R}$ will we still use the method for estimating $\mathbf{H}$ from [1] but with a few modifications.

Estimation of $\mathbf{H}$ basically comes down to estimating the image fixed entities $\mathbf{l}_s$, the projection of the rotation axis, and $\mathbf{l}_h$, the projection of the plane of rotation. Given these (estimated according to steps 1-4, Section 2.5 in [1]) $\mathbf{H}$ can be determined up to a 3-parameter solution

$$\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3] = [\mathbf{x}_s, \mu\mathbf{x}_a, \nu\mathbf{x}_s + \omega\mathbf{d}] \tag{6}$$

where $\mathbf{d} = \mathbf{l}_s \times (0, 0, 1)^T$ and μ, ν and ω are the unknown parameters. It is possible to estimate μ using the trifocal tensor and ν and ω using parallel scene lines.

Using known internal parameters, $\mathbf{K}$ allows us to simplify the estimation process. According to simplification C3, $\mathbf{H}$ can be written as

$$\mathbf{H} = \mathbf{KR} \tag{7}$$

leading to

$$\mathbf{R} = \mathbf{K}^{-1}\mathbf{H} \ . \tag{8}$$

Since we can not estimate $\mathbf{H}$ without resolving the 3-parameter ambiguity we can not simply estimate $\mathbf{R}$ according to (8). However, we do have good estimates of the first two columns in $\mathbf{H}$ (where the second column is estimated only up to scale), using this we get the following

$$[\mathbf{r}_1, \mu\mathbf{r}_2, \mathbf{0}] = \mathbf{K}^{-1}[\mathbf{h}_1, \mu\mathbf{h}_2, \mathbf{0}] \ . \tag{9}$$

Normalizing $\mathbf{r}_1$ and $\mathbf{r}_2$ allow us to estimate $\mathbf{R}$ as

$$\mathbf{R} = [\hat{\mathbf{r}}_1, \hat{\mathbf{r}}_2, \hat{\mathbf{r}}_1 \times \hat{\mathbf{r}}_2] \tag{10}$$

since we know that we are looking for a rotation matrix $\mathbf{R} \in \mathrm{SO}(3)$.

Due to measurement noise, $\mathbf{r}_1$ and $\mathbf{r}_2$ will not be perfectly orthogonal after (9). In order to remedy this we use a symmetric Gram-Schmidt orthogonalization procedure before computing (10). Ordinary Gram-Schmidt orthogonalization where $\hat{\mathbf{r}}_1$ is held still produces similar results.

One minor (but important) difference to the approach described in the paper is to average $\mathbf{R}$ from estimations between all in view-pairs, this gives considerably better result than just using one view-pair. It is also important to make sure that the averaged $\mathbf{R}$ is still a rotation matrix. This is easily achieved by taking the Singular Value Decomposition (SVD) of the sum of all rotations

$$\mathbf{U}\mathbf{D}\mathbf{V}^T = \sum \mathbf{R}_i \tag{11}$$

and then letting the averaged rotation be

$$\mathbf{R} = \mathbf{U}\mathrm{diag}(1, 1, \det(\mathbf{U})\det(\mathbf{V}))\mathbf{V}^T \ , \tag{12}$$

see [2] for justification.

3 Results

Two figures are shown below in order to visualize the performance of the method, see Fig. 1 and 2. The first figure shows the obtained 3D models when using the turn-table restrictions compared to creating the trajectory of cameras by the unconstrained 'triangle-approach' (introduced in the lectures) . The second figure shows the estimated camera positions for both approaches. The results for the turn-table method were obtained on the given real trajectories (which contain real noise) while the result for the 'triangle-approach' were obtained on noise free data with added Gaussian noise (with a standard deviation of 0.01 pixels). With the 'triangle-approach' on the real trajectories, it completely brakes down.

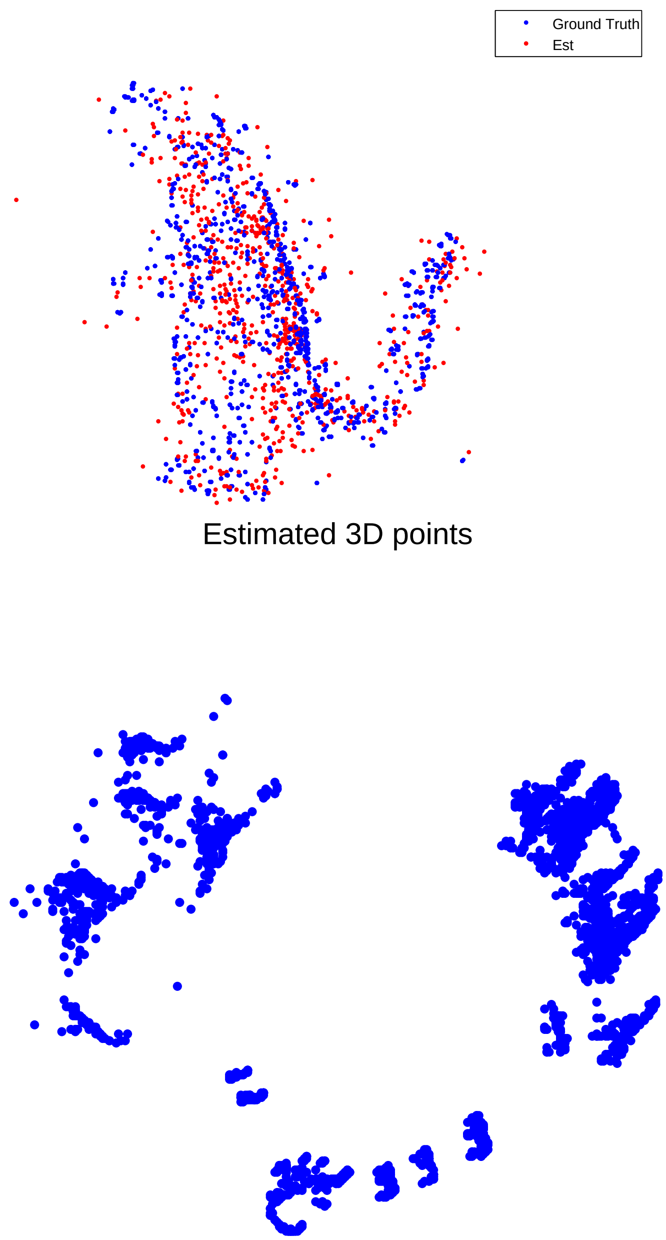

Figure 1: 3D model. Top: This method. Bottom: The unconstrained 'triangle-approach'

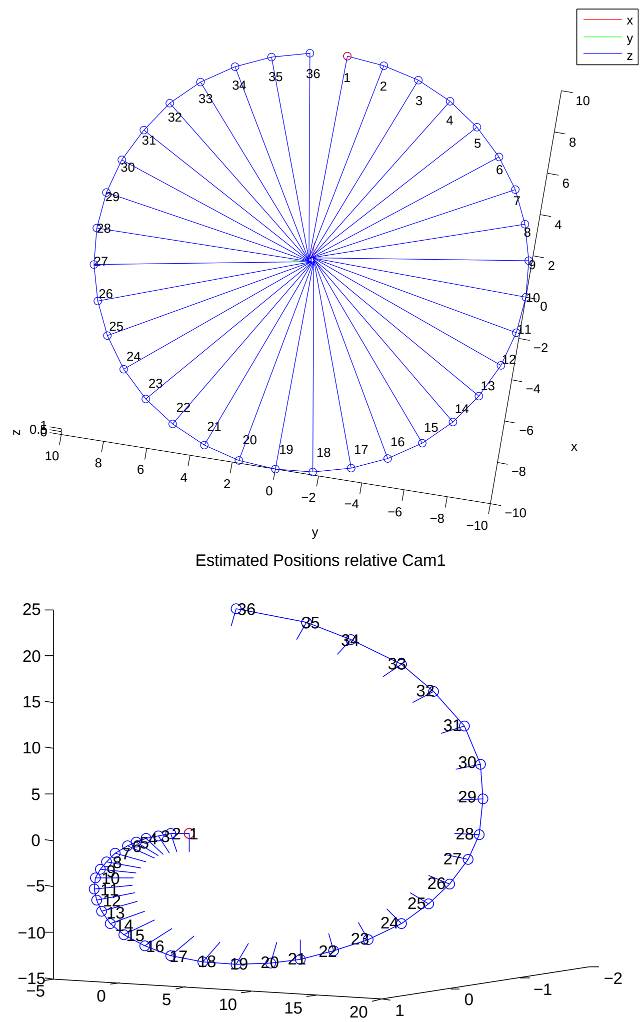

Figure 2: Camera position. Top: This method. Bottom: The unconstrained 'triangle-approach'

4 Conclusions

The results obtained by using further restrictions on the turn-table sequence (known internal parameters and same rotation between each camera pair) but without the BA-step are not too bad, average reprojection error of 1.1 pixels. Feeding these results into a BA-step (not restricted to optimizing over the rotations only but with all parameters free) gives an average reprojection error of 0.01 pixels.

The results obtained without the BA-step are not perfect but it is at least possible to see that the reconstructed shape is not too strange, hence making this modified method suitable for a minor undergraduate project using the paper [1] combined with the modifications described in this paper.

References

[1] Andrew W. Fitzgibbon, Geoff Cross, and Andrew Zisserman. Automatic 3d model construction for turn-table sequences. In R. Koch and L. VanGool, editors, *Proceedings of SMILE Workshop on Structure from Multiple Images in Large Scale Environments*, volume 1506 of *Lecture Notes in Computer Science*, pages 154–170. Springer Verlag, June 1998.

[2] Claus Gramkow. On averaging rotations. *International Journal of Computer Vision*, 42:7–16, 2001.