REPORT

Structure from Motion Pipeline

Peter Gemeiner , Ph.D.
Branislav Micusik , Ph.D.

Safety & Security Department
Video- and Security Technology

22.08.2013

AIT-DSS-0287

VERSIONSNUMMER - 01

Technical Documentation: Structure from Motion Pipeline

Peter Gemeiner

October 1, 2013

Contents

1 Introduction

This technical documentation describes our Structure from Motion pipeline from a system point of view. The idea is to explain in details, which input data is needed and what can the user expect as output. This documentation describes briefly the overall setting, but no internal parameters are mentioned. The algorithms are not mentioned in this document, because they were already described here [6].

This document has five sections. The first section introduces the problem of reconstructing a model of an unknown scene providing images from a single camera. In second section, the input for this reconstruction algorithms is described. The third section presents the output structures and the fourth section presents results from a reconstructed scene. The last section concludes this document.

1.1 Problem Declaration and Motivation

Recovering an unknown scene, also known as Structure from Motion (SfM), is a difficult problem in computer vision. The difficulty is that this task can not be treated as a single task, but has to be tackled as two problems. First, to compute the camera ego-motion and second, to estimate the metric coordinates of the surrounding scene. Fortunately, the computer vision community provides SfM algorithm, which can solve these problems. Our SfM pipeline is using the state-of-the-art algorithms as described in [6]. However, there is future potential for solving some open problems in SfM (e.g. loop closing), but this is out of scope of this document.

The motivation of this document can be considered as twofold. First, as mentioned in [6], SfM is needed to calibrate surveillance cameras in unknown scene (e.g. in the airport). Second, to provide an inexperienced user an idea what input and output data are needed to reconstruct an unknown scene.

1.2 Related Work

The problem of SfM is tackled in the seminal work of Pollefeys et al. [9]. To find an optimal solution the authors presented offline non-linear and linear self-calibration approaches.

In [1], an interesting three-dimensional real-time SfM estimation system is described. This system is based on non-linear filtering. The main disadvantage of this approach is the unavoidable presence of drift, which occurs when long motion is performed.

In robotics community, the same problem as tackled by SfM is called Simultaneous Localisation and Mapping (SLAM). In the same year, when Davison presented Moncular Simultaneous Localisation and Mapping (MonoSLAM) [2], Nister introduced a system capable of robust real-time SfM estimation [7]. Nister's system is based on a preemptive scoring in RANSAC and the framework uses the bundle adjustment [11] algorithm to find robust motion and structure estimates.

Recently, a very interesting real-time parallel tracking and mapping approach was introduced by Klein and Murray in [4]. They proposed splitting the tracking and mapping into two separate tasks, running them in parallel threads. The goal of their approach is to estimate a dominant plane in a small workspace, where virtual characters can be displayed. The result is a system capable of estimating the pose of the camera while mapping thousands of features in real-time.

An excellent overview of current state-of-the-art in SfM is provided in [10].

2 Expected Input

To successfully start our SfM pipeline two different inputs are needed. A sequence of images and a camera model with known intrinsic parameters. Despite the fact that only two inputs are needed we expect several furher assumptions.

As mentioned above, as first input we need a sequence of images. These images has to be taken sequentially, because otherwise the matching could fail establishing point correspodences. Further we assume that the scene contains enough texture and the camera does not perform abrupt motions.

The second expected input is a camera with known internal geometry. This does mean that the camera has to be calibrated before we start taking images [3]. After the images are taken with a perspective camera, these has to be undistorted before starting SfM pipeline. When using a omnidirectional camera, this step is not needed.

This is an overview of above mentioned inputs:

- an image sequence (undistorted when using a perspective camera),
- manually select pairs of images needed for loop closing
- calibration matrix [3] and
- two-parametric non-linear model for fisheye camera [5].

3 Output Structures

The output from SfM pipeline contains:

- detected feature points,
- inlier matches from relative pose computation,
- iterative results from non-linear optimisation and
- various output format files.

3.1 Detected Feature Points

We used four types of features, which provide complementary characteristics. First, FAST features are used to detect corner points [10]. Other two feature detectors are the scale-invariant SIFT and SURF points [10]. These points can be found in many object recognition applications, because they are stable in both location and scale. The last detector is MSER, which is an important affine invariant region detector [10].

For every feature detector a separate folder is created. In this folder, for each processed image frame the result of detection is one structure. This structure has three parts:

- Keypoints. Every keypoint contains: x and y position, size, angle, RGB code and extra information for tracking.
- Descriptors. For every keypoint one 128 dimensional array is assigned [10].
- A structure array with the same name as method for computing descriptors. Each structure contains: the ID of descriptor, keypoint coordinates and descriptor.

3.2 Relative Pose

An important step in our SfM pipeline is to connect consecutive image frames. This is done pairwise, where an Essential matrix is found using the well-known Five-point algorithm [8].

The output relative poses are stored in 'eg' folder. Every relative pose contains:

- indices of inliers,
- array with matched rays,
- essential matrix and
- information about descriptors.

3.3 Sparse Bundle Adjustment

The pose and structure results from non-linear optimisation are stored in 'sfm' folder in *1_c_cposes.mat* file. c is the count of estimated frames. After this file is loaded the user gets two most important varibles. P is a cell array with all projection matrics. X is an array with all three-dimensional points.

A futher interesting information for the user is that the 'sfm' folder contains every iterative output from non-linear optimisation. This provides an overview how the camera poses are added to the P cell array and how the X structure array is expanding.

3.4 Output Formats

For further visualisation or processing purposes is the output structure together with camera poses saved in four formats.

These output formats are:

- Virtual Reality Modeling Language (VRML),
- Polygon File Format (PLY),
- Snavely's Bundler Format (OUT) and
- Extensible Markup Language (XML).

The first two output formats are intended only for visualisation purposes. The saved information is the reconstructed scene and camera's poses. The third output file contains except scene and camera information also the image measurements. This means that for every reconstructed point a track of two-dimensional points is available. The last output file encompases only information about the camera poses.

4 Reconstruction Example

We tested our SfM pipeline in our department to reconstruct the corridor scene. We used a calibrated perspective camera and recorded 319 images. As depicted in Fig. 1, the result of the reconstruction contains 319 camera poses and 54784 points.

5 Conclusion

In this document, we described what are the inputs and outputs of our SfM pipeline. This description is intended to provide a better understanding for future users. With this document the users should be able to process image sequences and extract the usefull information. An example of a reconstructed scene is also enclosed.

6 Acknowledgements

The research leading to these results has received funding from the European Union Seventh Framework Programme under grant agreement no 257906.

References

[1] A. Chiuso, P. Favaro, H. Jin, and S. Soatto. Structure from motion causally integrated over time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):523–535, April 2002.

(Corridor.u3d)

Figure 1: Reconstructed scene of the corridor.

[2] A. J. Davison. Real-time simultaneous localisation and mapping with a single camera. In *IEEE International Conference on Computer Vision*, pages 1403–1410, Nice, France, 2003.

[3] R. Hartley and A. Zisserman. *Multiple View Geometry.* Cambridge University Press, 2003.

[4] G. Klein and D. Murray. Parallel tracking and mapping for small ar workspaces. In *ISMAR 2007 Video Results*, 2007.

[5] B. Mičušík. *Two-View Geometry of Omnidirectional Cameras.* PhD thesis, Prague, Czech Republic, 2004.

[6] B. Mičušík. Calibration of surveillance cameras at vie airport, a proof-of-concept study. Technical Report TR AIT-DSS-0283, AIT Austrian Institute of Technology, Video and Security Technology Unit, Department of Safety and Security, Donau-City-Strasse 1, 1220 Vienna, Austria, 2011.

[7] D. Nistér. Preemptive RANSAC for live structure and motion estimation. In *IEEE International Conference on Computer Vision*, volume 1, pages 199–206, 2003.

[8] D. Nistér. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:756–777, June 2004.

[9] M. Pollefeys, R. Koch, and L. Van Gool. Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. *International Journal of Computer Vision*, 32(1):7–25, 1999.

[10] R. Szeliski. *Computer Vision: Algorithms and Applications*. Springer, 2010.

[11] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment –a modern synthesis, 1999.

Kontakt

AIT Austrian Institute of Technology GmbH
Donau-City-Strasse 1, 1220 Vienna, Austria

www.ait.ac.at
Fax +43 50550- 4150

Peter Gemeiner
Scientist
Safety & Security Department
Video- and Security Technology
+43 50550-4140
peter.gemeiner@ait.ac.at

Branislav Micusik
Senior scientist
Safety & Security Department
Video- and Security Technology
+43 50550-4292
branislav.micusik@ait.ac.at