Theia: A Fast and Scalable Structure-from-Motion Library

Chris Sweeney Tobias Höllerer Matthew Turk
University of California Santa Barbara
2104 Harold Frank Hall
Santa Barbara, USA
{cmsweeney, holl, mturk}@cs.ucsb.edu

ABSTRACT

The Theia library provides students, researchers, and industry experts with a clean C++ library including a state-of-the-art Structure-from-Motion pipeline and a vast collection of multi-view geometry tools and algorithms that utilize image and video inputs to create high quality 3D reconstructions. The library is BSD licensed with strict Google C++ style guide adherence and comprehensive unit test coverage. All algorithms are intentionally designed to be scalable, and multithreaded computation is utilized automatically whenever possible. Theia is very modular so that all algorithms can be easily extended, modified, or used independently of the rest of the library. Feature extraction, image matching, RANSAC, pose estimation and SfM methods may all be chosen at runtime, enabling simple experimentation and fine-tuning for obtaining high quality SfM reconstructions. Since being released in February 2015 Theia has gathered an active community of users spanning graduate students, industry members, and computer vision experts.

Categories and Subject Descriptors

I.4.5 [**Computer Vision**]: Reconstruction; I.4.9 [**Computer Vision**]: Applications—*structure-from-motion*

Keywords

Computer vision, Structure-from-Motion, multi-view geometry

1. INTRODUCTION

The proliferation of online photo collections, such as Google Street View and Flickr, has allowed for large-scale mapping of urban environments via structure from motion (SfM)[15]. In particular, famous landmarks around the world have been captured by thousands or even millions of images and can be densely reconstructed using SfM. The 3D reconstructions obtained from SfM can be useful for an array of applications such as autonomous navigation, virtual tourism, and even disaster relief. While the benefits of SfM are clear, few publicly available tools exist for creating 3D reconstructions from SfM, and none of the currently available software is specifically geared towards large-scale SfM.

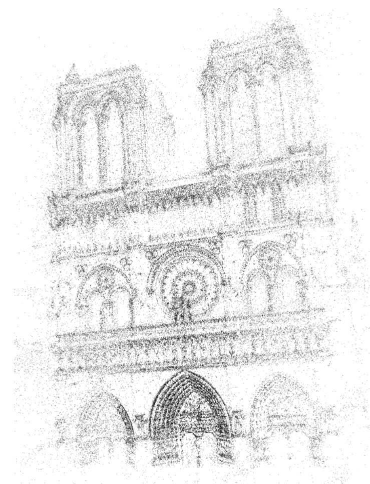

Figure 1: We are able to reconstruct Notre Dame from over 500 images in less than 120 seconds using Theia, compared to over 11 minutes using alternative software such as VisualSfM.

In this paper we introduce Theia, a fast and scalable SfM library designed to be simple to understand and easy to extend. Theia is intended to provide computer vision researchers and academics with a set of multi-view geometry tools that are useful as individual components while also providing full out-of-the-box SfM pipelines that are efficient and robust. The library utilizes the SSE optimized Eigen[1] library for matrix operations and Ceres Solver[2] for large-scale, multi-threaded nonlinear optimization. The code is extensively covered by unit tests and features significant, useful logging tools for performance evaluation and debugging. CMake[3] is used to ensure cross-platform portability and a simple build and installation process. Theia is well documented within the code and on the library's website[4]. To make this library as useful as possible for other researchers, we have also provided compatible I/O interfaces with match-

[1] http://eigen.tuxfamily.org
[2] http://ceres-solver.org
[3] http://www.cmake.org
[4] http://www.theia-sfm.org

MM'15, October 26–30, 2015, Brisbane, Australia.
© 2015 ACM. ISBN 978-1-4503-3459-4/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2733373.2807405.

Table 1: **Our RANSAC class is comprised of a `Sampler` class and a `QualityMeasurement` class that allow for alternative RANSAC approaches to be easily implemented. The table below outlines the RANSAC variants implemented in Theia by combining different sampling and quality measurement strategies.**

	Random Sampler	Progressive Sampler	EVT Sampler	SPRT Sampler
Inlier	RANSAC [5]	PROSAC [4]	EVSAC [6]	-
MLE	MLESAC [20]	MLE + PROSAC	MLE + EVSAC	ARRSAC [14]

ing and reconstruction files computed by Bundler[5] and VisualSFM[6] as well as interfaces to several benchmark datasets.

1.1 Comparison to other software

There are several open-source SfM libraries available, but they differ from Theia in some key features. The Bundler library robustly computes 3D reconstructions with incremental SfM, limiting the ability to scale (see Section 3 for a comparison of incremental and global SfM). VisualSfM is a closed-source software that contains an incremental SfM pipeline that utilizes multi-threading and GPU programming for high efficiency; however, the scalability is still limited because of the incremental nature of the pipeline. OpenMVG[7] is a SfM library with an active community that implements both an incremental SfM and global SfM pipeline; however, the focus of OpenMVG is on high accuracy for small to medium-sized problems and many of the methods were not designed with scalability in mind. Further, OpenMVG does not adhere to a consistent style guide and is not particularly modular, making it more difficult to extend and develop for casual users. The primary goals of the Theia library are usability, extendibility, and scalability.

2. OVERVIEW OF FEATURES

In this section we provide an overview of the core algorithms of Theia. The core features are modular so although they contribute to Theia's SfM pipeline, they may still be used independently. As such, the interface to the core features is typically generic and not dependent on Theia-specific data types. The library contains a large number of useful algorithms for multi-view geometry, linear algebra, and optimization.

2.1 Feature Extraction

Detecting salient image points is a fundamental aspect of computer vision. Feature detection and extraction methods that produce distinctive, repeatable image points and descriptors are desired for a wide range of applications such as object detection, image recognition, and multi-view stereo. In Theia, we implement a generic keypoint detector and feature descriptor extraction interface so that various types of image feature methods may be implemented and seamlessly integrated into the library. Further, this abstract interface allows the user to select the desired feature extraction method at run-time. The library currently contains implementations for SIFT, BRIEF, BRISK, FREAK, and AGAST features with support for AKAZE in development.

2.2 Feature Matching

To determine which images observe similar views of a scene, features are matched across images. There are currently two methods for feature matching in Theia: brute forces and a cascade hashing [3] that is over two orders of magnitude faster than brute force matching. Feature matching is also built with a generic interface so that new matching techniques may be seamlessly added and integrated into the library. The abstract matching interface utilizes dynamic thread-pooling to optimize for multi-threaded performance, allowing Theia users to implement new matching techniques while getting the multi-threaded performance for free.

2.3 RANSAC

Random sample and consensus, or RANSAC, is one of the most commonly used algorithms in Computer Vision. As a result, much research has gone into making RANSAC extensions and variants that increase the efficiency or accuracy of the estimation. We have implemented a templated class that makes using RANSAC for estimation extremely easy as well as simple to extend. The user defines an `Estimator` class that estimates a model from a set of data. This allows the user to easily deploy any RANSAC class for a variety of tasks without having to rewrite the RANSAC-specific code.

Further, the RANSAC class itself is composed of an abstract `Sampler` class that samples the data and a `QualityMeasurement` class that determines how a model fits the data. For standard RANSAC, the `Sampler` class performs random sampling and the `QualityMeasurement` class counts the number of inliers in the data. These classes can be used to implement different RANSAC methods. For instance, using a maximum likelihood error as the `QualityMeasurement` would result in MLESAC[20] as shown in Table 1. RANSAC [5], PROSAC [4], MLESAC [20], ARRSAC [14], and EVSAC [6] have been implemented with this generic RANSAC interface.

2.4 Pose Estimation

A fundamental problem in multi-view geometry is the ability to determine a camera's pose in a scene. This library implements numerous state-of-the-art pose estimation methods with generic interfaces so that they may be used independently of Theia's SfM pipeline. In addition to tcomprisedhe standard absolute and relative pose problems, we also implement methods for multi-camera systems, partially calibrated cameras, and scenarios where IMU information is available (*e.g.*, the vertical direction is known from IMU sensors). The following state-of-the-art solvers are currently implemented:

- Absolute pose: P3P [10], PnP [9]
- 4-point algorithm for absolute pose and focal length [1]
- 5-point algorithm for absolute pose, focal length, and radial distortion parameter(s) [11]
- Relative pose: 5-point essential matrix [16], 7-point fundamental matrix [7], 8-point fundamental matrix [8]

[5]`http://www.cs.cornell.edu/~snavely/bundler`
[6]`http://ccwu.me/vsfm`
[7]`http://openmvg.readthedocs.org`

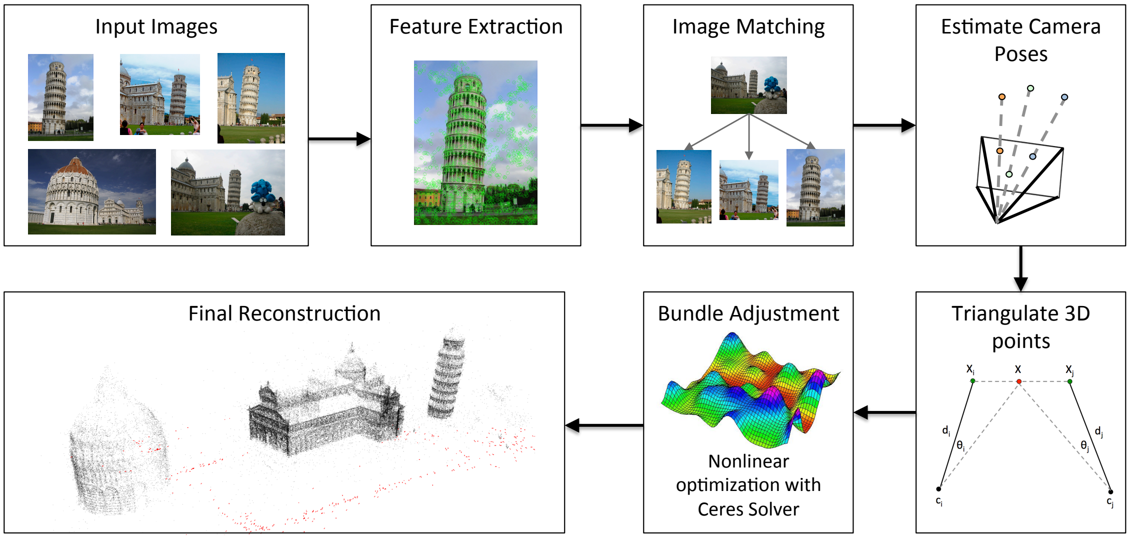

Figure 2: Overview of the global SfM pipeline implemented in Theia. In contrast to incremental SfM, each step in a global SfM pipeline is fully parallelizable and thus extremely scalable. Each step in Theia's SfM pipeline is modular and can be modified at runtime *e.g.*, to use different feature descriptors or pose estimation algorithms.

- Relative pose with known vertical direction: 3-point relative pose [17], 4-point for multi-camera systems [17]
- Homography from 4 points [7]
- Similarity transformations: from 3D-3D correspondences [21], from 2D-3D correspondences [18], from 2D-2D correspondences [19]

2.5 Mathematics and Optimization

We make a number of useful mathematics tools available as part of the library:

- A scalable L_1 minimizer
- Polynomial solvers (closed form and iterative)
- Sparse matrix eigen-decomposition
- RQ matrix decomposition
- Bundle Adjustment

3. SFM PIPELINE

At the core of Theia is the SfM module. Theia contains Incremental and Globabl SfM pipelines, but we only describe the Global SfM pipeline here (*c.f.* Figure 2). By combining the modular features presented in Section 2, we create SfM pipelines that are simple to follow, easily extendable, and highly scalable. The Incremental pipeline follows a standard sequential SfM procedure [15]. The Global SfM pipeline takes a set of pairwise relative poses between cameras as input, and outputs the orientation and position of all cameras in a global reference frame. The camera poses are computed through motion averaging algorithms. These global methods are inherently parallelizable and only require a single bundle adjustment, which is generally the most expensive part of SfM. This is in contrast to incremental SfM methods that add one new image at a time repeatedly perform bundle adjustment, making them slower and less scalable than global SfM methods. Our global SfM pipeline is summarized with the following steps: in Ojai

1. **Feature Extraction:** We extract feature descriptors (in parallel) at salient points within images. The feature type may be chosen at run-time for convenience.
2. **Image Matching:** After features are extracted, images must be matched to determine two-view geometry between images that observe the same scene. By default, Theia uses the extremely fast cascade hashing method [3] to compute image matches with multiple threads, though the matching technique may also be chosen at run-time.
3. **Estimate Camera Poses:** We use the geometrically verified two-view matches from the previous step to estimate camera poses with global motion averaging schemes. Camera orientations are estimated with either a robust orientation estimation algorithm [2] or with an L_2 averaging scheme [12]. After camera orientations are estimated, the camera positions are robustly estimated with a nonlinear position optimization [22]. The robust position estimation method of [13] is currently in development and will soon be integrated into the library.
4. **Triangulate 3D Points:** After camera poses are estimated, 3D points are triangulated in parallel and refined with a nonlinear optimization.
5. **Bundle Adjustment:** As a final step, camera poses and 3D points are refined with a nonlinear optimization to minimize reprojection error. We use the Ceres Solver for scalable multi-threaded optimization to ensure high quality results are obtained efficiently.

Table 2: Efficiency evaluation (in seconds) of Theia vs VisualSfM for large reconstructions using 8 threads. The number of images is given in parentheses for each dataset.

	VisualSfM	Theia
Notre Dame (553)	687	118
Pisa (481)	621	142
Trevi (1259)	2467	387

3.1 ReconstructionBuilder

The simplest way to create a 3D reconstruction with Theia is to utilize the `ReconstructionBuilder` class. This class takes images as input and outputs 3D reconstructions created from the input images. The caller may choose to use Incremental or Global SfM at runtime. The options set in the `ReconstructionBuilder` control parameters for feature extraction, matching, pose estimation, triangulation, and bundle adjustment. Creating a reconstruction can be done in just a few lines of code:

```
ReconstructionBuilder builder(options);
for (const std::string& image : image_files)
    builder.AddImage(image);

std::vector<Reconstruction*> reconstructions;
builder.BuildReconstruction(&reconstructions);
```

3.2 Performance

Theia achieves state-of-the-art performance on large scale datasets both in terms of efficiency and accuracy. Timing results for several large scale datasets are shown in Table 2, and more results are available on the Theia website.

4. CONCLUSIONS

In this paper, we have presented a comprehensive multi-view geometry library, Theia, that focuses on large-scale SfM. In addition to state-of-the-art scalable SfM pipelines, the library provides numerous tools that are useful for students, researchers, and industry experts in the field of multi-view geometry. Theia contains clean code that is well documented (with code comments and the website) and easy to extend. The modular design allows for users to easily implement and experiment with new algorithms within our current pipeline without having to implement a full end-to-end SfM pipeline themselves. Theia has already gathered a large number of diverse users from universities, startups, and industry and we hope to continue to gather users and active contributors from the open-source community.

5. ACKNOWLEDGMENTS

This work was supported in part by NSF Grant IIS-1219261, ONR Grant N00014-14-1-0133, and NSF Graduate Research Fellowship Grant DGE-1144085

6. REFERENCES

[1] M. Bujnak, Z. Kukelova, and T. Pajdla. A general solution to the p4p problem for camera with unknown focal length. In *CVPR*. IEEE, 2008.

[2] A. Chatterjee and V. M. Govindu. Efficient and robust large-scale rotation averaging. In *ICCV*. IEEE, 2013.

[3] J. Cheng, C. Leng, J. Wu, H. Cui, and H. Lu. Fast and accurate image matching with cascade hashing for 3d reconstruction. In *CVPR*. IEEE, 2014.

[4] O. Chum and J. Matas. Matching with PROSAC-progressive sample consensus. In *CVPR*, 2005.

[5] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981.

[6] V. Fragoso, P. Sen, S. Rodriguez, and M. Turk. EVSAC: Accelerating Hypotheses Generation by Modeling Matching Scores with Extreme Value Theory. In *ICCV*. IEEE, 2013.

[7] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[8] R. I. Hartley. In defense of the eight-point algorithm. *T-PAMI*, 1997.

[9] J. Hesch and S. Roumeliotis. A direct least-squares (dls) solution for pnp. In *ICCV*. IEEE, 2011.

[10] L. Kneip, D. Scaramuzza, and R. Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In *CVPR*. IEEE, 2011.

[11] Z. Kukelova, M. Bujnak, and T. Pajdla. Real-time solution to the absolute pose problem with unknown radial distortion and focal length. In *ICCV*. IEEE, 2013.

[12] D. Martinec and T. Pajdla. Robust rotation and translation estimation in multiview reconstruction. In *CVPR*. IEEE, 2007.

[13] O. Ozyesil and A. Singer. Robust camera location estimation by convex programming. In *CVPR*, 2015.

[14] R. Raguram, J.-M. Frahm, and M. Pollefeys. A comparative analysis of ransac techniques leading to adaptive real-time random sample consensus. In *ECCV*. Springer, 2008.

[15] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM TOG*. ACM, 2006.

[16] H. Stewenius, C. Engels, and D. Nistér. Recent developments on direct relative orientation. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2006.

[17] C. Sweeney, J. Flynn, and M. Turk. Solving for relative pose with a partially known rotation is a quadratic eigenvalue problem. In *3DV*. IEEE, 2014.

[18] C. Sweeney, V. Fragoso, T. Hollerer, and M. Turk. gdls: A scalable solution to the generalized pose and scale problem. In *ECCV*. Springer, 2014.

[19] C. Sweeney, L. Kneip, T. Hollerer, and M. Turk. Computing similarity transformations from only image correspondences. In *Proceedings of CVPR*, 2015.

[20] P. H. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 2000.

[21] S. Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE T-PAMI*, 1991.

[22] K. Wilson and N. Snavely. Robust global translations with 1dsfm. In *ECCV*. Springer, 2014.