Acta Numerica (2017), pp. 305–364
doi:10.1017/S096249291700006X
© Cambridge University Press, 2017
Printed in the United Kingdom

A survey of structure from motion

Onur Özyeşil
INTECH Investment Management LLC,
One Palmer Square, Suite 441,
Princeton, NJ 08542, USA
E-mail: oozyesil@intechjanus.com

Vladislav Voroninski
Helm.ai, Menlo Park,
CA 94025, USA
E-mail: vlad@helm.ai

Ronen Basri
Department of Computer Science and Applied Mathematics,
Weizmann Institute of Science,
Rehovot, 76100, Israel
E-mail: ronen.basri@weizmann.ac.il

Amit Singer
Department of Mathematics and PACM,
Princeton University,
Princeton, NJ 08544-1000, USA
E-mail: amits@math.princeton.edu

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

The structure from motion (SfM) problem in computer vision is to recover the three-dimensional (3D) structure of a stationary scene from a set of projective measurements, represented as a collection of two-dimensional (2D) images, via estimation of motion of the cameras corresponding to these images. In essence, SfM involves the three main stages of (i) extracting features in images (*e.g.* points of interest, lines, *etc.*) and matching these features between images, (ii) camera motion estimation (*e.g.* using relative pairwise camera positions estimated from the extracted features), and (iii) recovery of the 3D structure using the estimated motion and features (*e.g.* by minimizing the so-called *reprojection error*). This survey mainly focuses on relatively recent developments in the literature pertaining to stages (ii) and (iii). More specifically, after touching upon the early factorization-based techniques for motion and structure estimation, we provide a detailed account of some of the recent camera *location* estimation methods in the literature, followed by discussion of notable techniques for 3D structure recovery. We also cover the basics of the *simultaneous localization and mapping* (SLAM) problem, which can be viewed as a specific case of the SfM problem. Further, our survey includes a review of the fundamentals of feature extraction and matching (*i.e.* stage (i) above), various recent methods for handling ambiguities in 3D scenes, SfM techniques involving relatively uncommon camera models and image features, and popular sources of data and SfM software.

CONTENTS

1 Introduction 306
2 Early works 309
3 Camera location estimation 313
4 Structure estimation 328
5 Simultaneous localization and mapping (SLAM) 338
6 Other topics 342
7 Conclusion 357
References 359

1. Introduction

Recovering the three-dimensional structure of a scene from images is a fundamental goal of computer vision. A particularly effective approach to three-dimensional (3D) reconstruction involves the use of many images of a stationary scene. This problem, commonly referred to as *multiview structure from motion* (multiview SfM) (depicted in Figure 1.1), is the subject of a large body of research in computer vision, starting with the seminal paper of Longuet-Higgins (1981). A comprehensive and in-depth summary of this enormous body of work can be found in Hartley and Zisserman (2000).

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

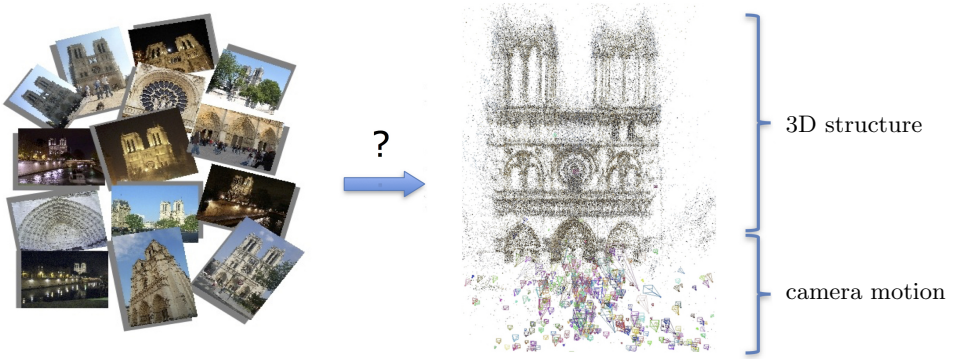


Figure 1.1. The structure from motion (SfM) problem.

Modern methods usually solve the multiview SfM problem via *bundle adjustment* techniques, which aim to optimize a cost function known as the *total reprojection error*; see Section 4 for a more detailed and general discussion of bundle adjustment. With this cost function, given n images of a stationary scene, the objective is to simultaneously determine the structure (3D coordinates of scene points) and the calibration parameters of each of the n cameras that minimize the discrepancy between image measurements and their *predictive model*. For instance, in the specific case of pairwise point correspondences between images, we let $\{\mathbf{P}_j\}_{j=1}^{p} \subseteq \mathbb{R}^3$ denote the unknown positions of scene points, and let $\{C_i\}_{i=1}^{n}$ be the 3×4 camera matrices, where the entries of each C_i encode the location of camera i, its orientation and internal calibration parameters. Also, let $(x_{ij}, y_{ij}) \in \mathbb{R}^2$ denote the measured projection of point $\mathbf{P}_j$ onto the image plane of camera C_i. Then we can define a (relatively simple) bundle adjustment problem based on minimizing the sum of squares cost function

$$\underset{\{\mathbf{P}_j\},\{C_i\}}{\text{minimize}} \quad \sum_{i \sim j} \left(x_{ij} - \frac{C_{i1}^T \mathbf{P}_j}{C_{i3}^T \mathbf{P}_j} \right)^2 + \left(y_{ij} - \frac{C_{i2}^T \mathbf{P}_j}{C_{i3}^T \mathbf{P}_j} \right)^2, \tag{1.1}$$

where $C_{ik} \in \mathbb{R}^4$ denotes the kth row of C_i $(1 \leq k \leq 3)$, $i \sim j$ means that the jth scene point is visible by the ith camera, and we abuse notation and identify the representation $[\mathbf{P}_j, 1] \in \mathbb{R}^4$ of $\mathbf{P}_j$ in homogeneous coordinates by $\mathbf{P}_j$. Unfortunately, as a result of the special structure of the camera matrices and the cost function in (1.1), the bundle adjustment problem (1.1) is not convex, and its (naïve) optimization in realistic settings typically converges to an often undesired local minimum. It is therefore critical to develop methods that can initialize bundle adjustment, that is, provide initial camera motion (and intrinsic calibration) and 3D structure estimates, as closely as possible to the true solution.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

Snavely, Seitz and Szeliski (2006) presented a sequential pipeline for SfM, demonstrating that it can produce accurate reconstructions in practical scenarios where hundreds or even thousands of independently captured photographs are provided, and sparked huge interest in the development of efficient SfM techniques for large unordered image sets (see Section 4). The suggested pipeline begins by detecting keypoints in each image. It then uses the SIFT descriptor (Lowe 2004) to compare those keypoints across images and to produce a set of potential matches. Random sampling and consensus (RANSAC: Bolles and Fischler 1981) is then used to robustly estimate essential matrices between pairs of images (for computation of relative motion of camera pairs) and to discard outlier matches. Then, starting with a pair of images for which the largest number of inlier matches were found and then greedily adding one image at a time, bundle adjustment is solved repeatedly. Although this sequential pipeline is computationally challenging, it successfully deals with large collections of images, in many cases producing highly accurate reconstructions. However, it is based on greedy steps that may not result in an optimal solution. Clearly, global approaches, which consider all images simultaneously (at least for initial camera motion estimation), may potentially yield improved solutions. Indeed, a number of recent methods attempt to estimate camera locations and orientations globally. In this survey, for the initial camera motion estimation part of SfM, we focus on methods for camera location estimation in Section 3, and refer the reader to other surveys that review the problems of camera orientation and calibration recovery (Tron, Zhou and Daniilidis 2016).

The outline of this survey is as follows. In Section 2 we briefly discuss some of the early works in the literature that introduced the basic SfM framework and the relatively simple factorization methods. Section 3 covers some of the recent work on camera location estimation from pairwise keypoint matches, such as the simpler least-squares methods based on the (homogeneous) epipolar constraints, relatively stable convex methods based on a linearized representation (see (3.8)) of pairwise direction measurements and heuristic methods for outlier rejection in pairwise directions. In Section 4 we provide some of the key ideas and methods in the literature for 3D structure estimation, with an emphasis on contemporary methods aimed at processing large unordered sets of images. In Section 5 we discuss details of influential works on the *simultaneous localization and mapping* (SLAM) problem, which has recently experienced a dramatic increase in popularity. Section 6 considers some of the remaining crucial topics in the vast SfM literature, including methods for feature point extraction and matching, works on handling instabilities induced by symmetries and ambiguities in the images, methods for relatively uncommon types of cameras (*e.g.* omnidirectional cameras), techniques using alternative basic measurements (*e.g.* based on lines instead of feature points), and some of the important software packages and popular

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

datasets. We conclude in Section 7 with a discussion of the current state of the art and possible future directions for the SfM problem.

Rather unfortunately, as is the case for any survey of a topic for which there is an extensive body of literature, we are unable to cover every important work in the field. In essence, we try to focus on relatively recent SfM techniques. For earlier works in the literature, we refer the reader to other sources, for example Triggs, McLauchlan, Hartley and Fitzgibbon (2000), Hartley and Zisserman (2000) and Oliensis (2000).

2. Early works

This section covers some of the relatively early works on SfM, which introduced the basic problem formalism and the factorization-based methods. We first consider the seminal work by Longuet-Higgins (1981), who introduced the first linear method based on point correspondences, later named the *eight-point algorithm*, to solve the SfM problem for a pair of cameras. Specifically, for a pair of cameras, Longuet-Higgins (1981) aimed to estimate the relative camera motion, that is, relative rotation and translation, and the 3D coordinates of the scene points captured by these cameras. Here we represent the ith camera by $(R_i, \mathbf{t}_i)$, where $R_i \in \mathrm{SO}(3)$ is the camera's orientation matrix and $\mathbf{t}_i \in \mathbb{R}^3$ is its position. Considering that we can always fix the (intrinsic) coordinate system of one of the two cameras to be the global coordinate system (*i.e.* we can set $R_i = I$ and $\mathbf{t}_i = \mathbf{0}$, since an absolute coordinate system cannot be determined from point correspondences), solving for the relative motion for a camera pair, and for the scene points based on the computed motion, then corresponds to actually solving the SfM problem for the pair. The problem setup[1] involves a simple pinhole camera model, in which a scene point $\mathbf{P} \in \mathbb{R}^3$ is represented in the ith image plane by $\mathbf{p}_i \in \mathbb{R}^3$, as shown in Figure 2.1; see also Özyeşil *et al.* (2015, Section 4) and Hartley and Zisserman (2000), for example. Here, we obtain $\mathbf{p}_i$ by first representing $\mathbf{P}$, in the ith camera's coordinate system, as $\mathbf{P}_i = R_i^T(\mathbf{P} - \mathbf{t}_i) = (\mathbf{P}_i^x, \mathbf{P}_i^y, \mathbf{P}_i^z)^T$, and then projecting it onto the ith image plane via $\mathbf{p}_i = (f_i/P_i^z)\mathbf{P}_i$, where $R_i \in \mathrm{SO}(3)$ denotes the orientation matrix, $\mathbf{t}_i \in \mathbb{R}^3$ denotes the location of the focal point, and $f_i \in \mathbb{R}^+$ is the

[1] Our notation and type of projective measurement (or equivalently the global coordinate system) are different from those of Longuet-Higgins (1981), even though the two choices are equivalent, that is, any result obtained by using one can be represented in terms of the other. Our choices (more or less) reflect the common terminology in the literature: see *e.g.* Özyeşil, Singer and Basri (2015), Özyeşil and Singer (2015) and Arie-Nachimson, Kovalsky, Kemelmacher-Shlizerman, Singer and Basri (2012). We also represent the epipolar constraints in their general form for a pair (i, j), to make use of them in the following sections.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

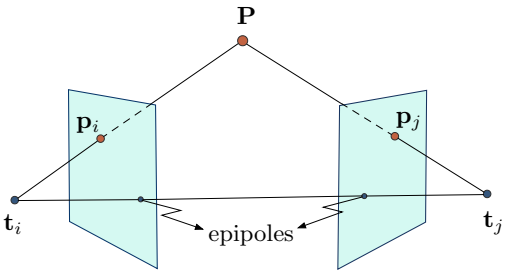


Figure 2.1. The pinhole camera model.

focal length of the ith camera. For a pair of cameras i and j as in Figure 2.1, we can restate the coplanarity of the points $\mathbf{P}, \mathbf{t}_i$ and $\mathbf{t}_j$, that is, $[(\mathbf{P}-\mathbf{t}_i)\times(\mathbf{P}-\mathbf{t}_j)]^T(\mathbf{t}_i-\mathbf{t}_j)=0$, in terms of the observable corresponding point measurements $\mathbf{p}_i, \mathbf{p}_j$ (see Figure 3.1 below for a depiction of a set of corresponding points between a pair of images) and the camera parameters as

$$\mathbf{p}_i^T([R_i^T(\mathbf{t}_j-\mathbf{t}_i)]_\times R_i^T R_j)\mathbf{p}_j =: \mathbf{p}_i^T E_{ij}\mathbf{p}_j = 0, \tag{2.1}$$

where $[\mathbf{t}]_\times$ is the skew-symmetric matrix corresponding to the cross-product with $\mathbf{t}$, and $E_{ij}=[R_i^T(\mathbf{t}_j-\mathbf{t}_i)]_\times R_i^T R_j$ is the *essential matrix* for the cameras i and j. Also, let $T_{ij} := [R_i^T(\mathbf{t}_j-\mathbf{t}_i)]_\times$ and $R_{ij} := R_i^T R_j$, yielding $E_{ij}=T_{ij}R_{ij}$. Longuet-Higgins (1981) made the crucial observation that, among the various equivalent restatements of the coplanarity of $\mathbf{P}, \mathbf{t}_i$ and $\mathbf{t}_j$, the useful property of (2.1), known as the *epipolar constraint*, is that it provides a basis for the estimation of the structured essential matrices in terms of the observable corresponding points. In other words, by fixing the undetermined scale for the entries of E_{ij} (*e.g.* the Frobenius norm $\|E_{ij}\|_F = 1$, or as in Longuet-Higgins (1981) $\|\mathbf{t}_j-\mathbf{t}_i\|=1$, which is equivalent to $\|E_{ij}\|_F=\sqrt{2}$), we can solve for $E_{ij}\in\mathbb{R}^{3\times 3}$ from eight linearly independent epipolar constraints (2.1) corresponding to eight 3D points.[2] Longuet-Higgins (1981) did not provide any algorithm for this purpose, nor did he consider the effects of uncertainties (*i.e.* noise) in the corresponding point measurements; however, the usual approach in the literature is simply to minimize the sum of squared errors in the epipolar constraints subject to the scale constraint, which is equivalent to finding the singular vector corresponding to the smallest singular value of the resulting data matrix. After estimating E_{ij} (up to a sign to be determined later), Longuet-Higgins (1981) solved for T_{ij} (up to another sign, again to be determined later) by

[2] In fact, due to the special structure of E_{ij}, it is well known (see *e.g.* Hartley and Zisserman 2000) that only five corresponding 3D points are sufficient to estimate E_{ij}. However, this requires us to solve a non-linear system of equations.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

eliminating R_{ij} using $E_{ij}E_{ij}^T = T_{ij}T_{ij}^T$. He then solved for R_{ij}, based on its orthogonality, and for the scene points using simple algebraic equations (see Longuet-Higgins 1981 for details). Lastly, the undetermined signs of E_{ij} and T_{ij} are fixed by requiring the scene points to lie in front of both cameras. If the scene points are behind both cameras, the sign of T_{ij} is altered, and the relative rotation R_{ij} and the scene points are recalculated. If the scene points fall behind one of the two cameras, the sign of E_{ij} is altered, and T_{ij}, R_{ij} and the scene points are recalculated. Further, Longuet-Higgins (1981) briefly discussed some of the degenerate cases of eight 3D points for which the eight-point estimation cannot be used, for example the cases of all points corresponding to the vertices of a cube, at least seven points lying on a plane, six of the eight points corresponding to the vertices of a regular hexagon, or at least four points lying on a line. We note that although Longuet-Higgins (1981) argued for the accuracy of the eight-point algorithm, it was in fact observed to be sensitive to noise, which resulted in the construction of alternative methods, such as a normalized version of the eight-point algorithm (Hartley 1997), for relative pairwise motion estimation. For further reading on alternative methods to obtain epipolar geometries between images, see for example Zhang (1998) and Hartley and Zisserman (2000).

Another early work in the literature is that of Tomasi and Kanade (1992), who introduced the *factorization method* for SfM. To model the SfM problem for objects that are relatively distant compared to their sizes, Tomasi and Kanade (1992) assumed an *orthographic* camera model, in which the 3D points are measured via parallel projections onto the image plane, consequently ignoring the camera translation along the optical axis. Let the location of the jth 3D point on the jth camera plane be given by $\mathbf{q}_{ij} = (x_{ij}, y_{ij}) \in \mathbb{R}^2$. In the noiseless case, these orthographic measurements of m 3D points by n cameras are represented in terms of a *measurement matrix* $W \in \mathbb{R}^{2n\times m}$ satisfying

$$W = \begin{bmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nm} \\ y_{11} & \cdots & y_{1m} \\ \vdots & \ddots & \vdots \\ y_{n1} & \cdots & y_{nm} \end{bmatrix}. \tag{2.2}$$

The crucial observation made by Tomasi and Kanade (1992) is that if the origin of the global reference system is chosen to be at the centre of the 3D structure points (implying that each row of W sums to zero), then W can

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

be decomposed into

$$W = RS, \quad \text{for } R = \begin{bmatrix} \mathbf{u}_1^T \\ \vdots \\ \mathbf{u}_n^T \\ \mathbf{v}_1^T \\ \vdots \\ \mathbf{v}_n^T \end{bmatrix} \in \mathbb{R}^{2n\times 3} \quad \text{and} \quad S = [\mathbf{P}_1 \dots \mathbf{P}_m] \in \mathbb{R}^{3\times m}, \tag{2.3}$$

where $\mathbf{u}_k$ and $\mathbf{v}_k$ represent the orientation of the kth camera (since, in the orthographic model, the third direction given by $\mathbf{u}_k \times \mathbf{v}_k$ is parallel to the viewing direction of the cameras) and $\mathbf{P}_k$ is the kth 3D point. As a result, W has rank 3. In the noisy case, which may result in a higher rank for W, Tomasi and Kanade (1992) considered a valid measurement matrix to be given by the best rank-3 approximation to the noisy W in the Frobenius norm sense, which is given by the singular value decomposition (SVD). Note that replacing R and S with RQ and $Q^{-1}S$, for an arbitrary invertible matrix $Q \in \mathbb{R}^{3\times 3}$, results in the same measurement matrix W. As a result, in order to compute R and S from W, Tomasi and Kanade (1992) proposed first computing a decomposition of W using SVD, and then solving for Q using the orthonormality of the n pairs $(\mathbf{u}_k, \mathbf{v}_k)$. That is, for $W = U_W \Sigma_W V_W^T$ representing the SVD of W, they first let $\hat{R} = U_W \Sigma^{1/2}$ and $\hat{S} = \Sigma^{1/2} V_W^T$, and then computed the solution $R = \hat{R}Q$ and $S = Q^{-1}\hat{S}$, where Q is given to be a solution to the non-linear system of $3n$ equations

$$\hat{\mathbf{u}}_k^T Q Q^T \hat{\mathbf{u}}_k = \hat{\mathbf{v}}_k^T Q Q^T \hat{\mathbf{v}}_k = 1, \quad \hat{\mathbf{u}}_k^T Q Q^T \hat{\mathbf{v}}_k = 0, \quad k = 1, \dots, n, \tag{2.4}$$

where $\hat{\mathbf{u}}_k$ and $\hat{\mathbf{v}}_k$ represent the kth and $(k+n)$th rows of $\hat{R}$ (see (2.3)). They also extended the factorization method to the case of *occlusions* (*i.e.* to the case when some scene points are not visible by all of the cameras) and provided empirical evidence demonstrating the stability of their factorization approach on various experimental scenarios.

Sturm and Triggs introduced an extension of the factorization methodology for the multiview SfM with perspective cameras. They considered the basic image projection equations

$$\lambda_{ij} \mathbf{p}_{ij} = C_i \mathbf{P}_j, \quad i = 1, \dots, n, \quad j = 1, \dots, m, \tag{2.5}$$

where, for the jth 3D point represented (by abuse of notation) in the homogeneous coordinates by $\mathbf{P}_j \in \mathbb{R}^4$ and C_i denoting the ith camera matrix, $\mathbf{p}_{ij} \in \mathbb{R}^3$ is the projection of the jth point onto the ith camera plane (in the homogeneous coordinates) and λ_{ij} are the undetermined scales, called *projective depths*. Similarly to Tomasi and Kanade (1992), Sturm and Triggs (1996) collected these projective measurements into a rank-4 measurement

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

matrix $Y \in \mathbb{R}^{3n\times m}$ given by

$$Y = \begin{bmatrix} \lambda_{11}\mathbf{p}_{11} & \dots & \lambda_{1m}\mathbf{p}_{1m} \\ \vdots & \ddots & \vdots \\ \lambda_{n1}\mathbf{p}_{n1} & \dots & \lambda_{nm}\mathbf{p}_{nm} \end{bmatrix} = \begin{bmatrix} C_1 \\ \vdots \\ C_n \end{bmatrix} \begin{bmatrix} \mathbf{P}_1 & \dots \mathbf{P}_m \end{bmatrix}. \tag{2.6}$$

The crucial point here is that we do not have direct access to the projective depths λ_{ij}, and if λ_{ij} were to be accurately estimated, which constitutes the main contribution of Sturm and Triggs (1996), a factorization technique similar to that of Tomasi and Kanade (1992) could easily be applied. In order to achieve this goal, Sturm and Triggs (1996) considered the linear system of equations for pairs of cameras (i, j) given by

$$F_{ij}\mathbf{p}_{jk}\lambda_{jk} = (\mathbf{e}_{ij} - \mathbf{p}_{ik})\lambda_{ik}, \tag{2.7}$$

where the *fundamental matrix* $F_{ij} \in \mathbb{R}^{3\times 3}$ satisfies $E_{ij} = K_i^T F_{ij} K_j$, for K_i denoting the ith camera calibration matrix, E_{ij} denotes the essential matrix for the pair (i, j) (see (2.1)), and $\mathbf{e}_{ij}$ is the epipole on the ith image plane (see Figure 2.1). Considering the homogeneous linear equations (2.7) for the minimal number of $n - 1$ camera pairs, Sturm and Triggs (1996) estimated the projective depths up to global scale. After substitution of the recovered λ_{ij} in (2.6), they proceeded similarly to Tomasi and Kanade (1992) and factorized Y into its camera and structure components using SVD. However, this time there is no need to compute an undetermined multiplier Q since the camera and structure parts of Y do not have particular forms. Lastly, Sturm and Triggs (1996) concluded with empirical results to evaluate the performance of their algorithm. A closely related approach, with an additional section for the extension of the algorithm in Sturm and Triggs (1996) to the case of lines, instead of points, as feature measurements, is given by Triggs (1996).

An excellent account of factorization-based methods, for various camera models and alternative feature measurements, can also be found in the survey by Kanade and Morris (1998).

3. Camera location estimation

In this section we discuss the existing literature focusing primarily on the camera location estimation part of the SfM problem. We consider the camera location estimation methods based on corresponding point estimates between pairs of images. In the majority of the methods we discuss, the corresponding point estimates are used to make relative motion measurements between pairs of images, which can be decomposed into pairwise rotational and translational measurements. The generic approach for these methods is to estimate the camera orientations separately based on the

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

pairwise rotational measurements, and to use these camera orientation estimates together with the pairwise translational measurements in order to solve for the camera locations. On the other hand, we also discuss some of the methods in the literature that, to some degree, deviate from the generic recipe mentioned above, and aim to jointly estimate camera orientations and locations, use local structure estimates for location estimation, or consider triplets of cameras.

In order to illustrate the generic approach mentioned above, consider the simple pinhole camera model introduced in Section 2 and depicted in Figure 2.1. As discussed in Section 2, the essential matrix estimates, computed using the epipolar constraints (2.1) in the presence of sufficiently many corresponding scene points (see Figure 3.1 for a real example of pairwise point correspondences) and knowledge of the intrinsic camera parameters, can be uniquely factorized into relative rotational and translational parts, that is, into estimates of $R_i^T R_j$ and $[R_i^T(\mathbf{t}_j - \mathbf{t}_i)]_\times$. In general, the relative rotational parts are used for the estimation of camera orientations R_i; see Tron *et al.* (2016) for a detailed account of the existing methods for camera orientation estimation. In order to estimate the camera locations, the estimated orientations can be used directly with the translational parts to obtain estimates of the *pairwise directions* $\gamma_{ij} = (\mathbf{t}_i - \mathbf{t}_j)/\|\mathbf{t}_i - \mathbf{t}_j\|$. An alternative approach is to go back to the epipolar constraints (2.1) to rewrite them as a set of linear homogeneous equations in the camera locations $\mathbf{t}_i$ given by

$$(\mathbf{t}_i - \mathbf{t}_j)^T \mathbf{y}_{ij}^k = 0, \quad k = 1, \ldots, m_{ij}, \tag{3.1}$$

where $\mathbf{y}_{ij}^k$ are functions of R_i, R_j and the kth corresponding points $\mathbf{p}_i^k, \mathbf{p}_j^k$, and m_{ij} denotes the number of corresponding points for the cameras i and j. These equations can then be used directly for camera location estimation, or to obtain pairwise direction estimates.

A crucial point to note about the estimates of the pairwise directions γ_{ij} is that they lack any *scale information* pertaining to the distance $\|\mathbf{t}_i - \mathbf{t}_j\|$ between camera pairs. The difficulties[3] arising from this homogeneity can be regarded as the most important contributor to the diversity of the camera location estimation methods in the literature.

A relatively direct approach for location estimation from pairwise directions is based on the homogeneous system of equations

$$(I - \gamma_{ij}\gamma_{ij}^T)(\mathbf{t}_i - \mathbf{t}_j) = \mathbf{0}, \tag{3.2}$$

[3] Note that the pairwise relative motion measurements are also invariant to global orientation and location shifts, that is, any global rotational shift of the form $R_i \to RR_i$ and any global translation of the form $\mathbf{t}_i \to \mathbf{t}_i + \mathbf{t}$ for $R \in \mathrm{SO}(3)$ and $\mathbf{t} \in \mathbb{R}^3$ produce equivalent solutions. However, contrary to the case of invariance to scale transformations of the form $\mathbf{t}_i \to c\mathbf{t}_i$ for $c \in \mathbb{R}^+$, these invariances are not observed to result in any difficulties in motion estimation.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

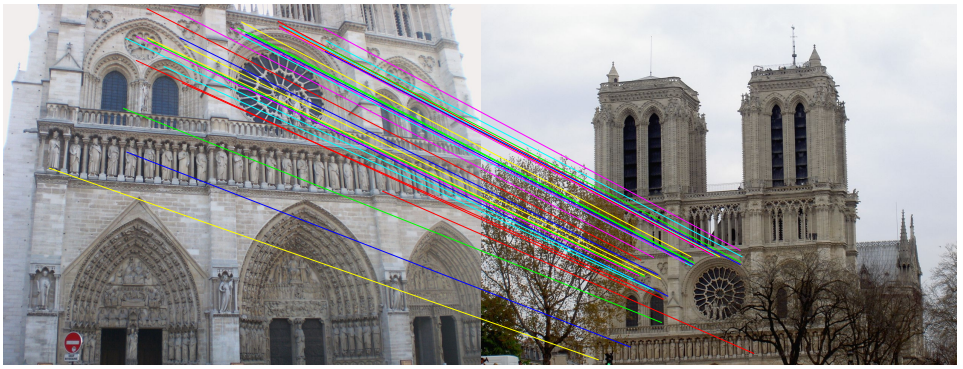


Figure 3.1. Two images from the Notre Dame Cathedral set in Snavely *et al.* (2006), with corresponding feature points, extracted using the scale-invariant feature transform (SIFT: Lowe 2004).

which encapsulates the fact that the difference vectors $\mathbf{t}_i - \mathbf{t}_j$ are parallel to the pairwise directions γ_{ij}. Brand, Antone and Teller (2004) estimated the locations by minimizing the *sum of squared errors* in the system obtained by replacing γ_{ij} in (3.2) with their estimates[4] and using constraints to fix the global scale and translation, given respectively by $\sum_i \|\mathbf{t}_i\|^2 = 1$ and $\sum_i \mathbf{t}_i = \mathbf{0}$, to prevent the trivial solution of $\mathbf{t}_i \equiv \mathbf{t}$ for some $\mathbf{t} \in \mathbb{R}^3$, which is summarized as

$$\begin{aligned} \underset{\{\mathbf{t}_i\}}{\text{minimize}} \quad & \sum_{i \sim j} (\mathbf{t}_i - \mathbf{t}_j)^T (I - \gamma_{ij}\gamma_{ij}^T)(\mathbf{t}_i - \mathbf{t}_j) \\ \text{subject to} \quad & \sum_i \|\mathbf{t}_i\|^2 = 1; \quad \sum_i \mathbf{t}_i = \mathbf{0}. \end{aligned} \tag{3.3}$$

Note that, considering the n locations stacked into a vector $\mathbf{t} \in \mathbb{R}^{3n}$, whose ith 3×1 block is equal to $\mathbf{t}_i$, we can rewrite the cost function of (3.3) as $\mathbf{t}^T L \mathbf{t}$, for some appropriate $L \in \mathbb{R}^{3n \times 3n}$, and the constraints as $\|\mathbf{t}\|^2 = 1$ and $V\mathbf{t} = \mathbf{0}$ (for $V = I_3 \otimes \mathbf{1}_n^T$). Then the solution to the least-squares problem (3.3) is given in terms of the (normalized) eigenvector corresponding to the fourth smallest eigenvalue of L, whose null space includes the three rows of V. Brand *et al.* (2004) also (heuristically) identified various classes of problems, which they referred to as 'problem pathologies', resulting in relatively unstable solutions. Among the listed 'pathologies', an important

[4] Brand *et al.* (2004) actually assumed a slightly more general setup, where the directional measurements are not necessarily unit length, that is, they may have some scale-related part (the source of which was unspecified). They also proposed a *maximum covariance spectral solution*, maximizing the alignment of each $\mathbf{t}_i - \mathbf{t}_j$ with γ_{ij}, instead of penalizing their deviation. We do not discuss this method due to its inferior performance, as reported by Brand *et al.* (2004).

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

one is the 'underconstrained' class of pathologies which possess additional degrees of freedom resulting in more than one set of location estimates (not related to each other via a global translation and/or scale transformation) that may be significantly different. As sources of these underconstrained pathologies, Brand *et al.* (2004) identified two (rather extreme) conditions, namely disconnectedness of the measurement graph[5] and the collinearity of all the directions at a single node, and also provide heuristic solutions to resolve the extra degrees of freedom; these two conditions are in fact only specific examples of a rich set of algebraic conditions defining underconstrained problems, as was fully characterized later by Özyeşil *et al.* (2015) and Özyeşil and Singer (2015). A similar location estimation method based on the system of equations (3.2) is given in Govindu (2001). In this method, instead of directly minimizing the sum of squared errors in the system (3.2), an iteratively reweighted least-squares solver, where in the $(k+1)$th iteration each equation in (3.2) is weighted with $1/\|\mathbf{t}_i^k - \mathbf{t}_j^k\|$, where $\mathbf{t}_i^k$ denotes the solution of the kth iteration, is used to democratize the contribution of each pair of cameras in the total cost function. Another approach, similar in spirit to these least-squares methods, is introduced in Arie-Nachimson *et al.* (2012). This time, instead of the system of equations (3.2), a least-squares method is used to minimize the sum of squared errors in the epipolar constraints (3.1).

Even though the least-squares methods of Brand *et al.* (2004), Govindu (2001) and Arie-Nachimson *et al.* (2012) are computationally very efficient and, for relatively small datasets, produce acceptable location estimates, the quality of their estimates degrades significantly for large, sparse, unordered and noisy datasets, that is, for most of the real datasets studied in the literature. Typically, they tend to produce spurious location estimates that cluster tightly around a small number of points: see Figure 3.2 for an example. This undesired tendency can be intuitively explained by considering the least-squares optimization problem (3.3), for which a clustering part of the solution results in a very small contribution to the cost function, and a few locations, usually corresponding to nodes in the measurement graph having few edges, are separated from the cluster to satisfy the constraints. In order to resolve this difficulty, several different methods have been proposed in the literature. A relatively straightforward starting point, introduced by Özyeşil *et al.* (2015), is to require the locations to satisfy a maximal proximity condition by incorporating 'repulsion constraints' of the form $\|\mathbf{t}_i - \mathbf{t}_j\| \geq c$, for some fixed $c > 0$ (taken to be $c = 1$ in Özyeşil *et al.* 2015), into the least-squares formulation of Brand *et al.* (2004). These nonconvex constraints convert the least-squares formulation into the notoriously

[5] The measurement graph is composed of nodes for each camera i and has an edge between nodes i and j if there is a direction estimate for this pair.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

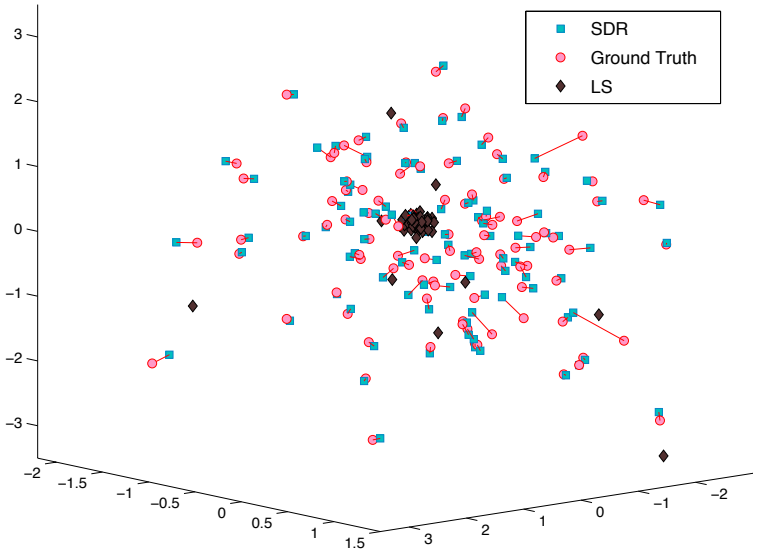


Figure 3.2. Estimates for a noisy, synthetic example of the camera location estimation problem for 100 cameras, demonstrating the clustering estimates of the least-squares (LS) method of Brand *et al.* (2004) versus estimates of the relatively stable semidefinite relaxation (SDR) solver of Özyeşil *et al.* (2015). The segments represent the error incurred by the SDR solution compared to the ground truth.

difficult problem

$$\begin{aligned} \underset{\{\mathbf{t}_i\}_{i\in V_t}\subseteq\mathbb{R}^d}{\text{minimize}} \quad & \sum_{i\sim j} \mathrm{Tr}((\mathbf{t}_i-\mathbf{t}_j)(\mathbf{t}_i-\mathbf{t}_j)^T(I-\gamma_{ij}\gamma_{ij}^T)) \\ \text{subject to} \quad & \|\mathbf{t}_i-\mathbf{t}_j\|_2^2 \geq 1, \quad \text{for all } (i,j)\in E_t, \\ & \sum_i \mathbf{t}_i = \mathbf{0}. \end{aligned} \tag{3.4}$$

To approximate this computationally difficult problem, Özyeşil *et al.* (2015) first introduce the rank-1, positive semidefinite matrix variable $T \in \mathbb{R}^{3n\times 3n}$ (where n denotes the number of cameras), the ijth 3×3 block T_{ij} of which is given by $T_{ij} = \mathbf{t}_i\mathbf{t}_j^T$. Using T, the cost function and the non-convex constraints of (3.4) are given, respectively, by linear functions

$$\sum_{i\sim j} \mathrm{Tr}(\mathcal{L}_{ij}(T)(I-\gamma_{ij}\gamma_{ij}^T))$$

and $\mathrm{Tr}(\mathcal{L}_{ij}(T)) \geq 1$ of T, where the operator $\mathcal{L}_{ij}(\cdot)$ satisfies

$$\mathcal{L}_{ij}(T) = T_{ii} + T_{jj} - T_{ij} - T_{ji} = (\mathbf{t}_i-\mathbf{t}_j)(\mathbf{t}_i-\mathbf{t}_j)^T. \tag{3.5}$$

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

As a result of the linearity in T of the cost function and the constraints, the non-convexity of (3.4) is fully represented by the non-convex rank-1 constraint on T (note that positive semidefiniteness is a convex constraint). As a result, Özyeşil *et al.* (2015) remove this final non-convex constraint to obtain the *semidefinite relaxation* formulation given by

$$\begin{aligned} \underset{T}{\text{minimize}} \quad & \sum_{i\sim j} \mathrm{Tr}(\mathcal{L}_{ij}(T)(I - \gamma_{ij}\gamma_{ij}^T)) \\ \text{subject to} \quad & \mathrm{Tr}(\mathcal{L}_{ij}(T)) \geq 1, \quad \text{for all } i \sim j, \\ & \mathrm{Tr}(VT) = 0, \\ & T \succeq 0. \end{aligned} \tag{3.6}$$

The approximate solution $\mathbf{t}_i$ in (3.4) is obtained to be the ith 3×1 block of the leading eigenvector of the solution of the SDR (3.6). The tightness[6] of this relaxation up to relatively high levels of noise is demonstrated empirically by Özyeşil *et al.* (2015), who also prove a *stability of recovery* result, which quantifies the amount of distortion in the location estimates to be of the order of the noise present in the pairwise directions, under fairly general assumptions for the noise model. To improve the quality of their location estimates, Özyeşil *et al.* (2015) propose a 'robust pairwise direction estimation' method based on the system (3.1), where the $\mathbf{y}_{ij}^k$ are viewed as noisy samples from the two-dimensional subspace orthogonal to γ_{ij} and efficient robust subspace recovery techniques are used to estimate each pairwise direction γ_{ij}. Further, and perhaps more importantly, Özyeşil *et al.* (2015) provide a complete characterization of *well-posed* location recovery from pairwise directions[7] by demonstrating its equivalence to the existing results in the field of *parallel rigidity theory*. For instance, while camera orientation estimation only requires the connectivity of the measurement graph, connectivity alone is insufficient to make a camera location estimation well-posed; see Figure 3.3 for an example. The complete characterization of well-posed location recovery problems (for arbitrary dimensions $d \geq 2$), via (generically) parallel rigid measurement graphs, is summarized in Theorem 3.1.

[6] Tightness here refers to obtaining the solution of the non-convex problem as the solution of the relaxed convex problem, that is, the two globally optimal solutions are the same and hence there is no *relaxation gap*.

[7] To be more precise, in Özyeşil *et al.* (2015), the information assumed to be available is actually a collection of potentially incomplete and noisy versions of the *pairwise lines* (or unsigned directions), which can be represented by the projection matrices $(\mathbf{t}_i - \mathbf{t}_j)(\mathbf{t}_i - \mathbf{t}_j)^T / \|\mathbf{t}_i - \mathbf{t}_j\|^2$. However, the well-posed examples of this case turn out to be the same as in the case of pairwise directions.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

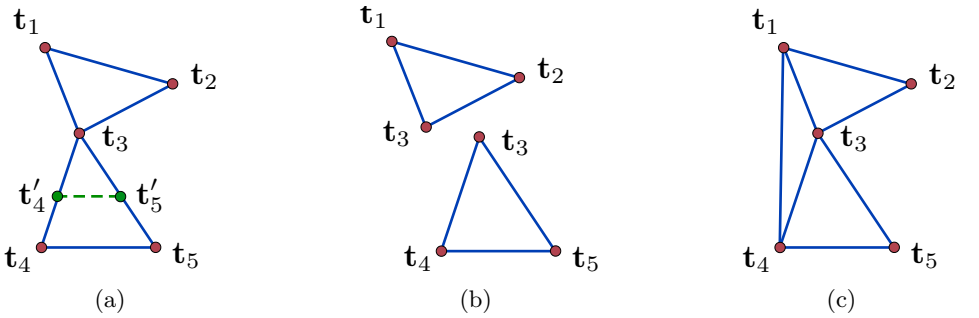


Figure 3.3. (a) A noiseless example of the camera location estimation problem for five locations on a connected graph (we assume noiseless direction measurements between location pairs having an edge), which is *not* parallel rigid (in $\mathbb{R}^2$ and $\mathbb{R}^3$). Non-uniqueness is demonstrated by two non-congruent location solutions $\{\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3, \mathbf{t}_4, \mathbf{t}_5\}$ and $\{\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3, \mathbf{t}'_4, \mathbf{t}'_5\}$, each of which can be obtained from the other by an *independent rescaling* of the solution for one of its maximally parallel rigid components. (b) Maximally parallel rigid components of the formation in (a). (c) A parallel rigid formation (in $\mathbb{R}^2$ and $\mathbb{R}^3$) obtained from the formation in (a) by adding the extra edge $(1, 4)$ linking its maximally parallel rigid components.

Theorem 3.1. A graph $G = (V, E)$ is generically parallel rigid in $\mathbb{R}^d$ if and only if it contains a non-empty set of edges $E' \subseteq E$, with $(d-1)|E'| = d|V| - (d+1)$, such that for all subsets E'' of E' we have

$$(d-1)|E''| \leq d|V(E'')| - (d+1), \tag{3.7}$$

where $V(E'')$ denotes the vertex set of the edges in E''.

Özyeşil *et al.* (2015) also provide an efficient randomized algorithm for testing well-posedness. Further, they introduce an efficient *alternating direction augmented Lagrangian method* (ADM) to solve the SDR (3.6), and a distributed algorithm to apply the SDR method to large sets of images.

An alternative two-step procedure, studied by Wilson and Snavely (2014), is composed of the detection of outliers among the pairwise direction measurements, via a procedure called 1DSfM, followed by a (non-convex) location estimation method; see Figure 3.4 for a simplified illustration of 1DSfM. The main idea in the outlier detection step is to project all of the pairwise directions onto a one-dimensional (random) subspace and try to solve the resulting ordering problem[8] as consistently as possible, from which

[8] This ordering problem, namely the *minimum feedback arc set problem*, is known to be an NP-complete problem, so Wilson and Snavely (2014) use a heuristic/approximate method observed to perform well in the literature.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

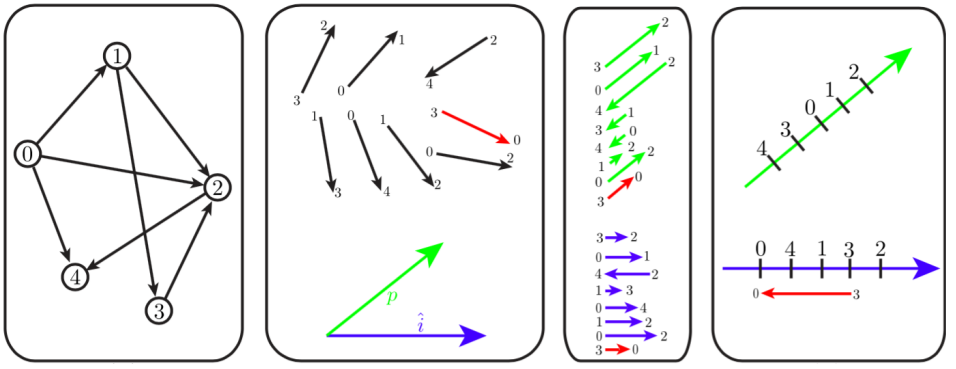


Figure 3.4. A simplified illustration (courtesy of Wilson and Snavely 2014) of outlier detection via 1DSfM. (a) Ground truth camera locations with (potentially noisy) pairwise directions available for camera pairs connected with arrows. (b) Available pairwise directions, with an outlier edge shown in red, and two directions, $\hat{i}$ and p, for one-dimensional (1D) projections. (c) Projections of pairwise directions onto $\hat{i}$ and p, resulting in two variants of the minimum feedback arc set problem. (d) Solutions to the 1D problems in (c), depicting the violation (satisfaction) of the 1D constraints by the outlier edge $(3, 0)$ in the 1D problem induced by $\hat{i}$ (p), since outlier edges may be consistent with only some of the solutions to the 1D subproblems.

potential outliers can be identified as having larger inconsistencies. The detection procedure 'solves' several of these randomized one-dimensional problems and identifies the outliers as those directions having an inconsistency score larger than a certain value. After removal of outlier direction measurements, the locations are estimated by an unconstrained maximization, using the Levenberg–Marquardt algorithm, of the non-convex cost function $\sum_{ij} \hat{\gamma}_{ij}^T(\mathbf{t}_i - \mathbf{t}_j)/\|\mathbf{t}_i - \mathbf{t}_j\|$ that measures the alignment of the pairwise directions $(\mathbf{t}_i - \mathbf{t}_j)/\|\mathbf{t}_i - \mathbf{t}_j\|$ with their 'cleaned' estimates $\hat{\gamma}_{ij}$. Wilson and Snavely (2014) also provide extensive experimental results for large datasets, demonstrating the accuracy and efficiency of their method.

Instead of the systems of equations (3.1) or (3.2), on which the various least-squares methods are based, an alternative approach aiming to prevent the clustering phenomena is to consider the system of equations

$$\mathbf{t}_i - \mathbf{t}_j - \alpha_{ij}\gamma_{ij} = \mathbf{0}, \tag{3.8}$$

where $\alpha_{ij} := \|\mathbf{t}_i - \mathbf{t}_j\|$. Note that if we ignore this defining (non-convex) relation between the α_{ij} and $\mathbf{t}_i$, (3.8) turns out to be a linear system in α_{ij} and $\mathbf{t}_i$ for given γ_{ij}. This system forms the basis of the location estimation method used by Tron and Vidal (2014), where the cost function is the sum

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

of squares of the errors in (3.8) and the additional constraints $\alpha_{ij} \geq 1$ are used to prevent the trivially collapsing solution $\mathbf{t}_i \equiv \mathbf{0}, \alpha_{ij} \equiv 0$. That is, for location estimation, Tron and Vidal (2014) solve

$$\begin{aligned} \underset{\{\mathbf{t}_i\},\{\alpha_{ij}\}}{\text{minimize}} \quad & \sum_{i \sim j} \|\mathbf{t}_i - \mathbf{t}_j - \alpha_{ij}\gamma_{ij}\|^2 \\ \text{subject to} \quad & \sum_i \mathbf{t}_i = \mathbf{0}; \quad \alpha_{ij} \geq 1. \end{aligned} \tag{3.9}$$

In fact, the main theme in Tron and Vidal (2014) is to provide a distributed algorithm for camera motion estimation, that is, joint estimation of orientations and locations. In this respect, they provide a *consensus*-based three-step algorithm, and local convergence and partial stability results (for the rotational part). The quadratic location estimation method (3.9) is used to obtain an initial set of locations (*i.e.* it constitutes the second step of their distributed algorithm), and it is then fused with the rotational part to obtain a joint motion estimation method. In terms of the camera location estimation problem with given pairwise directions, the crucial contribution of Tron and Vidal (2014) is using (3.8) as an alternative to (3.1) or (3.2), together with the relaxed repulsion constraints $\alpha_{ij} \geq 1$. As reported by Özyeşil and Singer (2015), for example, this framework provides location estimates with acceptable quality, which do not tend to cluster for large and noisy datasets, while preserving computational efficiency. A closely related optimization method based on (3.8) and the constraints $\alpha_{ij} \geq 1$ has been introduced by Moulon, Monasse and Marlet (2013). The cost function of this method is given by the *maximum absolute deviation* in the system (3.8).[9] Moulon *et al.* (2013) also introduce robust rotation and pairwise direction estimation methods. We note, however, that such maximal deviation penalty methods (*i.e.* ℓ_∞-norm minimization methods) are highly sensitive to outliers in the pairwise direction estimates.

3.1. Robust convex methods

The existence of a large proportion of outliers among pairwise direction estimates, for example stemming from systematically mismatched features between images as a result of symmetries or ambiguities in the images (see also Section 6.2), manifests itself as large errors in the estimated locations. For some datasets this difficulty may be tamed by preprocessing the pairwise directions for rejection of outliers (as in Wilson and Snavely 2014),

[9] The method of Moulon *et al.* (2013) actually assumes the possibility of multiple pairwise direction measurements for each camera pair i and j, computed from each triangle containing the pair.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

or by making use of consistencies in locally constructed 3D structures. However, generally applicable efficient algorithms, which are resilient to a large number of outliers and which exhibit provable convergence to globally optimal solutions and admit theoretical guarantees of robustness, are of high value. In this section we will consider two recent algorithms that have some of these desired properties.

The first location estimation algorithm, based on the system (3.8) and closely related to Tron and Vidal (2014) and Moulon *et al.* (2013), is given in Özyeşil and Singer (2015), where this time, instead of using a cost function given as the sum of squared errors (as in Tron and Vidal 2014) or the maximum error (as in Moulon *et al.* 2013) in the system of equations (3.8), the method minimizes the sum of *unsquared* errors in (3.8) (hence the name *least unsquared deviations* (LUD) method) by employing the same relaxed repulsion constraints $\alpha_{ij} \geq 1$. In other words, the $\mathbf{t}_i$ are estimated by solving

$$\begin{aligned} \underset{\{\mathbf{t}_i\},\{\alpha_{ij}\}}{\text{minimize}} \quad & \sum_{i\sim j} \|\mathbf{t}_i - \mathbf{t}_j - \alpha_{ij}\gamma_{ij}\| \\ \text{subject to} \quad & \sum_i \mathbf{t}_i = \mathbf{0}; \quad \alpha_{ij} \geq 1. \end{aligned} \tag{3.10}$$

The motivation for this formulation (mainly inspired by compressed sensing methods designed to recover corrupted signals from a minimal set of linear measurements), is to maintain robustness to outliers in the measurements of the pairwise directions γ_{ij}. As a convex *second-order cone program* (SOCP), the LUD solver (3.10) is a computationally efficient location estimator with guaranteed convergence to the globally optimal solution. More importantly, the rather interesting phenomenon of exact location recovery[10] by the LUD method in the presence of sufficiently many exact direction measurements (*i.e.* sufficiently few outlier direction measurements), the number of which varies with the level of sparsity in the measurement graph and the number of cameras, was first empirically demonstrated by Özyeşil and Singer (2015) (see Figure 3.5). An efficient iteratively weighted least-squares (IRLS) solver for the LUD problem (3.10) is provided in Özyeşil and Singer (2015).

Moreover, Özyeşil and Singer (2015) introduce a (non-convex) direction estimation algorithm based on the system (3.1) (similar to that of Özyeşil *et al.* 2015 but more efficient). The pairwise directions are estimated in a two-step procedure that first produces estimates of unsigned directions $\gamma_{ij}^0 = b_{ij}\gamma_{ij}$, and then computes the initially undetermined signs $b_{ij} \in \{-1, +1\}$

[10] Here, 'exact location recovery' corresponds to the estimation of locations up to a global scale and translation (*i.e.* up to a gauge freedom of the form $\mathbf{t}_i \to c\mathbf{t}_i + \mathbf{t}$ for arbitrary $c > 0$ and $\mathbf{t} \in \mathbb{R}^3$), whose ground truth values are known, with a total error smaller than a fixed value, which may be chosen to represent machine precision errors.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

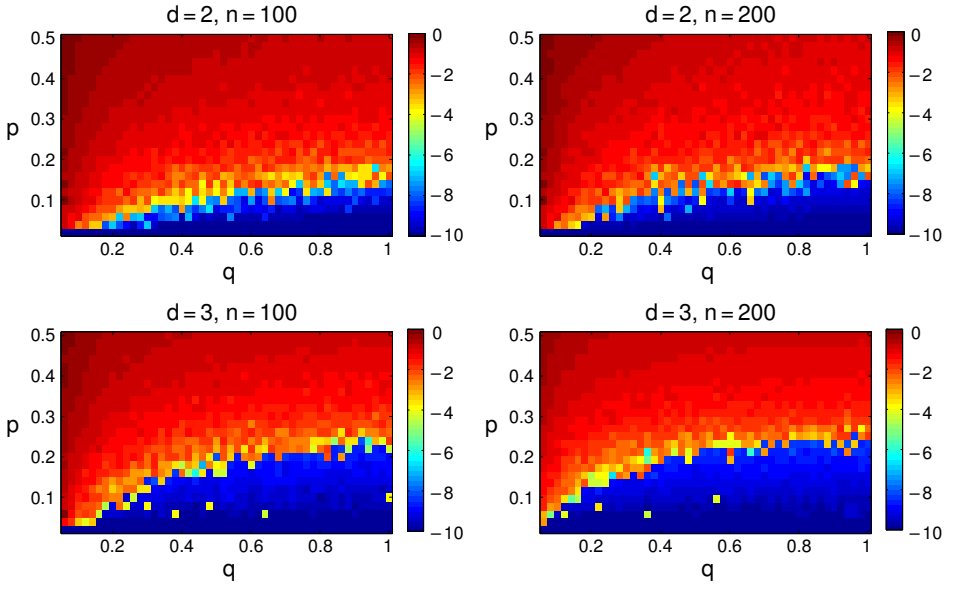


Figure 3.5. Recovery error of the LUD solver (3.10), demonstrating exact recovery regimes in the presence of outliers for dimensions $d = 2, 3$ and number of cameras $n = 100, 200$. Measurement graphs, with an edge between cameras having pairwise directions, are sampled from the Erdős–Rényi ensemble. Camera locations are i.i.d. Gaussian samples, independent of the measurement graph. The colour intensity of each pixel represents the logarithmic recovery error $\log_{10}$(NRMSE) (see equation (14) in Özyeşil and Singer (2015) for the definition of the normalized root mean squared error, NRMSE), depending on the probability q of the existence of an edge for each pair (x-axis), and the probability p of each existing edge being an outlier (y-axis). NRMSE values are averaged over 10 trials.

via the chirality requirement on the 3D points. In order to maintain robustness to outliers among estimates $\hat{\mathbf{y}}_{ij}^k$ of $\mathbf{y}_{ij}^k$ in (3.1) for the estimation of unsigned directions γ_{ij}^0, Özyeşil and Singer (2015) consider the following (non-convex) problem:

$$\begin{aligned} \underset{\gamma_{ij}^0}{\text{minimize}} \quad & \sum_{k=1}^{m_{ij}} |(\gamma_{ij}^0)^T \hat{\mathbf{y}}_{ij}^k| \\ \text{subject to} \quad & \|\gamma_{ij}^0\| = 1. \end{aligned} \tag{3.11}$$

To obtain the estimate $\hat{\gamma}_{ij}^0$ they use a (heuristic) IRLS method, which is not guaranteed to converge to global optima (since the optimization problem (3.11) is not convex). However, they provide empirical evidence for the

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

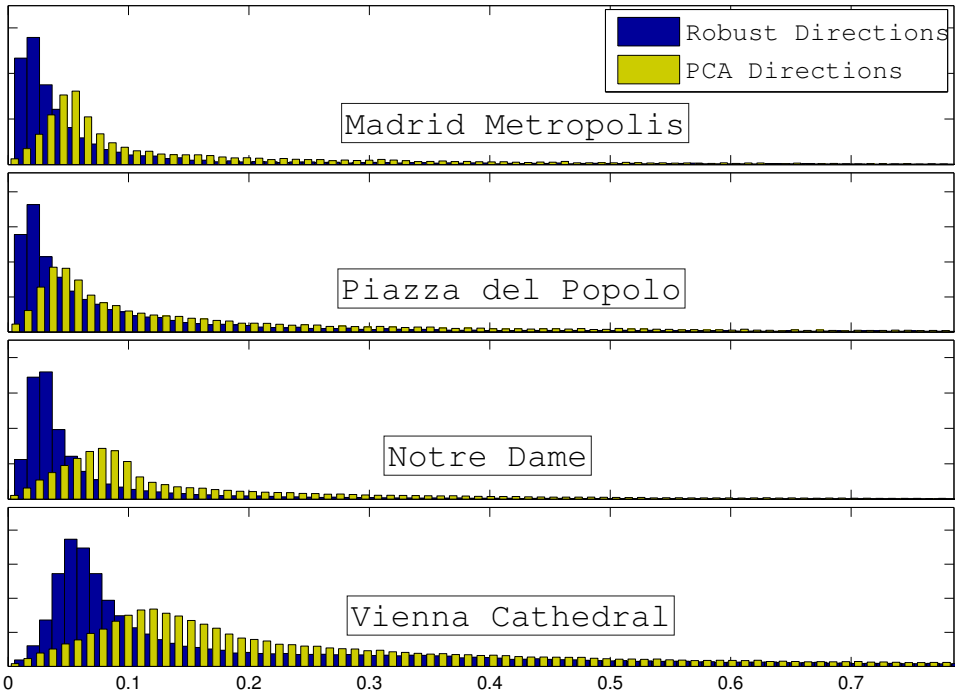


Figure 3.6. Histogram plots of the errors in direction estimates computed by the robust method (3.11) and the PCA method, which replaces the cost function in (3.11) with the sum of squares version $\sum_{k=1}^{m_{ij}}((\gamma_{ij}^0)^T\hat{\mathbf{y}}_{ij}^k)^2$, for some of the datasets studied in Özyeşil and Singer (2015). The errors represent the angles between the estimated directions and the corresponding ground truth directions, computed from a sequential SfM method based on Snavely *et al.* (2006), and provided in Wilson and Snavely (2014). The errors take values in $[0, \pi]$, but the histograms are restricted to $[0, \pi/4]$ to emphasize the difference in quality for the estimated directions.

robustness of the solution to (3.11), while preserving computational efficiency (see Figure 3.6). Finally, Özyeşil and Singer (2015) demonstrate the high quality and efficiency of the overall method on large datasets from Wilson and Snavely (2014).

The excellent empirical performance of the LUD method has recently inspired the development of a new method for location estimation from pairwise directions called *ShapeFit* (Hand, Lee and Voroninski 2017). ShapeFit is based on a convex program that minimizes the sum of the *unsquared* errors in the system (3.2) subject to the constraint $\sum_{ij}(\mathbf{t}_i - \mathbf{t}_j)^T\gamma_{ij} = 1$, which fixes the scale and sign of the location estimates, as well as encouraging correlation with the true relative directions. Specifically, the ShapeFit

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

program consists of the following:

$$\begin{aligned} \underset{\{\mathbf{t}_i\}}{\text{minimize}} \quad & \sum_{i\sim j} \|(I - \gamma_{ij}\gamma_{ij}^T)(\mathbf{t}_i - \mathbf{t}_j)\|_2 \\ \text{subject to} \quad & \sum_i \mathbf{t}_i = \mathbf{0}, \quad \sum_{i\sim j} (\mathbf{t}_i - \mathbf{t}_j)^T \gamma_{ij} = 1. \end{aligned} \tag{3.12}$$

Similar to the LUD method of Özyeşil and Singer (2015), the motivation behind the cost function of ShapeFit is to improve robustness to outliers among the pairwise direction measurements. Also, ShapeFit is amenable to theoretical analysis, and the results of Hand *et al.* (2017) provide the first rigorous results guaranteeing exact location recovery from corrupted relative direction observations. More concretely, Theorem 1 in Hand *et al.* (2017) considers a high-dimensional version of the location recovery problem (*i.e.* $\mathbf{t}_i \in \mathbb{R}^d$ for sufficiently large d) and proves exact location recovery with high probability for i.i.d. Gaussian locations in the presence of a measurement graph sampled from an Erdős–Rényi ensemble, for which at most a constant fraction of direction measurements for any particular location is arbitrarily corrupted. For the same setup, with the exception of requiring the fraction of corrupted directions measurements for any particular location to be at most a constant, *up to poly-logarithmic factors*, Theorem 2 of Hand *et al.* (2017) proves exact location recovery for the physically relevant case $d = 3$. The physically relevant theorem is as follows.

Theorem 3.2. There exists $n_0 \in \mathbb{N}$ and $c \in \mathbb{R}$ such that the following holds for all $n \geq n_0$. Let the measurement graph $G([n], E)$ be drawn from the Erdős–Rényi ensemble $G(n, p)$ for some $p = \Omega(n^{-1/5} \log^{3/5} n)$. Let $\mathbf{t}_1^{(0)}, \ldots, \mathbf{t}_n^{(0)} \in \mathbb{R}^3$ denote the ground truth location, where $\mathbf{t}_i^{(0)} \sim \mathcal{N}(0, I_{3\times 3})$ are i.i.d. independent from G. There exists $\gamma = \Omega(p^5/\log^3 n)$ and an event of probability at least $1 - 1/n^4$ on which the following holds.

For arbitrary subgraphs $E_b \subset E$ satisfying that, for any vertex i, the number of edges emanating from i which are corrupted is bounded by γn, and for arbitrary pairwise direction corruptions $\mathbf{v}_{ij} \in \mathbb{S}^2$ for $(i, j) \in E_b$, the convex program (3.12) has a unique minimizer equal to

$$\left\{\alpha\left(\mathbf{t}_i^{(0)} - \bar{\mathbf{t}}^{(0)}\right)\right\}_{i\in[n]}$$

for some positive α and for

$$\bar{\mathbf{t}}^{(0)} = \frac{1}{n} \sum_{i\in[n]} \mathbf{t}^{(0)}.$$

Both of these theorems follow from the much more general Theorem 3 in Hand *et al.* (2017), which establishes that ShapeFit succeeds in recovering locations from corrupted relative directions under broad deterministic

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

assumptions on the measurement graph G and the geometry of locations. Note that the setting for ShapeFit theorems is that of robust statistics, in that the corruptions are assumed to be adversarial, and thus for a fixed set of locations and observations the theorem above establishes that ShapeFit succeeds uniformly in the data corruptions, provided there is a bound on the number of corrupted edges emanating from any particular location node.

The proof strategy for ShapeFit theorems is a direct analysis of the optimality conditions of the convex objective function, which relies on the combinatorial propagation of various local geometric properties. For instance, one such property is that if a collection of triangles share the same base and the locations opposite the base are sufficiently 'well-distributed', then an infinitesimal rotation of the base induces infinitesimal rotations in edges of many of the triangles. It is then ensured that for each corrupted edge (k, l), one can find sufficiently many triangles in the observation graph with two good edges and base (k, l), with locations at the opposing vertices being 'well-distributed'. However, the full proof is more nuanced, and needs the constraints of the ShapeFit program. The other important local property is that for a tetrahedron in $\mathbb{R}^3$ with well-distributed vertices, any discordant parallel deviations on two of its disjoint edges induce enough infinitesimal rotational motion on some other edge of the tetrahedron. Combinatorial propagation is then handled separately in two regimes of the relative balance of parallel deviations on the good and corrupted subgraphs.

Hand *et al.* (2017) also provide empirical evidence supporting the main theoretical results about ShapeFit. These results demonstrate that the ShapeFit method can tolerate a larger fraction of pure outliers as compared to the LUD estimator for exact location recovery from randomly generated data, whereas in a mixed noise setting the LUD solver has a smoother change in its recovery performance (as corruption probability varies) and lower (higher) recovery errors in the high (low) corruption probability regime. More importantly, Goldstein, Hand, Lee, Voroninski and Soatto (2016) test ShapeFit empirically on a dozen standard 'photo tourism' datasets. They introduce a novel numerical framework for solving programs like ShapeFit and LUD, leading to an ADMM-based approach called ShapeKick, which solves location recovery problems orders of magnitude faster than competing methods. Goldstein *et al.* (2016) provide a comparative study, with respect to competing methods such as LUD and 1DSfM, demonstrating the accuracy and high efficiency of ShapeFit and ShapeKick.

3.2. Other methods

One of the methods deviating from the generic recipe considered above is studied in Govindu (2004). In this method, camera orientations and locations are jointly estimated using an *iterative averaging method on the*

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

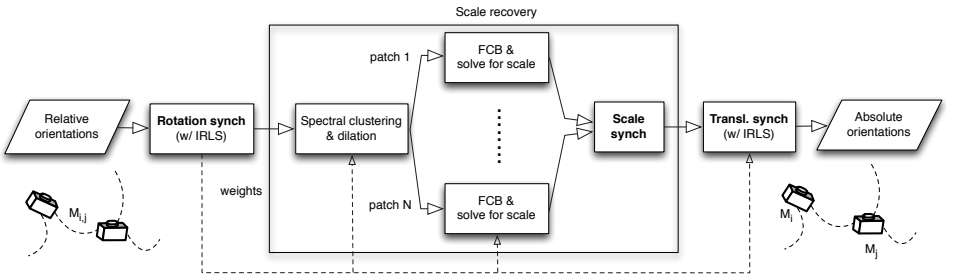


Figure 3.7. Depiction of the group synchronization pipeline (GSP) in Arrigoni *et al.* (2016), including the group synchronization stages in bold. Figure courtesy of Arrigoni *et al.* (2016).

Lie algebra of the group of Euclidean motions. Although Govindu's paper lacks theoretical and strong empirical stability guarantees, its approach is an efficient alternative for joint motion estimation in the low-noise regime. Another recent method slightly deviating from the generic recipe has been introduced by Arrigoni, Fusiello and Rossi (2016). Called the *group synchronization pipeline* (GSP), this method is based on a three-step estimation, via (robust) synchronization, of the camera orientations in SO(3), baseline scales[11] (*i.e.* α_{ij} in (3.8)) in $\mathbb{R}^+$, and the camera locations in $\mathbb{R}^3$; see Figure 3.7 for a depiction of the GSP. The term *synchronization* here refers to a class of methods aiming to (globally) estimate a set of mathematical group elements $\{g_i\}$ from a potentially noisy and incomplete set of observations of their pairwise ratios[12] $g_i^{-1}g_j$. The term was originally coined by Singer (2011), who studied the problem of estimating planar rotations from measurements of their pairwise ratios. As an example of a synchronization problem, in order to estimate the camera orientations $R_i \in \mathrm{SO}(3)$ (represented as 3×3 orthonormal matrices with determinant one) from measurements $\hat{R}_{ij}$ of their ratios $R_{ij} = R_i^T R_j$, we can consider the problem

$$\begin{aligned} \underset{\{R_i\}}{\text{minimize}} \quad & \sum_{i\sim j} \|R_i - R_j \hat{R}_{ij}^T\|_F^2 \\ \text{subject to} \quad & \{R_i\} \subseteq \mathrm{SO}(3), \end{aligned} \tag{3.13}$$

which is a difficult non-convex problem due to the set of constraints. However, the problem (3.13) (and its variations using robust cost functions

[11] Note that Arrigoni *et al.* (2016) perform the synchronization of the baseline scales via a distributed two-step method in order to mitigate error propagation, where each step uses spectral methods to produce its estimates.

[12] Such an estimate can be obtained only up to a global multiplication gg_i of the group elements by a group element g, since the ratios $g_i^{-1}g_j$ are invariant to such transformations.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

instead of the sum of squared deviations in (3.13)) has been shown in the literature to admit high-quality approximations via convex relaxations and spectral methods; see for example Arie-Nachimson *et al.* (2012), Özyeşil *et al.* (2015), Chatterjee and Govindu (2013), Wang and Singer (2013) and Cucuringu, Singer and Cowburn (2012). The estimates of Arrigoni *et al.*'s first two steps are fused into the available data in the last step to produce the final location estimates via an iteratively reweighted least-squares (IRLS) solver for the resulting linear system of equations. The high quality of the overall approach in Arrigoni *et al.* (2016) is empirically presented on large datasets. Instead of relying solely on relations between pairs of cameras, Jiang, Cui and Tan (2013) introduce a linear method based on minimizing the error in a set of geometrical relations among triplets of cameras (for details see Jiang *et al.* 2013, Section 4.1), which requires knowledge of *ratios* of baseline lengths computed from locally reconstructed scene points. Furthermore, to improve the quality of the location estimates in the presence of outliers among pairwise measurements, Jiang *et al.* (2013) employ various heuristic outlier rejection techniques. We note that although this linear method has some desirable properties, such as being applicable to collinear motion, computational efficiency, *etc.*, it is rather sensitive to outlier measurements, and hence the outlier rejection techniques proposed by Jiang *et al.* are of critical importance to maintaining stability. Another alternative location estimation approach, introduced by Sinha, Steedly and Szeliski (2010), which aims to tame the instability resulting from unknown scales, uses two-view reconstructions for camera pairs. The main idea is to obtain relative scales and translation between the two-view reconstructions sharing sufficiently many 3D points, and to use this information in the camera location and scale estimation. The pipeline in Sinha *et al.* (2010) also produces an initial 3D structure estimate, which, together with its motion estimate, can be used to initialize reprojection error minimization algorithms for final structural refinement.

4. Structure estimation

This section considers some of the key contributions in the literature mainly concerning the 3D structure estimation part of the SfM problem. These works have a relatively wide spectrum, including incremental and global methods, joint structure and motion estimation methods, and methods specifically targeting very large-scale SfM. Typically, the number of 3D structure points in SfM is much greater than the number of corresponding cameras: for an unordered collection of about 10^3 cameras, for example, it is reasonable to expect of the order of 10^6 structure points. Therefore, the fundamental challenge is the *efficient* processing of the given data in a consistent way, while maintaining robustness to noise in the data.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

The primary algorithms for joint refinement of the 3D structure, the camera motion and possibly also the intrinsic camera parameters (also called the camera calibration parameters) are known as *bundle adjustment methods*. As indicated in Triggs *et al.*'s excellent survey of bundle adjustment (see Triggs *et al.* 2000, Figure 9), their origin can be traced back to the works of Gauss and Legendre on least-squares estimation in the late eighteenth and early nineteenth centuries. However, in this section we will focus on more recent developments, and more generally on structure estimation methods that mainly aim to facilitate the processing of large unordered sets of images.

Triggs *et al.* (2000) discussed several different aspects of bundle adjustment, including the basic projection model and problem parametrization, error modelling and the choice of cost function, the main types of numerical optimization algorithms, the sparse problem structure as induced by the network of variables (and how to exploit it to improve reconstruction accuracy), gauge invariance, and quality control to detect outliers and characterize the overall accuracy of the estimated parameters. They presented the projection model and the problem parametrization in a generalized, rather abstract framework to incorporate various scene models (*e.g.* relatively simple isolated 3D features made up of points, lines or planes, or more complicated models involving dynamics, photometry or complex objects linked by constraints) and camera models (*e.g.* perspective, affine or orthographic projections, or less frequently encountered models such as rational polynomial cameras). To define a *feature prediction error* (also sometimes referred to as the *total reprojection error*), Triggs *et al.* (2000) considered a simple scene model comprising static, individual (abstract) 3D features $\mathbf{X}_j$ imaged by a set of cameras corresponding to the position and intrinsic calibration parameters $\mathbf{C}_i$. In this framework we are given noisy measurements $\bar{\mathbf{x}}_{ij}$ of the image features $\mathbf{x}_{ij}$ representing the true image of features $\mathbf{X}_j$ in image i, that are assumed to satisfy a *predictive model* given by $\mathbf{x}_{ij} = \mathbf{x}(\mathbf{C}_i, \mathbf{X}_j)$. The feature prediction errors for the measurements $\bar{\mathbf{x}}_{ij}$ are then defined by

$$\Delta\mathbf{x}_{ij}(\mathbf{C}_i, \mathbf{X}_j) := \bar{\mathbf{x}}_{ij} - \mathbf{x}(\mathbf{C}_i, \mathbf{X}_j). \tag{4.1}$$

To reconstruct the scene, a measure of these prediction errors is minimized, where bundle adjustment corresponds to the *refinement* part of this optimization, and requires an initial set of estimates for the camera and structure parameters (*e.g.* computed using a simpler technique). In summary, as noted by Triggs *et al.* (2000, p. 302), bundle adjustment 'is essentially a matter of optimizing a complicated nonlinear cost function (the total prediction error) over a large nonlinear parameter space (the scene and camera parameters)', the accuracy of which is highly dependent on the initial estimates of camera and structure parameters (explaining, for example, the wide

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

variety of initial motion estimation methods), and on the choice of the parametrization of the parameter space; for a more detailed discussion see Triggs *et al.* (2000, Section 2.2). Moreover, Triggs *et al.* studied the choice of the cost function defining the measure of the errors in (4.1), reaching the main conclusion that robust, statistically based error metrics allowing the presence of outliers in the measurements are appropriate choices. They discussed three main categories of algorithms for the numerical optimization of the chosen cost function in detail, namely the second-order Newton-style methods, first-order methods (demonstrating linear asymptotic convergence) and the sequential methods incorporating a series of observations one by one (as opposed to solving for the whole system globally). For second-order methods, which have fast asymptotic convergence rates but relatively high costs per iteration, they explored several different techniques for improving efficiency, mainly exploiting the sparsity patterns of the Hessian of the cost function. They also discussed various first-order methods, including the simple gradient descent, linear and non-linear Gauss–Seidel methods and Krylov subspace methods. For the third category of sequential algorithms, they considered low-rank update techniques for updating the Hessian and various filtering techniques. Triggs *et al.* (2000) also provided methods for handling theoretical and algorithmic difficulties that arise from several gauge symmetries involved in the problem, and (essentially) concluded with various analytical and heuristic techniques for quality control, that are used to maintain algorithmic stability and robustness to outlier measurements, and to characterize the overall accuracy of the estimates.

A relatively recent work on large-scale bundle adjustment (for tens of thousands of images) is that of Agarwal, Snavely, Seitz and Szeliski (2010). The main challenge identified by Agarwal *et al.* (2010) is the low sparsity levels (*i.e.* relatively high density) present in the connectivity graphs[13] of *unstructured* sets of images, such as community photo collections, as opposed to the higher sparsity of the *structured* image sets (*e.g.* street-level datasets such as images captured from a moving truck); for an example see Figure 4.1. The decrease in sparsity drastically increases the computational cost of the sparse factorization methods employed in classical bundle adjustment methods. To address this difficulty, the inexact (truncated) Newton-type algorithm introduced by Agarwal *et al.* (2010) makes use of conjugate gradients for the Newton step computation combined with various preconditioners. In more detail, they consider a non-linear least-squares[14]

[13] Here, 'connectivity graph' refers to the graph, the nodes of which denote the cameras corresponding to the images, and the edges of which exist between camera pairs having sufficiently many corresponding feature points in their corresponding images.

[14] As noted by Agarwal *et al.* (2010), the simple non-linear least-squares setup encapsulates cost functions more general than the ℓ_2-norm. For example, robust cost functions

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press



(a) structured: 6375 photos (b) unstructured: 4585 photos

Figure 4.1. Sparsity patterns of the adjacency matrices of the connectivity graphs for two sets of images: (a) a structured set of photos (taken from a moving truck), and (b) an unstructured collection of community photos corresponding to a search for the term 'Dubrovnik' in Flickr. Figure courtesy of Agarwal *et al.* (2010).

bundle adjustment problem stated (abstractly) as

$$\underset{\mathbf{x}}{\text{minimize}} \quad \frac{1}{2}\|\mathbf{F}(\mathbf{x})\|^2, \tag{4.2}$$

where $\mathbf{x} \in \mathbb{R}^p$, $\mathbf{F} : \mathbb{R}^p \to \mathbb{R}^q$. Let $J(\mathbf{x}) \in \mathbb{R}^{q \times p}$ denote the Jacobian of $\mathbf{F}$ at $\mathbf{x}$ and let

$$\mathbf{g}(\mathbf{x}) := \nabla \frac{1}{2}\|\mathbf{F}(\mathbf{x})\|^2 = J(\mathbf{x})^T \mathbf{F}(\mathbf{x}).$$

Agarwal *et al.* (2010) consider a Levenberg–Marquardt (LM) approach to solving (4.2), which, to compute the update $\mathbf{x} \to \mathbf{x} + \Delta\mathbf{x}$ at each iteration (while ensuring convergence by controlling the step size $\|\Delta\mathbf{x}\|$), solves the regularized least-squares problem

$$\underset{\Delta\mathbf{x}}{\text{minimize}} \quad \frac{1}{2}\big(\|J(\mathbf{x})\Delta\mathbf{x} + \mathbf{F}(\mathbf{x})\|^2 + \mu\|D(\mathbf{x})\Delta\mathbf{x}\|^2\big), \tag{4.3}$$

where $D(\mathbf{x})$ is a non-negative diagonal matrix, for example, given by

$$D(\mathbf{x})_{ii} = \sqrt{\sum_j J(\mathbf{x})_{ji}^2},$$

and $\mu > 0$ determines the regularization strength. Note that if the regularized Hessian $H_\mu(\mathbf{x}) := J(\mathbf{x})^T J(\mathbf{x}) + \mu D(\mathbf{x})^T D(\mathbf{x})$ is positive definite, we can solve (4.3) via the *normal equations*

$$H_\mu(\mathbf{x})\Delta\mathbf{x} = -\mathbf{g}(\mathbf{x}). \tag{4.4}$$

such as the Huber norm can be used by casting the problem as reweighted non-linear least-squares.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

Here, the crucial point is that $H_\mu(\mathbf{x})$ usually turns out to have a special block structure induced by the sparsity pattern of the bundle adjustment problem, which simplifies the solution to (4.4). For instance, assuming that the components of $\mathbf{F}(\cdot)$ only depend on a single camera or a single 3D feature, which is usually the case, then we get

$$H_\mu = \begin{bmatrix} B & E \\ E^T & C \end{bmatrix}, \tag{4.5}$$

where B and C are block diagonal (with blocks of size < 10, and hence can be efficiently inverted), and E is block sparse. Therefore, by also considering the appropriate block representations $\Delta\mathbf{x} = [\Delta\mathbf{y}, \Delta\mathbf{z}]$ and $-g = [v, w]$, and solving for $\Delta\mathbf{z}$ by $\Delta\mathbf{z} = C^{-1}(w - E^T\Delta\mathbf{y})$, we obtain

$$\left[B - EC^{-1}E^T\right]\Delta\mathbf{y} = v - EC^{-1}w. \tag{4.6}$$

Here the Schur complement S of C in H_μ given by $S = B - EC^{-1}E^T$, and known as the *reduced camera matrix*, has a block sparse structure with a non-zero ijth block if and only if the ith and the jth images have at least one common feature point. As noted by Agarwal *et al.* (2010), computationally speaking, the reduction of (4.4) to (4.6), via the *Schur complement trick* given above, allows us to obtain a solution to (4.4) using classical Cholesky factorization methods (that do not exploit the sparsity pattern in S) in $O(n^2)$ space and $O(n^3)$ time complexity, where n denotes the number of cameras. These computational costs become prohibitive for large datasets, and hence an alternative is to use decomposition algorithms that exploit the sparsity in S. However, as the datasets grow and the level and simplicity of sparsity in S decreases (as argued by Agarwal *et al.* 2010 for the case of community photo collections), even the construction and storage of S becomes computationally challenging. As a result, instead of trying to solve (4.3) exactly, for example by using the Schur complement trick discussed above, they propose using an inexact Newton method based on an approximate iterative linear system solver, such as conjugate gradient (CG), possessing a termination rule given by

$$\|H_\mu(\mathbf{x})\Delta\mathbf{x} + \mathbf{g}(\mathbf{x})\| \leq \eta_k\|\mathbf{g}(\mathbf{x})\|, \tag{4.7}$$

where $0 < \eta_k \leq \eta_0 < 1$, k denoting the LM iteration number, is used as a forcing sequence to guarantee convergence. Further, Agarwal *et al.* (2010) consider several *preconditioners* M for which the matrix $M^{-1}H_\mu(\mathbf{x})$ of the equivalent linear system $M^{-1}H_\mu(\mathbf{x})\Delta\mathbf{x} = -M^{-1}\mathbf{g}(\mathbf{x})$ has reduced condition number, which therefore improves the CG convergence rate. Agarwal *et al.* (2010) conclude their paper by extensively comparing the experimental stability and efficiency of their preconditioned CG method with other preconditioners, and with the classical methods factoring the Schur complement,

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

obtaining significant improvements in accuracy and efficiency, in both time and memory, using their proposed algorithm.

For large, unordered and hence computationally challenging datasets, Crandall, Owens, Snavely and Huttenlocher (2011) have introduced an alternative global approach, whose formulation is based on first computing a coarse initial solution via a hybrid *discrete-continuous optimization*, using discrete belief propagation (BP) within a Markov random field (MRF) formulation of pairwise camera and camera-point constraints to estimate camera parameters and a continuous LM improvement, and then refining the resulting solution with classical bundle adjustment. This method also employs various alternative sources of data such as noisy geotags and vanishing point estimates extracted from images. More concretely, the method first solves for camera orientations by minimizing a robust cost function of the error in the (classical) pairwise rotational consistency equations (*i.e.* the equations $R_{ij} = R_i^T R_j$, for which we have access to noisy measurements of some of the relative rotations R_{ij}), using a mixture of BP (on a discretization of the 3D sphere) and LM refinement. Having solved for the orientations R_i, the joint estimation of the camera locations and structure points is based on the minimization of (a robust cost function of) the total *angles of deviation* between pairwise translations $\mathbf{t}_i - \mathbf{t}_j$ and pairwise directions γ_{ij} (see *e.g.* (3.2)) and between 3D point-to-camera displacements (given by $\mathbf{P}_k - \mathbf{t}_i$ between the kth 3D point $\mathbf{P}_k$ and camera i) and 'ray directions' (given in the noiseless case by $\mathbf{r}_{ik} = R_i^T K_i^{-1} \mathbf{p}_{ik}$, for the known intrinsic matrix K_i of camera i, and the projective measurement $\mathbf{p}_{ik}$ of $\mathbf{P}_k$ by camera i). This minimization is again performed with a hybrid BP and LM method. Even though, in principle, such non-convex methods are highly sensitive to initial points, and hence susceptible to local minima, Crandall *et al.* (2011) have been able to improve its sensitivity and obtain, as empirically demonstrated, high-quality reconstructions by also making use of external, additional sources of information like geotags and vanishing points. Further, the parallelizable structure of the methods in Crandall *et al.* (2011) allows the realization of significant improvements in computational efficiency.

A relatively popular main approach to handling challenging datasets is to employ *sequential processing* of the available images, that is, to incorporate the images one by one, or in small groups, to obtain a solution for the whole set based on a 'core solution' computed from a small subset of the available data. A systematic method of this sequential type was introduced by Snavely, Seitz and Szeliski (2008*b*), whose main idea was to identify a small subset of the available images, called the *skeletal set*, by first estimating the accuracy of pairwise reconstructions of overlapping images and then extracting a subset whose reconstruction accuracy approximates that of the total set. The remaining images are then incorporated by position estimation, and a final bundle adjustment can be used to improve accuracy. The

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

computation of the skeletal set is rather complicated,[15] but the main idea is to identify a (directed) measure of proximity between cameras, identified as nodes of the *image graph*, given by the (translational) uncertainty in the pairwise reconstruction (obtained by fixing the location and orientation of one of the cameras in the pair and then repeating it for the other one) as quantified by the trace of the *covariance* matrix given by the translational part of the inverse of the Hessian of the reduced camera system (4.6) (*i.e.* the Schur complement $S = B - EC^{-1}E^T$ for the camera pair); see Figure 4.2 for a simplified depiction. The skeletal set problem is to identify the smallest subset of cameras, subject to the constraint that the distance (length of the shortest path, as defined by the total uncertainty of the path) between any pair of cameras in the (directed) subgraph induced by this subset is at most t times longer than their distance in the original digraph (induced by the set of all cameras). The factor t is called the *stretch factor*, and controls the amount of additional uncertainty introduced between camera pairs by removing cameras from the available dataset. Also, the skeletal set is required to satisfy the constraint of being uniquely realizable, which is modelled and controlled via a structure that they call the *pair graph*. Unfortunately, this problem (even if we ignore the additional realizability constraint), known as *computing a minimum t-spanner* for the digraph of camera pairs, is NP-complete for general graphs. Hence, Snavely *et al.* (2008*b*, Section 4) employ a two-step heuristic approximation to compute the skeletal set. The empirical results provided by Snavely *et al.* (2008*b*) demonstrate a significant increase in overall SfM efficiency, and the accuracy of the reconstructed structure as compared to applying existing SfM techniques to the full set of images.

Another incremental method, similar in spirit to Snavely *et al.* (2008*b*), has been studied by Havlena, Torii and Pajdla (2010). Their main idea for improving the efficiency of the overall SfM pipeline is to identify a small subset of input images by computing an approximate *minimally connected dominating set*, for which each of the remaining images has significant overlaps with at least one of the images in this set, and to employ *task prioritization* to promote easier image matching. Moreover, instead of computing actual image matchings, which is computationally prohibitive for thousands of cameras if performed on each pair of images, they employ fast image indexing techniques based on large image vocabularies to measure visual overlaps between images. To identify image similarities, they use a

[15] For brevity, we skip the finer details of the computation of the skeletal set in Snavely *et al.* (2008*b*), such as removal of outlier pairwise reconstructions, removal of (duplicate) images with similar content, or identification of *infeasible paths* between triplets of images using higher-level connectivity structures called the *pair graph* and its augmentation with the *image graph*.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

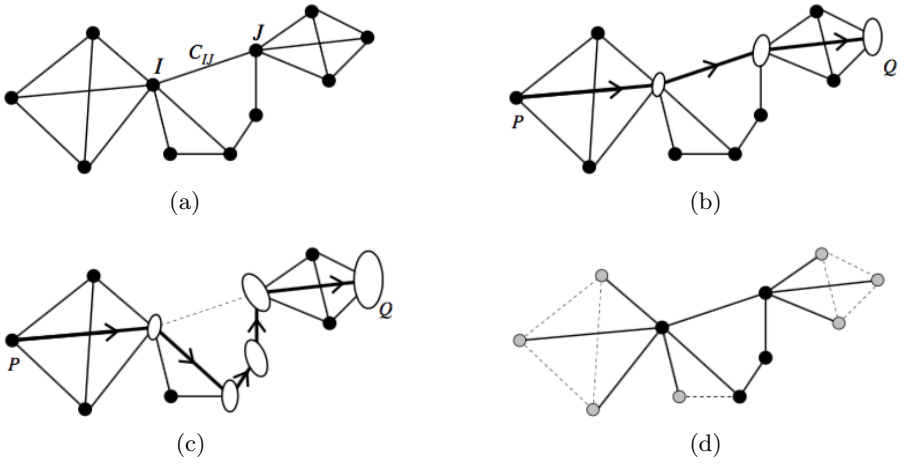
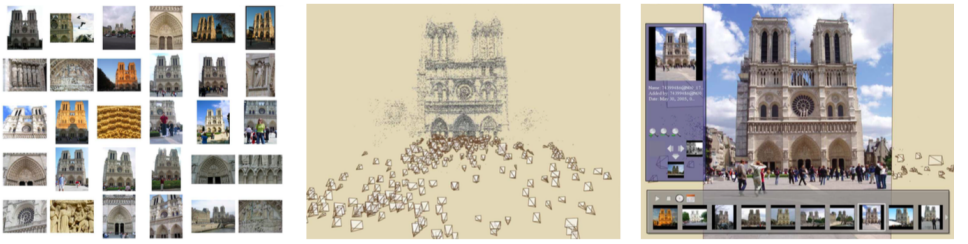


Figure 4.2. A toy example of the construction of the skeletal graph in Snavely *et al.* (2008*b*). (a) Nodes of the image graph represent images and its edges represent pairwise reconstructions (in fact the image graph is directed, and each connected pair has two directed edges; only one is shown for simplicity). Each (directed) edge (I, J) is endowed with a covariance matrix C_{IJ} representing the uncertainty in image J relative to I. (b) Relative uncertainty between the nodes P and Q is quantified by the trace of the covariance matrices (ellipses) chained up along the shortest path (arrows) between the nodes. (c) Removing an edge lengthens the shortest path, resulting in larger estimated covariance (larger ellipses). (d) The solid edges are those of the skeletal graph, while the dotted edges have been removed. The black (interior) nodes form the skeletal set, and are to be reconstructed first, while the grey (leaf) nodes are added to it using position estimation. The skeletal graph is constructed by trying to minimize the number of interior nodes, while bounding the increase in relative uncertainty between all pairs of nodes in the original graph in (a). Figure courtesy of Snavely *et al.* (2008*b*).

'bag-of-words approach' (Havlena *et al.* 2010, Section 2.1). The computation of a minimally connected dominating set for the induced (unweighted, undirected) graph, defined as a minimum-sized connected subgraph, of which each vertex of the original graph is either a member or a neighbour, is in general NP-hard. Hence, they use a polynomial-time approximation algorithm from the literature for this task (Havlena *et al.* 2010, Algorithm 1). For the 3D structure, they make use of the *atomic 3D models from image triplets* approach of Havlena, Torii, Knopp and Pajdla (2009), and grow the structure by using a prioritization queue, which orders the different tasks of either creating a new atomic 3D model or incorporating one image into a given 3D model (Havlena *et al.* 2010, Section 3). They conclude with various experimental results, including image sets captured by omnidirectional cameras, demonstrating the efficiency and accuracy of their approach.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press



(a) (b) (c)

Figure 4.3. Photo tourism system of Snavely *et al.* (2006): (a) An unstructured collection of images, *e.g.* downloaded from the Internet, as input to the system. (b) SfM step composed of camera motion estimation and 3D structure reconstruction. (c) Content exploration, such as 3D scene visualization, object-based photo browsing, geolocation from images, object annotation, *etc.* Figure courtesy of Snavely *et al.* (2006).

A notable work of an incremental nature is the *photo tourism* system of Snavely *et al.* (2006), which was also discussed in Section 1. Although the SfM part of their paper is primarily based on a rather classical, generic type of sequential incorporation of images, one by one, using the sparse bundle adjustment algorithm of Lourakis and Argyros (2009), it provides a very rich set of tools for content exploration, such as 3D scene visualization, object-based photo browsing, geolocation from images, or object annotation; see Figure 4.3 for a simplified illustration of the photo tourism system. For sequential SfM, Snavely *et al.* identify an initial pair of cameras sharing a large set of feature points, which also have a known baseline to improve the accuracy of the reconstructed initial 3D points. The selection criterion for a new camera is to have a maximal number of feature points that correspond to the already reconstructed 3D points. This iterative procedure is repeated until no remaining camera observes any reconstructed 3D point; for additional details, such as detection of outlier tracks, geo-registration, and specific algorithms used, see Snavely *et al.* (2006, Section 4). Even though the overall computational cost of the proposed SfM pipeline can be fairly high for relatively large datasets,[16] it provides an accurate alternative to SfM, and more importantly, the rich set of tools it provides for content exploration strikingly demonstrates the wide range of applicability and usefulness of SfM.

Yet another remarkable SfM approach for processing large (specifically *city-scale*) unordered datasets has been introduced by Agarwal, Snavely, Simon, Seitz and Szeliski (2009). The key paradigm they explore is the

[16] For example, it is reported to take *two weeks* to process and match the 2635 images in the Notre Dame dataset and register 597 of them.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

parallel processing of the various stages in SfM, and hence its main contributions are the identification of challenges and introduction of solutions in the parallelization of the SfM pipeline. In its multi-stage image matching method, where each stage consists of a proposal step (of possible image pairs sharing common features) and a verification step, they employ several different techniques, such as vocabulary tree-based whole-image similarity and query expansion, while simultaneously controlling the computational efficiency of the distributed framework; for details see Agarwal *et al.* (2009, Sections 2.2, 2.3). For the SfM part, they use the skeletal set method of Snavely *et al.* (2008*b*) to reduce the computational cost significantly, and they compute the initial camera motion and 3D structure for the skeletal set via the sequential SfM method of Snavely *et al.* (2006). To improve upon the computed motion and structure, they perform bundle adjustment, where, depending on the size of the problem, they choose between a preconditioned CG method (as in Agarwal *et al.* 2010) for large datasets, and an exact step LM algorithm for smaller datasets. Finally, Agarwal *et al.* (2009) demonstrate the accuracy and efficiency of their overall approach via extensive experimental results for various city-scale datasets.

Interestingly, after camera rotations have been estimated, the rest of the SfM pipeline may be reformulated as a bipartite location recovery problem. Namely, camera locations and 3D structure points form the two sets of nodes of the bipartite graph, and directions between cameras and scene points obtained from correspondences between pairs of views form the edges. This observation, coupled with fast algorithms for ShapeFit and LUD, has potentially substantial implications for real-time robotics applications, as discussed in Section 5. Hand, Lee and Voroninski (2015), in follow-up work to Hand *et al.* (2017), have proved a set of corruption-robust exact recovery theorems for ShapeFit in the bipartite setting. As obtained after the correspondence estimation step of the SfM pipeline, Hand *et al.* (2015) assume the availability of measurements (partially corrupted with adversarial noise) of camera-to-point directions $(\mathbf{t}_i - \mathbf{p}_j)/\|\mathbf{t}_i - \mathbf{p}_j\|$, where $\mathbf{t}_i \in \mathbb{R}^d$ is the ith location and $\mathbf{p}_j \in \mathbb{R}^d$ the jth structure point. Using the convex ShapeFit algorithm introduced in Hand *et al.* (2017) for the simultaneous estimation of locations and structure points, they prove a (high-dimensional) exact recovery result for a set of i.i.d. Gaussian locations and structure points with high probability if the underlying bipartite measurement graph is sampled from an Erdős–Rényi ensemble, the problem dimension d is sufficiently large, and at most a constant fraction of direction measurements involving any particular location or structure point are corrupted; for a precise statement of this main result see Hand *et al.* (2015, Theorem 1). Although this result is in the high-dimensional regime, it is the first *simultaneous* exact location and structure recovery result for partially corrupted camera-to-point direction measurements. The proofs in Hand *et al.* (2017) crucially rely on the

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

existence of sufficiently many triangles in random graphs, while any bipartite graph cannot contain a triangle. The results in Hand *et al.* (2015) establish the necessary properties for exact recovery in high dimensions from corrupted relative-direction observations, similar to Hand *et al.* (2017, Theorem 1), by relying instead on the existence of randomly distributed four-cycles in random bipartite graphs. Hand *et al.* (2015) also provide empirical evidence on simulated data supporting their theoretical findings, and Goldstein *et al.* (2016) evaluate bipartite ShapeFit on real datasets, providing evidence of robust recovery in the bipartite setting.

5. Simultaneous localization and mapping (SLAM)

Simultaneous localization and mapping (SLAM) consists in estimating a depth map of the 3D environment, as well as localizing the agent which is moving throughout this environment. SLAM is a fundamental problem in robotics, providing crucial information for autonomous navigation of cars, drones and consumer robots. Pre-built maps of the environment can be used to provide much useful information for navigation, provided that localization within those maps can be performed accurately. The correct assertion that a vehicle or robotic entity is in a previously observed location is called a loop closure. SLAM is effectively a special case of the SfM problem, in that it is specific to robotics and augmented/virtual reality applications, yet retains many of the same difficulties.

A multitude of combinations of sensing techniques have been used for SLAM, including LIDAR, inertial sensors, GPS and vision. LIDAR is a form of active sensing, which consists of measuring laser time of flight to infer depth in real time. While LIDAR to a large extent trivializes the depth estimation problem and also eases the task of loop closure, it is an expensive sensor with low resolution, and suffers performance degradation in sub-optimal weather conditions such as rain and fog. Nonetheless, LIDAR is the centrepiece of many of today's approaches to autonomous navigation, by allowing pre-mapping of urban environments and accurate loop closures in such maps, which provides information about the static part of the environment in real time. Localization can also be aided by differential GPS, which has very high accuracy but is also expensive. The holy grail of SLAM research is a purely vision-based approach, since vision sensors are cheap and ubiquitous, and provide very rich and high-resolution information about the environment. However, due to the aperture problem and various other ambiguities of the image formation process, using vision alone for SLAM is a difficult task and remains at the forefront of robotics research, in particular due to its high computational complexity. In this section we will cover state-of-the-art vision-based and visual-inertial SLAM approaches as well as some of the particular difficulties faced in robotics applications.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

5.1. New approaches to visual SLAM and their limitations

As outlined above, the SfM pipeline (and similarly a SLAM pipeline that does not rely on pre-built maps) begins by (i) establishing feature correspondences and then (ii) estimating camera orientations from relative orientations. Once camera orientations have been estimated, the task is to recover camera locations and 3D structure points, which can be done in several ways. The standard pipeline involves first solving for camera locations from relative directions between cameras, obtained from epipolar geometry and orientation estimates. Several algorithms, as described in Section 3, can be used for this step. Once camera locations have been recovered, 3D structure may be obtained via triangulation. Often, the resulting camera position and 3D structure estimates are not as accurate as desired, and there is a global step which estimates these quantities simultaneously given suitable initial data, which is called bundle adjustment (see Section 4). Standard approaches to bundle adjustment are computationally prohibitive for real-time applications. Recent progress in algorithms and numerical methods allow for fast solutions of the location recovery problem. In particular, Goldstein *et al.* (2016) have developed a numerical framework based on ADMM and tailored to SfM, which allows solution of the LUD and ShapeFit programs at 10–100 times the speed of competing approaches. This capability offers a new path towards real-time SLAM by reformulating the SfM pipeline: that is, after obtaining camera orientations, the point correspondences give relative directions between camera and 3D structure point locations. Thus, we may frame estimating camera locations and 3D structure points as a location recovery problem on a bipartite graph. ShapeFit and LUD can be applied to such a bipartite formulation, and ShapeFit has been proved to recover locations and 3D structure in this regime exactly, under suitable assumptions. Empirical results demonstrate the effectiveness of ShapeFit and LUD in the bipartite regime. Bundle adjustment can potentially be avoided if the output of this pipeline is accurate enough, which would enable usage in real-time robotics applications. The entire process depends on the accuracy of the estimated camera orientations, and in practice, accelerometers are relatively accurate at providing such estimates.

The main difficulty in vision-based SfM systems is the abundance of outliers in correspondence estimates, which are due to the highly ambiguous nature of local photometric information due to illumination changes, occlusions, specularities, repetitive structures and objects that move independently of the scene. In offline applications such as photo tourism (Snavely *et al.* 2006), it is possible to lower the outlier rates in correspondences to still high but acceptable levels (say, 40 %) by using RANSAC (Bolles and Fischler 1981), which is a probabilistic technique that randomly samples a set of observations, fits a model, and validates that model by how many of

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

the other observations it approximately fits. In the case of SfM, RANSAC exploits the eight-point algorithm, which, given a set of eight accurately estimated correspondences between a pair of views, estimates the relative orientation and direction between camera locations. Having obtained a candidate relative orientation and translation between camera views, it is possible to validate correspondences. While effective, RANSAC is ultimately a combinatorial procedure, and thus it is only tractable to run to completion in offline applications such as photo tourism. In real-time settings, tradeoffs have to be made with the use of RANSAC, at times leaving the fraction of outliers in correspondences as high as 80 or 90 %. These very high outlier ratios make SfM intractable via methods such as ShapeFit and LUD, and extra information needs to be taken into account to enable efficient SfM in such regimes.

5.2. Visual-inertial navigation systems

To alleviate the issue outlined above, it is common practice to combine visual information with input from inertial sensors (accelerometers and gyrometers), to provide estimates of 3D orientation and position of the sensor platform, also called visual-inertial navigation (VINS). Such an approach is prevalent in applications such as augmented and virtual reality. It is common to model the visual inertial navigation problem as the evolution of a dynamical system, the states of which signify 3D positions of the sensor platform as an element of the Euclidean group of motions SE(3), which produces sensor measurements as outputs, up to some uncertainty. It is a natural first step to analyse the observability of such a system, under assumptions on model parameters such as calibration coefficients, the sensor biases, and a generative 'noisy' input which drives the evolution of the system. While most models treat sensor bias as noise independent of other states, this assumption is frequently violated in practice. Hernandez, Tsotsos and Soatto (2015) show that in the absence of such an independence assumption, the resulting model is not observable. The problem is then recast as one of sensitivity analysis, with results of Hernandez *et al.* (2015) deriving an explicit characterization of the set of trajectories indistinguishable under the observed measurements, as a function of the properties of the sensor and sufficient excitation conditions (the properties of motion of the sensor platform), which can be used for analysis and validation purposes.

Tsotsos, Chiuso and Soatto (2015) consider the VINS task while specifically focusing on the handling of outliers. This task falls under the umbrella of robust statistics, and most VINS systems employ some form of outlier/inlier testing. Tsotsos *et al.* (2015) derive the optimal outlier discriminant, showing that it is intractable, motivating state augmentation with a delay-line in the model and examining various analytical and

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

empirical approximations. Compared to decoupled systems, that is, those in which position estimates are computed from visual and inertial measurements individually and then fused, the approach in Tsotsos *et al.* (2015) is guaranteed to stay within a bounded set of the state trajectories.

5.3. Application-specific difficulties

Structure from motion is a well-developed subfield of computer vision, and has led to a number of successful commercial applications. However, the distribution of camera locations strongly affects the performance of SfM and SLAM algorithms, and is dictated by the application at hand. For instance, photo tourism has been successfully commercialized and is one of the easier cases of SfM, because camera locations tend to be well-distributed in space, thus providing sufficiently diverse views of the scene which can be used to infer depth information. On the other hand, the application of autonomous driving requires handling scenarios in which the camera moves in essentially a straight line for long periods of time. There are many issues with this special case.

(i) The most informative features (the ones that undergo the most motion) are in the periphery and quickly move out of the field of view, which explains the popularity of omnidirectional cameras for such applications.

(ii) There exist many local minima in the least-squares reprojection error, which is the basis for many algorithms, each corresponding to various ambiguities such as Necker-reversal, plane-translation and bas-relief.

(iii) Considering the worst case of moving exactly in parallel with the optical axis, it is evident that depth at the centre pixel of the image is unrecoverable, and this phenomenon induces many local minima around the true direction of translation.

Semi-global approaches aim to overcome these issues, but they are too computationally intensive for real-time applications. One may ponder whether these ambiguities are inherent to the problem or are a by-product of the least-squares reprojection functional. Vedaldi, Guidi and Soatto (2007) show that the answer is at least partly the latter, by demonstrating that enforcing a bound on the depth of the scene makes the reproduction error continuous. However, local minima are still present, as evidenced by empirical results.

5.4. Direct versus feature-based methods

One challenge of applications, such as robotics and augmented/virtual reality, is transitioning between differently scaled scenes, for instance from indoors to outdoors. Sensors which provide scale measurements, such as stereo

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

or LIDAR, have a limited scale range in which they perform well. For instance, depth that far exceeds the baseline of a stereo rig is difficult to infer via stereo without sufficiently high image resolution, which in turn impacts real-time computational tractability. Meanwhile, monocular SLAM is immune to this issue since it is inherently scale-invariant. At the same time, due to the fact that absolute scale cannot be inferred in monocular SLAM, scale estimates are subject to drift over time.

SfM approaches fall into two categories: feature-based and direct methods. Feature-based approaches, as described above, decouple the problem into two steps: (i) features are extracted from images, and (ii) camera position and 3D scene structure are computed based on (i) only. Naturally, the efficacy of such approaches depends crucially on the admissible set of features, which in many approaches is limited to a sparse set of keypoints in each image. In contrast, direct methods for visual odometry attempt to utilize all the photometric information in images by optimizing for camera position and 3D structure directly. While direct methods have an information-theoretic advantage over feature-based ones, they are typically more computationally intensive. A method recently proposed by Engel, Schöps and Cremers (2014), called large-scale direct monocular SLAM (LSD-SLAM), is capable of building consistent large-scale maps of a 3D scene in real time, by estimating semi-dense depth maps via filtering and direct image alignment. The map of the scene is represented as a collection of keyframes with 3D similarity transformations as constraints between positions at keyframes. This representation allows scale changes to be taken into account naturally, and corrects accumulated drift.

For alternative methods and further discussion of SLAM, we refer the reader to Aulinas, Petillot, Salvi and Lladó (2008), Fuentes-Pacheco, Ruiz-Ascencio and Rendón-Mancha (2015), Younes, Asmar and Shammas (2016), and the references therein.

6. Other topics

This section briefly discusses some of the additional topics in the SfM literature, not considered in previous sections. These topics include methods for feature extraction and matching, alternative core measurement types and camera models, methods aiming to handle symmetries and ambiguities in images, and widely utilized sources of data and SfM software.

6.1. Feature description and matching

Although the processes of detection, description and matching of image features are usually considered to precede the SfM steps, here we touch upon

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

some of the most frequently used methods for these tasks.[17] One of these methods is *scale-invariant feature transform* (SIFT) (Lowe 1999, 2004), which has been extensively used as a building block for many of the existing corresponding point-based SfM methods; see, for example, Snavely *et al.* (2006), Wu (2011), Crandall *et al.* (2011), Özyeşil *et al.* (2015), Özyeşil and Singer (2015) and Arie-Nachimson *et al.* (2012). In essence, SIFT transforms an input image into a collection of *local features* (*e.g.* represented as vectors in some Euclidean space), which are invariant to scaling and rotation of the underlying image, and partially invariant to 3D camera viewpoint and to changes in illumination. The detection and description stages of SIFT are mainly composed of four steps. In the first step, potential keypoints, which are invariant to scale and rotation, are determined via detection of extrema in a *scale-space representation* of the image, using difference of Gaussians (DoG). The scale-space representation of an image, depicted in Figure 6.1, is a parametrized collection of smoothed images, where the single parameter corresponds to the size of a kernel used to filter out fine structures in the image. SIFT computes scale-space representations, using DoG kernels, repeatedly for down-sampled copies of the original image, and then identifies locally extremal points on these images by simply comparing the pixel values with the nearest neighbours in the current and adjacent scales. After identifying these candidate points, that is, the extrema in the scale-space representation, the next step is to compute (approximately) location, scale and the ratio of principal curvatures (in order to identify and reject the candidates having low contrast or that are poorly localized on an edge: for details see Lowe 2004, Section 4). The third step aims to achieve invariance to rotation in the descriptor by assigning an orientation (to be phased out in the representation of the descriptor) to each keypoint based on local image gradient directions. At this stage, SIFT has assigned a scale, an orientation and a location to each remaining keypoint in the image, and the next step involves the computation of a descriptor for (the local regions around) each keypoint, while maintaining invariance to changes in illumination and 3D view-point, as much as possible. By operating on the image having a scale closest to that of a keypoint to fix a level of Gaussian blur, SIFT first samples gradient magnitudes and orientations around the keypoint location, and rotates the coordinates of the descriptor and the gradient orientations relative to the keypoint orientation (computed in the previous step) in order to achieve rotational invariance; see Figure 6.2 for a depiction. The

[17] The literature on the (classical) problem of *detection* of local feature points (sometimes also referred to as feature extraction) is a vast accumulation of works of roughly the last five decades, including techniques for detection of edges, corners, blobs and regions, segmentation-based methods and machine-learning-based methods, which we do not intend to discuss here in detail. For an exposition of some of these methods, see for example Tuytelaars and Mikolajczyk (2008).

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

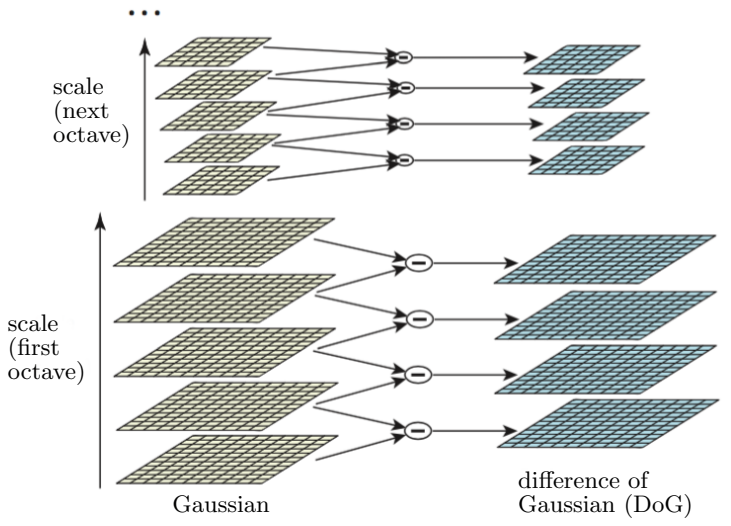
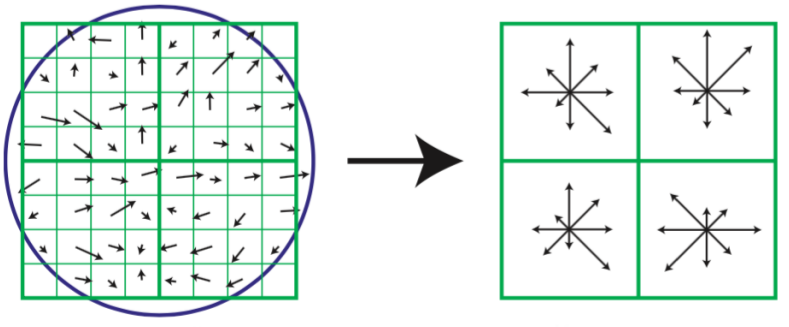


Figure 6.1. The scale-space representation of an image in Lowe (2004) using difference of Gaussians (DoG). For each octave of the scale space, the original image is first convolved with Gaussians to obtain the set of scale space images (on the left), and then adjacent Gaussian images are subtracted to produce the DoG images (on the right). After each octave, the Gaussian image is down-sampled by a factor of 2, and the process is repeated. Figure courtesy of Lowe (2004).



(a) image gradients (b) keypoint descriptor

Figure 6.2. To obtain a keypoint descriptor, first the gradient (magnitude and orientation) at each image sample point in a region around the keypoint location is computed (a) and then scaled by a Gaussian window (represented by the overlaid circle). These samples are then collected into orientation histograms representing the contents over 4×4 subregions (b), where the length of each arrow is given by the sum of the gradient magnitudes quantized to that direction within the region. For simplicity, the figure shows a 2×2 descriptor array computed from an 8×8 set of samples, whereas 4×4 descriptors computed from 16×16 sample arrays are used in practice. Figure courtesy of Lowe (2004).

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

descriptor is given as a collection of orientation histograms (having 8 bin values corresponding to a weighted sum of magnitudes of gradients, whose directions are close to one of 8 quantized directions) of 4×4 subregions around the keypoint location. The resulting 128-dimensional descriptor is then normalized to have unit norm to attain invariance to homogeneous changes in illumination. To also improve invariance to non-linear changes in illumination, the entries of this unit norm feature vector are thresholded (to compensate for large relative changes in gradient magnitudes), and are normalized to have unit norm once more. In the last stage of matching the extracted feature vectors (*i.e.* descriptors) between images, the Euclidean distance between feature vectors is used to compute nearest neighbours. Matched points are then identified to be those closer to their nearest neighbours than a fixed threshold.

Several works in the literature build upon the SIFT method. One such method, called PCA-SIFT (Ke and Sukthankar 2004), is different from SIFT only in its computation of descriptors from local neighbourhoods around keypoints. Similar to SIFT, for a given keypoint with assigned scale and orientation, PCA-SIFT extracts a 41×41 patch around the keypoint (in the image corresponding to the assigned scale) and rotates the patch using the assigned orientation to attain rotational invariance. We then compute a $2 \times 39 \times 39 = 3042$-dimensional vector of concatenated vertical and horizontal gradients for the 41×41 patch centred at the keypoint. As in SIFT, this vector is normalized to have unit length, to improve invariance to changes in illumination. A low-dimensional representation of this unit norm vector is the desired descriptor. To compute this low-dimensional representation, PCA-SIFT initially computes (offline) an *eigenspace* spanned by the top n eigenvectors (say, $n = 20$) of the covariance matrix of a large number (taken to be 21 000 in Ke and Sukthankar 2004) of 3042-dimensional vectors corresponding to patches extracted from a diverse collection of images. The descriptor for each keypoint is then computed to be the orthogonal projection of the 3042-dimensional vector corresponding to its local patch onto the n-dimensional precomputed eigenspace. The computed descriptors are matched between images via the same method in SIFT. Ke and Sukthankar (2004) also demonstrate empirically that PCA-SIFT is resilient to various deformations and noise in the images; see also Mikolajczyk and Schmid (2005) for a comparative study.

An alternative variation of SIFT, called the *gradient location and orientation histogram* (GLOH), was introduced by Mikolajczyk and Schmid (2005).[18] In essence, GLOH uses a similar approach to PCA-SIFT to compute descriptors. The difference is mainly in its representation of the

[18] The main focus of Mikolajczyk and Schmid (2005) is actually the performance evaluation of various descriptors, based on local regions, with respect to various factors

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

locations of the gradients in the neighbourhood of the keypoints. Instead of dividing the local neighbourhood (in the image having the assigned scale) of the keypoint using a rectangular grid, GLOH uses *logarithmic polar coordinates* with three bins in the radial component and eight bins in the angular component (without dividing the central bin into angular components), resulting in a total of 17 locations. Further, using a quantization of the gradient orientations into 16 bins, GLOH computes a histogram of $16 \times 17 = 272$ bins. Similar to PCA-SIFT, this histogram is represented in a 128-dimensional eigenspace computed from a collection of 47 000 image patches (via PCA of the corresponding covariance matrix). Mikolajczyk and Schmid (2005) also provide extensive empirical results demonstrating the performance of GLOH relative to nine different methods including SIFT and PCA-SIFT.

For details and performance evaluation of some of the other approaches in the literature (including the computationally efficient *speeded up robust features* (SURF) method of Bay, Tuytelaars and Van Gool 2006) for detection and description of local features, we refer the reader to Gauglitz, Höllerer and Turk (2011).

6.2. Symmetries and ambiguities in images

A fundamental difficulty for SfM methods based on *local features* in images is the presence of distinct objects in the 3D structure that look similar. These similarities may arise from (relatively local) symmetries present in the 3D structure (*e.g.* four similar sides of a clock tower, spherically symmetric dome of a cathedral, *etc.*) or repeated structures in the scene (*e.g.* multiple buildings with similar windows). Such *ambiguous* structures may cause unrelated features in different images to be incorrectly matched, and hence result in major errors in structure recovery, for example partial structure estimates incorrectly superimposed on top of each other, *phantom* structures floating in the scene, or broken structural integrity due to clustering. As a result, for scenes with ambiguous structures, special care must be taken to disentangle incorrect correspondences of local features. In this section we briefly discuss some of the recent progress in the literature aiming to handle such ambiguities.

A disambiguation method based on a global integration of geometric relations between images in a Bayesian framework was introduced by Zach, Klopschitz and Pollefeys (2010). In order to detect and remove inconsistently matched features, the main idea employed by Zach *et al.* (2010) is to utilize consistency relations induced by (invertible) pairwise geometric transformations (*e.g.* homographies, relative camera rotations, similarity

influencing their performance. Hence, for details of some of the other methods for feature extraction, we refer the reader to their paper.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

transformations between local 3D reconstructions) along cycles of related images, that is, images presumably sharing measurements of similar parts of the scene. More concretely, Zach *et al.* (2010) consider the underlying graph of related images (having edges endowed with the pairwise transformations) and propose detecting inconsistent edges by using the fact that (in the noiseless case) chaining all pairwise transformations along any cycle must result in the identity transformation. In order to identify inconsistent edges from cycle consistencies, they use a Bayesian inference approach, where the inconsistency of each edge is represented using a latent binary variable. The objective is then to maximize the joint posterior probability of these latent variables given inconsistencies of each cycle[19] they belong to, that is, given some measure of the deviations, from the identity transformation, of the transformations corresponding to these variables chained along cycles. In order to obtain an approximate (more precisely, a locally optimal) solution to this difficult problem, they use *loopy belief propagation*. Also, to obtain quality certificates, they propose approximating the optimization of the log-likelihood function corresponding to the posterior probability using a (standard) convex relaxation approach (for details see Zach *et al.* 2010, Section 3). Finally, they provide experimental evidence of the improvements gained by using their overall disambiguation approach.

Relying solely on pairwise geometric relations for disambiguation may be insufficient for scenes involving large duplicate structures, which result in large, self-consistent sets of incorrectly matched feature pairs. A relatively recent method studied by Roberts, Sinha, Szeliski, Steedly and Szeliski (2011), which fuses an *expectation maximization* (EM) algorithm to estimate camera positions and a sampling technique (similar to RANSAC: Bolles and Fischler 1981) to identify erroneous matches, aims to disambiguate such cases. In designing their method, the basic argument of Roberts *et al.* (2011) is that merely obtaining a geometrically consistent set of edges (as in Zach *et al.* 2010) may be insufficient for disambiguation, since such an approach inherently assumes the *statistical independence of erroneous matches*. For camera motion estimation, they use a Gaussian mixture model (involving hidden binary variables to represent correct/erroneous matches as in Zach *et al.* 2010) to infer the erroneous matches that are *geometrically inconsistent* with the majority of the others. The inference employs an iterative EM algorithm applied first to relative camera rotations, to obtain camera orientations in SO(3), and then to (a spanning tree of) triplet 3D reconstructions, to obtain a camera motion estimate (for details see Roberts *et al.* 2011,

[19] Since, in principle, the underlying graph can have an exponentially large number of cycles, Zach *et al.* (2010) restrict the number of inspected cycles to the set of all triangles together with cycles of maximum length six induced by a set of cycle bases (specifically, spanning tree bases); for further details see Zach *et al.* (2010, Section 3).

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

Section 3). Since such an approach applied to the whole pairwise measurements with an underlying assumption of independence is argued to be ineffective for disambiguation, they use a specifically designed *random sampling* of spanning trees of the pairwise measurement graphs, which constitutes the main contribution of the paper. To increase the probability of sampling desired spanning trees (*i.e.* minimal sets of pairwise measurements with possibly higher accuracy), they use an efficient sampling method from the literature to generate random spanning trees according to a specific distribution of spanning trees. The distribution is based on a set of edge weights, where the probability of each tree is proportional to the product of its edge weights. To determine the edge weights, Roberts *et al.* (2011) consider two sources of information from the images.

(i) The ratio of the number of matched features between an image pair to the number of all features (in one of the images in the pair) matched to any other image (adjusted by the distance of the missing correspondences to the matched features between the image pair).

(ii) In the presence of *time stamps* for the cameras, the ratio between the time difference of the matched image pair and the smallest time difference of any matched image pair including one of the images in the pair.

For each sampled minimal hypothesis, Roberts *et al.* (2011) compute a set of matches consistent with the hypothesis and update the estimated camera motion (using the EM approach discussed above, while fixing the edges of the sampled spanning tree to be correct edges), and select the hypothesis with the highest score to obtain the final camera motion estimate; for the numerical details, see Roberts *et al.* (2011, Section 5). A final bundle adjustment is applied, only to the matched pairs inferred as correct, to obtain the refined camera motion and the 3D reconstruction. Experimental results provided in Roberts *et al.* (2011), using relatively small sets of highly similar images, demonstrate improvements obtained via the disambiguation approach, and also suggest an increase in computational time required to obtain the final results. For example, disambiguation steps for the CUP dataset of 64 images and the BLDG dataset of 76 images take 44 and 90 minutes, respectively.

Another disambiguation method based on the optimization of a measure of missing image projections of potential 3D structures is introduced in Jiang, Tan and Cheong (2012). To quantify the quality of a 3D reconstruction, Jiang *et al.* (2012) employ the fact that a 3D point has similar appearances in images in which it is captured. A SIFT descriptor for a 3D point is defined in Jiang *et al.* (2012) to be the SIFT descriptor of the corresponding feature point in its *foremost parallel image* (determined by using an approximate surface normal computed for the 3D point). Using this

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

definition, one can hope to identify the visibility of a 3D point in an image by matching its SIFT descriptor (as defined above) to the SIFT descriptor of its projection on the image. Hence, the quality of a 3D reconstruction can be quantified by measuring the average probability $P_{\text{miss}}(p, i)$ (averaged over images and 3D points) for the points p that are missing in the images i, that is, the average probability that the SIFT descriptors of its 3D points do not match with those of their image projections (see Jiang *et al.* 2012, Section 3 for the parametric assumptions they make in order to compute $P_{\text{miss}}(p, i)$). To minimize the average $P_{\text{miss}}(p, i)$, they first compute pairwise 3D reconstructions and relative camera positions. The optimization is performed on spanning trees of the match graph (induced by edges corresponding to camera pairs with a computed relative position), where they employ various heuristics in each iteration to improve the efficiency of its search (for details see Jiang *et al.* 2012, Section 5). They conclude with experimental results demonstrating the accuracy, efficiency and limitations of their disambiguation approach.

Most of the disambiguation methods in the literature are designed to operate on relatively small sets of images created to exemplify relatively difficult SfM disambiguation problems. However, the level of difficulty and the efficiency requirements of large unordered image collections can be significantly different. A recent method, based purely on a graph-theoretic approach, designed to disambiguate such large datasets is studied in Wilson and Snavely (2013). The main concept in Wilson and Snavely (2013) is the bipartite *visibility graph* $G = (I, T, E)$, where nodes are the images I and the tracks T (*i.e.* sets of matched features across several images), and an edge $(i, t) \in E$ exists if the track t includes a feature in image i. The main idea in Wilson and Snavely (2013) is to identify and remove *bad tracks* in T (*i.e.* tracks which correspond to more than one 3D point, possibly due to ambiguities in the scene), which tend to form *bridges* between unrelated parts of the scene (and hence between the corresponding images and tracks of these scenes in G). To quantify the quality of a track t (*i.e.* its dissimilarity to a bridge-type node), they use the *bipartite local clustering coefficient* of t defined to be the ratio of the number of closed 4-paths centred at t to the number of all 4-paths centred at t. Also, in order to eliminate the sensitivity of bipartite local clustering coefficients to the variations in the sizes of local clusters in the visibility graph (*e.g.* resulting from largely varying popularities of different parts of a tourist scene), they compute a covering subgraph G' of the visibility graph G with improved uniformity; see Wilson and Snavely (2013, Section 5) for technical details and specific parameters. The bad tracks of G' are determined to be those having bipartite local clustering coefficients less than a threshold, and are removed from G' (the same procedure is also applied to G). The remaining (good) tracks and their corresponding images are used in an incremental SfM pipeline from

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

Agarwal *et al.* (2009) to obtain an initial (skeletal) reconstruction, and then remaining tracks and images in G are used to enrich the reconstruction. Although this purely graph-theoretic approach may be insufficient for difficult SfM disambiguation problems, Wilson and Snavely (2013) provide extensive empirical results (for large Internet photo collections) demonstrating the accuracy and the high computational efficiency of their disambiguation technique.

For alternative approaches aiming to resolve the difficulties in SfM for symmetric and ambiguous scenes, we refer the reader to other sources, for example Cohen, Zach, Sinha and Pollefeys (2012), Snavely, Seitz and Szeliski (2008*a*), Schaffalitzky and Zisserman (2002), Martinec and Pajdla (2007) and Pachauri, Kondor, Sargur and Singh (2014).

6.3. Alternative features and camera models

Our discussion in the previous sections has mainly focused on SfM techniques based on *point features* extracted from images formed by *perspective projection*. In this section we touch upon some of the SfM methods in the literature designed for alternative features and camera models.

A relatively popular alternative to using correspondences of point features for SfM is to use corresponding *lines* across multiple images. A solution to line-based SfM is composed of a camera motion estimate and 3D line reconstructions, which may be more suitable, as compared to 3D point reconstructions, for applications such as modelling of urban scenes or augmented reality. A crucial point for line-based SfM methods is the mathematical representation of the four-dimensional set of 3D lines and a corresponding model for their measurements in images, since the usefulness of an existing representation may depend on the particular problem considered. Relatively early factorization methods for line-based SfM (see *e.g.* Martinec and Pajdla 2003, Quan and Kanade 1997) are usually constrained in their accuracy and applicability to scenes with occlusions. A direct reprojection error minimization approach (susceptible to local minima) is proposed in Taylor and Kriegman (1995), where a line l is represented as $l = (\mathbf{d}, \mathbf{v})$ using a *closest point* $\mathbf{d}$ *to the origin* and a unit norm *direction* $\mathbf{v}$.[20] Two different representations of lines are used in Bartoli and Sturm (2005) for the two problems of triangulation of matched features and bundle adjustment. For triangulation, Plücker coordinates are used to represent lines, mainly due to the resulting linearity of the reprojected lines as functions of the 3D lines. For a pair of 3D points $\mathbf{x} = [\bar{\mathbf{x}}\ x_0]$ and $\mathbf{y} = [\bar{\mathbf{y}}\ y_0]$ in homogeneous coordinates, the six-dimensional vector $[\mathbf{a}\ \mathbf{b}]$ representing the Plücker

[20] Note that this representation is a two-to-one correspondence, between the four-dimensional manifold of parameters $(\mathbf{d}, \mathbf{v})$ satisfying $\|\mathbf{v}\| = 1, \mathbf{v}^T\mathbf{d} = 0$ and the set of 3D lines, since $(\mathbf{d}, \mathbf{v})$ and $(\mathbf{d}, -\mathbf{v})$ correspond to the same line.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

coordinates of the line through the points $\mathbf{x}$ and $\mathbf{y}$ satisfies $\mathbf{a} = \bar{\mathbf{x}} \times \bar{\mathbf{y}}$, $\mathbf{b} = x_0\bar{\mathbf{y}} - y_0\bar{\mathbf{x}}$ and $\mathbf{a}^T\mathbf{b} = 0$. A linear algorithm and two quasi-linear algorithms (combined with a final step of rounding to the Plücker coordinates) for triangulation of line features are introduced in Bartoli and Sturm (2005). For bundle adjustment, Bartoli and Sturm (2005) employ the *orthonormal representation* of 3D lines $l = (U, W)$, for $U \in \mathrm{SO}(3)$ and $W \in \mathrm{SO}(2)$; see Bartoli and Sturm (2005, Section 5.3) for details of the computation of U, W. This is mainly due to the completeness and minimal updatability (*i.e.* only four parameters are required to update the representation) of the orthonormal representation, which also lacks gauge freedoms and internal constraints, making it a numerically stable choice for the non-linear, unconstrained bundle adjustment step. Bartoli and Sturm (2005) also provide experimental results to evaluate the performance of their overall approach. An alternative method specifically designed for line-based reconstruction of 3D urban scenes has been introduced by Schindler, Krishnamurthy and Dellaert (2006). Their main idea is to classify the lines in the images into different orientation classes (*e.g.* horizontal lines, vertical lines) using an expectation-maximization approach to estimate 3D vanishing point assignments (representing vanishing directions, at least three of which are assumed to be mutually orthogonal) for every pixel in the images. They argue that such a classification improves the accuracy of matching of lines between images (since only lines in separate classes are not allowed to be matched), and also increases the efficiency and accuracy of the 3D structure reconstruction by reducing the degrees of freedom of 3D lines during optimization. To represent a 3D line, they consider an embedding of the four-dimensional manifold of lines in the 11-dimensional space $\mathrm{SO}(3) \times \mathbb{R}^2$, where the $\mathrm{SO}(3)$ and $\mathbb{R}^2$ parts essentially represent plane of orthogonality of the line and the point of intersection of the line and the plane of orthogonality, respectively. In order to overcome the singularities induced by this over-parametrization during non-linear optimization for 3D structure recovery, Schindler *et al.* (2006) use local parametrizations around each point in $\mathrm{SO}(3) \times \mathbb{R}^2$ (encoded by the exponential map of the Lie group $\mathrm{SO}(3)$) with varying degrees of freedom for each class of lines: for example, horizontal lines have three degrees of freedom whereas vertical lines have only two degrees of freedom. They also give experimental results providing the 3D line reconstructions (for relatively small image sets). For additional reading on urban scene reconstruction, we refer the reader to Musialski *et al.* (2013), Pollefeys *et al.* (2008) and the references therein.

There exist several works in the SfM literature based on relatively uncommon camera models that aim to solve the SfM problem with specific applications in mind or improve the accuracy of existing 3D recovery techniques by using different measurement apparatus (*i.e.* different cameras). One such study is that of Chang and Hebert (2000), who consider the SfM

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

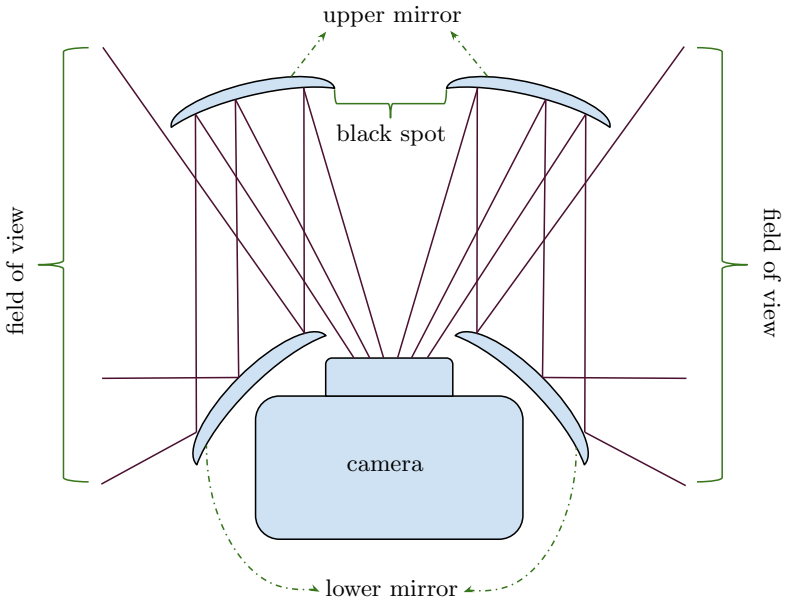


Figure 6.3. (Simplified) diagram of a 2D slice of an omnidirectional camera with two mirrors (upper and lower).

problem for *omnidirectional cameras*, that is, cameras with large visual coverage, possibly having a 360° field of view in the horizontal plane, or one that approximately covers the entire visual sphere: see Figure 6.3 for a simplified diagram. Chang and Hebert (2000) derived the epipolar constraints for catadioptric omnidirectional cameras. To estimate the pairwise camera motions they employed a two-step minimization of the error in these constraints, composed of an initial (classical) estimation of the relative motion parameters from the essential matrix estimates, together with direct minimization of the epipolar errors with respect to the relative motion parameters. The solution of the first step is taken as initial point. They also provided a brief analysis to quantify the errors in the estimated motion parameters induced by measurement uncertainties, along with experimental results demonstrating (occasional) improvements in accuracy obtained by using omnidirectional cameras, instead of conventional ones. A more systematic and detailed study of SfM from variants of omnidirectional cameras (*i.e.* fish-eye lenses and catadioptric cameras with conical mirrors) has been provided by Micusik and Pajdla (2006), covering two-view geometry estimation, camera calibration and 3D reconstruction problems. Micusik and Pajdla (2006) introduce different parametric measurement models for para-catadioptric cameras and fish-eye lenses, and solve the corresponding camera auto-calibration problems by minimizing an angular error measure

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

via *polynomial eigenvalue* computation. For other works on SfM in the omnidirectional setup, see Gluckman and Nayar (1998), Shakernia, Vidal and Sastry (2003), Kannala and Brandt (2006) and Aliaga (2001), for example, and the references therein. An alternative approach in the literature is to use *generic camera models*, representing cameras as sets of projection rays, to incorporate various kinds of cameras in a unified framework, for example cameras with radial distortion, non-central catadioptric cameras, pinhole cameras or stereo rig cameras. Interestingly, such a generic SfM framework allows simultaneous processing of images captured by different camera types. The main approach to motion estimation in the generic setting is to assign projection rays to points of interest in images, by using camera calibration information, and to minimize the error in the *generalized epipolar constraints* (also known as the *Pless equations*) written in terms of the corresponding projection rays (*e.g.* represented using Plücker coordinates). Further, 3D structure points can be estimated using geometric relations between the projection rays and the 3D points of interest; see Ramalingam, Lodha and Sturm (2006, Sections 3 and 4), for example. For experimental results and technical details of these generic SfM methods, we refer the reader to Ramalingam *et al.* (2006), Mouragnon, Lhuillier, Dhome, Dekeyser and Sayd (2009), Kannala and Brandt (2006) and the references therein.

6.4. Popular software packages and data sources

For most of the existing SfM methods, the most computationally involved part is the bundle adjustment step. As a result, efficient and accurate implementations for bundle adjustment are crucial for high-quality SfM solutions. One of the most popular software packages for bundle adjustment is the *Bundler* package of Snavely *et al.* (2006). Essentially, Bundler is based on an incremental application of a modified version of the *sparse bundle adjustment* (SBA) package of Lourakis and Argyros (2009), and, being an incremental method, it incorporates a few images at a time to produce a camera motion and a 3D structure estimate from a set of images, image features and image matches provided as input. In general, for large, unordered, weakly connected sets of images, incremental methods may suffer from accumulation of estimation errors, which result in large deformations in the structure estimate. Further, several intermediate applications of bundle adjustment may hinder the overall computational efficiency. Nevertheless, the Bundler package has been empirically shown to produce accurate solutions that are usually considered as reference solutions for experimental comparisons (see *e.g.* Özyeşil and Singer 2015, Wilson and Snavely 2014). As the name suggests, the efficiency of the crucial SBA method of Lourakis and Argyros (2009) at the core of Bundler essentially arises from exploitation of

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

the sparsity of the Jacobian in the normal equations (4.4) of the Levenberg–Marquardt iterations. We note that, as a standalone package independent of Bundler, SBA is an efficient and stable alternative for final refinements of initial motion and structure estimates, that is, for a final bundle adjustment step.

Wu, Agarwal, Curless and Seitz (2011) have introduced a highly efficient multicore CPU and GPU implementation of the preconditioned conjugate gradient (CG) solver for bundle adjustment (similar to Agarwal *et al.* 2010; see Section 4), called *multicore bundle adjustment*. The multicore bundle adjustment methodology is reported to maintain high accuracy (*i.e.* comparable to alternatives) while requiring less memory usage and providing runtime improvements of up to ten times for the CPU, and up to 30 times for the GPU implementations. This improvement in efficiency is solely based on the optimized usage of parallel computational resources (*i.e.* multicore CPUs and GPUs). An incremental SfM package, called *VisualSfM*, based on the multicore bundle adjustment algorithm of Wu *et al.* (2011), was later provided by Wu (2011). Endowed with a very useful graphical user interface (GUI) that allows explicit visualization of several different stages of SfM, VisualSfM can be considered to be an efficient and relatively stable incremental SfM solver (albeit possessing the problems inherent in the incremental approach). Further, by the integration of the GPU implementation of SIFT given in Wu (2007), VisualSfM can be used as an end-to-end SfM engine taking as input a set of images and producing as output a motion and a 3D structure estimate (however, such a *black-box* usage may result in degradation of accuracy in the final estimates).

Estimation of a 3D structure in SfM usually refers to the computation of a *point cloud* (also known as the *sparse structure*) corresponding to points from the stationary scene. However, we might also need a *dense structure*, for visualization or 3D printing, for example, corresponding to a relatively continuous representation of the corresponding scene, in the form of a dense set of rectangular patches covering the surfaces visible in the input images; see Figure 6.4 for an example. This problem is known as *multiview stereopsis*. An accurate algorithm for this purpose, namely *patch-based multiview stereo software* (PMVS), which is based on a three-step procedure of matching, expansion and filtering of patches of keypoints,[21] is given in Furukawa and Ponce (2010*a*, 2010*b*). When provided with accurate initial camera motion and calibration estimates, PMVS produces high-quality, well-connected dense 3D reconstructions. However, dense reconstruction can be

[21] Here we do not provide any detailed description of this particular algorithm for the optional procedure of dense 3D reconstruction: see Furukawa and Ponce (2010*a*, 2010*b*) for details.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press



(a) (b)

Figure 6.4. Three-dimensional structures for the Montreal Notre-Dame dataset from Wilson and Snavely (2014). (a) Sparse 3D structure obtained by using the method of Özyeşil and Singer (2015) for camera motion estimation and the PBA package of Wu *et al.* (2011) for final refinement. (b) A snapshot of the dense 3D reconstruction obtained by using the PMVS package of Furukawa and Ponce (2010*a*, 2010*b*) with the motion estimate of Özyeşil and Singer (2015).

computationally very expensive for large sets of images. To tame this computational inefficiency, Furukawa, Curless, Seitz and Szeliski (2010*a*, 2010*b*) propose a highly scalable (to tens of thousands of images) distributed approach based on the decomposition of the images into overlapping sets to be processed in parallel, that is, to produce local dense 3D structures using PMVS, for example. The emerging dense structures are then merged to produce the final dense 3D estimate. The empirical results of Furukawa, Curless, Seitz and Szeliski (2010*b*) demonstrate the high accuracy and improved efficiency of this distributed approach. We also note that the software packages PMVS and CMVS are integrated into the VisualSfM package.

A popular open-source software library consisting of several hundreds of computer vision algorithms, including those for the SfM problem, is *OpenCV* (Open Source Computer Vision Library).[22] The SfM module of the OpenCV library provides a rich selection of algorithms for various stages of the problem ranging from fundamental and essential matrix estimation, pairwise camera pose estimation to triangulation and 3D structure reconstruction. Being a well documented and supported library, OpenCV is particularly valuable for relatively well-known algorithms frequently used by existing SfM solvers.

[22] The homepage of the OpenCV library is http://opencv.org.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

An alternative open-source library providing a complete SfM pipeline, from feature detection and matching to 3D structure reconstruction, is the openMVG library by Moulon *et al.* The well documented and supported openMVG library provides efficient implementations of multiple algorithms for each stage of the SfM problem, and also includes accurate solvers for classical methods (*e.g.* the eight- or five-point methods to estimate essential matrices) used in various stages of the problem.

Another extensive software library providing a complete SfM pipeline, also including the dense 3D reconstruction stage, is the *COLMAP* library of Schönberger and Frahm (2016) and Schönberger, Zheng, Frahm and Pollefeys (2016).[23] The COLMAP library includes well documented implementations of several methods in the literature for each stage of the SfM problem. COLMAP also supports various camera models and allows a rich selection of parameters for various stages. Additionally, COLMAP includes a graphical user interface that allows easy access to most of its functionality and a real-time visualization of the reconstruction process.

A recent end-to-end library, called the *Theia Vision Library*, which incorporates efficient, reliable and supported implementations of various alternative methods for each stage of SfM, is available in Sweeney (2016). Theia provides implementations of various methods and useful routines for image handling, feature detection, description and matching, relative pairwise position estimation, camera modelling, global camera orientation and location estimation, initial structure estimation, incremental SfM, motion and structure refinement using bundle adjustment, and gauge symmetry computation for performance evaluation. Theia also provides extensive documentation for all of its components and is open to contributions for extensions and improvements.

Evaluation of the accuracy and the efficiency of many of the existing methods for SfM are solely based on empirical observations. Consequently, easily accessible datasets (*e.g.* sets of raw images, extracted and matched features, or intrinsic camera calibrations) possessing specific characteristics, ranging from benchmark image sets with ground truth motion parameters to large unordered Internet photo collections, are of great value for comparative performance evaluations. Moreover, and perhaps more importantly, collection and preprocessing (*e.g.* feature extraction and matching) of such datasets can be fairly time-consuming and expensive. Hence, the unavailability of such datasets may significantly hinder the development and testing of new methods for SfM. A relatively early work providing several benchmark datasets (now considered to have small sizes) including high-resolution images, intrinsic camera parameters, ground truth camera motion and resulting dense 3D constructions, is that of Strecha, Hansen, Van Gool, Fua

[23] The COLMAP library is accessible at https://colmap.github.io.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

and Thoennessen (2008).[24] The availability of ground truth intrinsic and extrinsic camera parameters in these datasets makes them useful for evaluating camera motion recovery accuracy (see *e.g.* Arie-Nachimson *et al.* 2012, Özyeşil *et al.* 2015) with *high precision*. As discussed in Section 4, more recently there has been growing interest in SfM using large and unordered images. One highly popular source of data is the *BigSFM* platform,[25] which includes large SfM examples with original images, extracted and matched features, relative camera positions and intrinsic camera parameters, and is based on the work of Agarwal *et al.* (2009, 2010), Snavely *et al.* (2006), Wilson and Snavely (2013, 2014) and others; see also the links in the data section of the BigSFM web page. As a widely used source of data, the BigSFM platform provides a relatively standardized basis for comparative performance evaluation of various methods in the literature, targeting efficient feature extraction and matching, global camera motion estimation, incremental and global bundle adjustment, handling symmetries and ambiguities, geo-positioning, and so on.

For an extensive account of other open-source software packages and data resources, we refer the reader to the *awesome 3D reconstruction list*[26] by Pierre Moulon.

7. Conclusion

Our survey has covered various methods in the SfM literature pertaining to camera location estimation, 3D structure and motion refinement, simultaneous localization and mapping (SLAM), feature extraction and matching, symmetric and ambiguous scenes, and alternative features and camera models. As also mentioned in the previous sections, for additional topics and methods in the literature, we refer the reader to other works, for example Hartley and Zisserman (2000), Triggs *et al.* (2000), Oliensis (2000), Zhang (1998), Tron *et al.* (2016), Kanade and Morris (1998), Musialski *et al.* (2013), Mikolajczyk and Schmid (2005) and Gauglitz *et al.* (2011). In this section we conclude by briefly mentioning potential areas of progress in the SfM problem.

For most of the existing methods in the SfM literature, one of the most computationally intensive steps (usually assumed to have been accomplished by an existing method) is the feature extraction and matching. Even though there exist sufficiently accurate methods for this task, the computational cost of these methods can be prohibitive for specific applications, and some

[24] Datasets are available at http://cvlabwww.epfl.ch/data/multiview/.
[25] Accessible at http://www.cs.cornell.edu/projects/bigsfm/.
[26] See https://github.com/openMVG/awesome_3DReconstruction_list/.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

of the relatively efficient techniques turn out to induce significant degradation in accuracy. In other words, the existing methods seem to have peaked in improving the computational efficiency without suffering losses in accuracy. In this respect, more efficient feature extraction and matching methods, which have improved invariance properties and enable fusion of local and global image characteristics (*e.g.* to maintain higher resilience to ambiguities in images), are of high value.

As discussed in Section 6.2, existing SfM methods for ambiguous scenes can be classified into the two main categories of relatively computationally inefficient methods designed for high proportions of mismatched features (usually observed for small, experimental image sets) and relatively efficient techniques for large image sets (*e.g.* community photo collections) that tend to fail at disambiguation of scenes with large ambiguous structures. Hence, we believe that there is potential for alternative methods with acceptable levels of computational efficiency that have improved robustness to higher numbers of mismatched features.

Considering the recent boost of interest in 3D scene perception and reconstruction (*e.g.* for augmented and virtual reality applications or self-driving cars), we believe that specific SfM algorithms targeting efficient solutions of relatively tightly constrained problems (*e.g.* accurate and fast depth estimation with known camera motion) will attract more attention in the future. Consequently, the design and analysis of efficient SfM techniques, with predictable stability in accuracy and provable robustness, are an important potential area of development.

Finally, even though there exist various theoretical works in the literature that study fundamental problems in SfM and/or provide rigorous analysis of stability and robustness of specific methods (see *e.g.* the early works of Chiuso, Brockett and Soatto 2000, Ma, Košecká and Sastry 2001, Soatto 1997 and the relatively recent works of Özyeşil *et al.* 2015, Hand *et al.* 2017, Hand *et al.* 2015), we believe that the SfM community would still benefit greatly from rigorous results on fundamental problems, for example, what is the theoretically maximal amount of mismatched features or level of noise in images that can be tolerated for stable structure recovery, and can this be achieved efficiently? Further, there is a need for theoretical analysis of stability, robustness and the computational efficiency of existing and new methods.

Acknowledgements

The authors wish to thank Noah Snavely, David G. Lowe and Federica Arrigoni for their permission to include Figures 3.4, 4.1, 4.2, 4.3, Figures 6.1, 6.2 and Figure 3.7, respectively; see the captions of these figures for the corresponding publications of these authors.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

A.S. was partially supported by Award Number R01GM090200 from the NIGMS, FA9550-12-1-0317 from AFOSR, the Simons Foundation Investigator Award and Simons Collaboration on Algorithms and Geometry, and the Moore Foundation Data-Driven Discovery Investigator Award. R.B. was supported in part by the Israel Science Foundation grant 1265/14 and by the Minerva Foundation with funding from the Federal German Ministry for Education and Research.

REFERENCES[27]

S. Agarwal, N. Snavely, S. Seitz and R. Szeliski (2010), Bundle adjustment in the large. In *ECCV 2010: 11th European Conference on Computer Vision*, part II, Vol. 6312 of Lecture Notes in Computer Science, Springer, pp. 29–42.

S. Agarwal, N. Snavely, I. Simon, S. Seitz and R. Szeliski (2009), Building Rome in a day. In *ICCV 2009: 12th IEEE International Conference on Computer Vision*.

D. Aliaga (2001), Accurate catadioptric calibration for real-time pose estimation in room-size environments. In *ICCV 2001: 8th IEEE International Conference on Computer Vision*, pp. 127–134.

M. Arie-Nachimson, S. Kovalsky, I. Kemelmacher-Shlizerman, A. Singer and R. Basri (2012), Global motion estimation from point matches. In *3DimPVT 2012: 2nd IEEE International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, pp. 81–88.

F. Arrigoni, A. Fusiello and B. Rossi (2016), Camera motion from group synchronization. In *3DV 2016: 4th IEEE International Conference on 3D Vision*, pp. 546–555.

J. Aulinas, Y. Petillot, J. Salvi and X. Lladó (2008), The SLAM problem: A survey. In *2008 Conference on Artificial Intelligence Research and Development; 11th International Conference of the Catalan Association for Artificial Intelligence*, IOS Press, pp. 363–371.

A. Bartoli and P. Sturm (2005), 'Structure-from-motion using lines: Representation, triangulation, and bundle adjustment', *Comput. Vision Image Underst.* **100**, 416–441.

H. Bay, T. Tuytelaars and L. Van Gool (2006), SURF: Speeded up robust features. In *ECCV 2006: 9th European Conference on Computer Vision*, Vol. 3951 of Lecture Notes in Computer Science, Springer, pp. 404–417.

R. Bolles and M. Fischler (1981), A RANSAC-based approach to model fitting and its application to finding cylinders in range data. In *IJCAI '81: 7th International Joint Conference on Artificial intelligence*, part 2, pp. 637–643.

M. Brand, M. Antone and S. Teller (2004), Spectral solution of large-scale extrinsic camera calibration as a graph embedding problem. In *ECCV 2004: 8th European Conference on Computer Vision*, Vol. 3022 of Lecture Notes in Computer Science, Springer, pp. 262–273.

[27] The URLs cited in this work were correct at the time of going to press, but the publisher and the authors make no undertaking that the citations remain live or are accurate or appropriate.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

P. Chang and M. Hebert (2000), Omni-directional structure from motion. In *2000 IEEE Workshop on Omnidirectional Vision*, pp. 127–133.

A. Chatterjee and V. Govindu (2013), Efficient and robust large-scale rotation averaging. In *ICCV 2013: IEEE International Conference on Computer Vision*, pp. 521–528.

A. Chiuso, R. Brockett and S. Soatto (2000), 'Optimal structure from motion: Local ambiguities and global estimates', *Int. J. Comput. Vision* **39**, 195–228.

A. Cohen, C. Zach, S. Sinha and M. Pollefeys (2012), Discovering and exploiting 3D symmetries in structure from motion. In *CVPR 2012: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1514–1521.

D. Crandall, A. Owens, N. Snavely and D. Huttenlocher (2011), Discrete-continuous optimization for large-scale structure from motion. In *CVPR 2011: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3001–3008.

M. Cucuringu, A. Singer and D. Cowburn (2012), 'Eigenvector synchronization, graph rigidity and the molecule problem', *Inf. Inference* **1**, 27–67.

J. Engel, T. Schöps and D. Cremers (2014), LSD-SLAM: Large-scale direct monocular SLAM. In *ECCV 2014: 13th European Conference on Computer Vision*, Vol. 8690 of Lecture Notes in Computer Science, Springer, pp. 834–849.

J. Fuentes-Pacheco, J. Ruiz-Ascencio and J. Rendón-Mancha (2015), 'Visual simultaneous localization and mapping: A survey', *Artificial Intelligence Review* **43**, 55–81.

Y. Furukawa and J. Ponce (2010*a*), 'Accurate, dense, and robust multiview stereopsis', *IEEE Trans. Pattern Anal. Mach. Intel.* **32**, 1362–1376.

Y. Furukawa and J. Ponce (2010*b*), PMVS: Patch-based multi-view stereo software. http://www.di.ens.fr/pmvs/

Y. Furukawa, B. Curless, S. Seitz and R. Szeliski (2010*a*), CMVS: Clustering views for multi-view stereo. http://www.di.ens.fr/cmvs/

Y. Furukawa, B. Curless, S. Seitz and R. Szeliski (2010*b*), Towards Internet-scale multi-view stereo. In *CVPR 2010: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1434–1441.

S. Gauglitz, T. Höllerer and M. Turk (2011), 'Evaluation of interest point detectors and feature descriptors for visual tracking', *Int. J. Comput. Vision* **94**, 335–360.

J. Gluckman and S. Nayar (1998), Ego-motion and omnidirectional cameras. In *Sixth International Conference on Computer Vision*, IEEE cat. no. 98CH36271, pp. 999–1005.

T. Goldstein, P. Hand, C. Lee, V. Voroninski and S. Soatto (2016), ShapeFit and ShapeKick for robust, scalable structure from motion. In *ECCV 2016: 14th European Conference on Computer Vision*, Vol. 9911 of Lecture Notes in Computer Science, Springer, pp. 289–304.

V. Govindu (2001), Combining two-view constraints for motion estimation. In *CVPR 2001: IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, part 2, pp. II-218–II-225.

V. Govindu (2004), Lie-algebraic averaging for globally consistent motion estimation. In *CVPR 2004: IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, part 1, pp. I-684–I-691.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

P. Hand, C. Lee and V. Voroninski (2015), Exact simultaneous recovery of locations and structure from known orientations and corrupted point correspondences. arXiv:1509.05064

P. Hand, C. Lee and V. Voroninski (2017), 'ShapeFit: Exact location recovery from corrupted pairwise directions', *Comm. Pure Appl. Math.*, to appear.

R. Hartley (1997), 'In defense of the eight-point algorithm', *IEEE Trans. Pattern Anal. Mach. Intel.* **19**, 580–593.

R. Hartley and A. Zisserman (2000), *Multiple View Geometry in Computer Vision*, Cambridge University Press.

M. Havlena, A. Torii and T. Pajdla (2010), Efficient structure from motion by graph optimization. In *ECCV 2010: 11th European Conference on Computer Vision*, Vol. 6312 of Lecture Notes in Computer Science, Springer, pp. 100–113.

M. Havlena, A. Torii, J. Knopp and T. Pajdla (2009), Randomized structure from motion based on atomic 3D models from camera triplets. In *CVPR 2009: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2874–2881.

J. Hernandez, K. Tsotsos and S. Soatto (2015), Observability, identifiability and sensitivity of vision-aided inertial navigation. In *ICRA 2015: IEEE International Conference on Robotics and Automation*, pp. 2319–2325.

N. Jiang, Z. Cui and P. Tan (2013), A global linear method for camera pose registration. In *ICCV 2013: IEEE International Conference on Computer Vision*, pp. 481–488.

N. Jiang, P. Tan and L. Cheong (2012), Seeing double without confusion: Structure-from-motion in highly ambiguous scenes. In *CVPR 2012: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1458–1465.

T. Kanade and D. Morris (1998), 'Factorization methods for structure from motion', *Philos. Trans. Royal Soc. London* **356**, 1153–1173.

J. Kannala and S. Brandt (2006), 'A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses', *IEEE Trans. Pattern Anal. Mach. Intel.* **28**, 1335–1340.

Y. Ke and R. Sukthankar (2004), PCA-SIFT: A more distinctive representation for local image descriptors. In *CVPR 2004: IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, part 2, pp. 506–513.

H. Longuet-Higgins (1981), 'A computer algorithm for reconstructing a scene from two projections', *Nature* **293**, 133–135.

M. Lourakis and A. Argyros (2009), 'SBA: A software package for generic sparse bundle adjustment', *ACM Trans. Math. Softw.* **36**, 2:1–2:30.

D. Lowe (1999), Object recognition from local scale-invariant features. In *ICCV 1999: 7th IEEE International Conference on Computer Vision*, part 2, pp. 1150–1157.

D. Lowe (2004), 'Distinctive image features from scale-invariant keypoints', *Int. J. Comput. Vision* **60**, 91–110.

Y. Ma, J. Košecká and S. Sastry (2001), 'Optimization criteria and geometric algorithms for motion and structure estimation', *Int. J. Comput. Vision* **44**, 219–249.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

D. Martinec and T. Pajdla (2003), Line reconstruction from many perspective images by factorization. In *CVPR 2003: IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, part 1, pp. 497–502.

D. Martinec and T. Pajdla (2007), Robust rotation and translation estimation in multiview reconstruction. In *CVPR 2007: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8.

B. Micusik and T. Pajdla (2006), 'Structure from motion with wide circular field of view cameras', *IEEE Trans. Pattern Anal. Mach. Intel.* **28**, 1135–1149.

K. Mikolajczyk and C. Schmid (2005), 'A performance evaluation of local descriptors', *IEEE Trans. Pattern Anal. Mach. Intel.* **27**, 1615–1630.

P. Moulon, P. Monasse and R. Marlet (2013), Global fusion of relative motions for robust, accurate and scalable structure from motion. In *ICCV 2013: IEEE International Conference on Computer Vision*, pp. 3248–3255.

P. Moulon, P. Monasse, R. Marlet *et al.*, OpenMVG: An open multiple view geometry library. https://github.com/openMVG/openMVG

E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser and P. Sayd (2009), 'Generic and real-time structure from motion using local bundle adjustment', *Image and Vision Computing* **27**, 1178–1193.

P. Musialski, P. Wonka, D. Aliaga, M. Wimmer, L. Van Gool and W. Purgathofer (2013), 'A survey of urban reconstruction', *Computer Graphics Forum* **32**, 146–177.

J. Oliensis (2000), 'A critique of structure-from-motion algorithms', *Comput. Vision Image Underst.* **80**, 172–214.

O. Özyeşil and A. Singer (2015), Robust camera location estimation by convex programming. In *CVPR 2015: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2674–2683.

O. Özyeşil, A. Singer and R. Basri (2015), 'Stable camera motion estimation using convex programming', *SIAM J. Imaging Sci.* **8**, 1220–1262.

D. Pachauri, R. Kondor, G. Sargur and V. Singh (2014), Permutation diffusion maps (PDM) with application to the image association problem in computer vision. In *Advances in Neural Information Processing Systems 27* (Z. Ghahramani *et al.*, eds), Curran Associates, pp. 541–549.

M. Pollefeys, D. Nistér, J.-M. Frahm, A. Akbarzadeh, P. Mordohai, B. Clipp, C. Engels, D. Gallup, S.-J. Kim, P. Merrell, C. Salmi, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewénius, R. Yang, G. Welch and H. Towles (2008), 'Detailed real-time urban 3D reconstruction from video', *Int. J. Comput. Vision* **78**, 143–167.

L. Quan and T. Kanade (1997), 'Affine structure from line correspondences with uncalibrated affine cameras', *IEEE Trans. Pattern Anal. Mach. Intel.* **19**, 834–845.

S. Ramalingam, S. Lodha and P. Sturm (2006), 'A generic structure-from-motion framework', *Comput. Vision Image Underst.* **103**, 218–228.

R. Roberts, S. Sinha, R. Szeliski, D. Steedly and R. Szeliski (2011), Structure from motion for scenes with large duplicate structures. In *CVPR 2011: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3137–3144.

F. Schaffalitzky and A. Zisserman (2002), Multi-view matching for unordered image sets, or 'How do I organize my holiday snaps?' In *ECCV 2002: 7th European*

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

Conference on Computer Vision, part 1, Vol. 2350 of Lecture Notes in Computer Science, Springer, pp. 414–431.

G. Schindler, P. Krishnamurthy and F. Dellaert (2006), Line-based structure from motion for urban environments. In *Third International Symposium on 3D Data Processing, Visualization, and Transmission*, IEEE, pp. 846–853.

J. Schönberger and J.-M. Frahm (2016), Structure-from-motion revisited. In *CVPR 2016: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4104–4113.

J. Schönberger, E. Zheng, J.-M. Frahm and M. Pollefeys (2016), Pixelwise view selection for unstructured multi-view stereo. In *ECCV 2016: 14th European Conference on Computer Vision*, part III, Springer, pp. 501–518.

O. Shakernia, R. Vidal and S. Sastry (2003), Omnidirectional egomotion estimation from back-projection flow. In *CVPRW 2003: Computer Vision and Pattern Recognition Workshop*, part 7, pp. 82–82.

A. Singer (2011), 'Angular synchronization by eigenvectors and semidefinite programming', *Appl. Comput. Harmon. Anal.* **30**, 20–36.

S. Sinha, D. Steedly and R. Szeliski (2010), A multi-stage linear approach to structure from motion. In *ECCV 2010 Workshops: Trends and Topics in Computer Vision*, part II, Vol. 6554 of Lecture Notes in Computer Science, Springer, pp. 267-281.

N. Snavely, S. Seitz and R. Szeliski (2006), 'Photo tourism: Exploring photo collections in 3D', In *ACM Trans. Graph.* **25**, 835–846.

N. Snavely, S. Seitz and R. Szeliski (2008*a*), 'Modeling the world from internet photo collections', *Int. J. Comput. Vision* **80**, 189–210.

N. Snavely, S. Seitz and R. Szeliski (2008*b*), Skeletal graphs for efficient structure from motion. In *CVPR 2008: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8.

S. Soatto (1997), '3-D structure from visual motion: Modeling, representation and observability', *Automatica* **33**, 1287–1312.

C. Strecha, W. Hansen, L. Van Gool, P. Fua and U. Thoennessen (2008), On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *CVPR 2008: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8.

P. Sturm and B. Triggs (1996), A factorization based algorithm for multi-image projective structure and motion. In *ECCV '96: 4th European Conference on Computer Vision*, part II, Vol. 1065 of Lecture Notes in Computer Science, Springer, pp. 709–720.

C. Sweeney (2016), Theia multiview geometry library: Tutorial and reference. http://theia-sfm.org

C. Taylor and D. Kriegman (1995), 'Structure and motion from line segments in multiple images', *IEEE Trans. Pattern Anal. Mach. Intel.* **17**, 1021–1032.

C. Tomasi and T. Kanade (1992), 'Shape and motion from image streams under orthography: A factorization method', *Int. J. Comput. Vision* **9**, 137–154.

B. Triggs (1996), Factorization methods for projective structure and motion. In *CVPR '96: IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 845–851.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press

B. Triggs, P. McLauchlan, R. Hartley and A. Fitzgibbon (2000), Bundle adjustment: A modern synthesis. In *Vision Algorithms: Theory and Practice*, Vol. 1883 of Lecture Notes in Computer Science, Springer, pp. 298–375.

R. Tron and R. Vidal (2014), 'Distributed 3-D localization of camera sensor networks from 2-D image measurements', *IEEE Trans. Automatic Control* **59**, 3325–3340.

R. Tron, X. Zhou and K. Daniilidis (2016), A survey on rotation optimization in structure from motion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 77–85.

K. Tsotsos, A. Chiuso and S. Soatto (2015), Robust inference for visual-inertial sensor fusion. In *ICRA 2015: IEEE International Conference on Robotics and Automation*, pp. 5203–5210.

T. Tuytelaars and K. Mikolajczyk (2008), 'Local invariant feature detectors: A survey', *Found. Trends Comput. Graphics Vision* **3**, 177–280.

A. Vedaldi, G. Guidi and S. Soatto (2007), Moving forward in structure from motion. In *CVPR 2007: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–7.

L. Wang and A. Singer (2013), 'Exact and stable recovery of rotations for robust synchronization', *Inf. Inference* **2**, 145–193.

K. Wilson and N. Snavely (2013), Network principles for SfM: Disambiguating repeated structures with local context. In *ICCV 2013: IEEE International Conference on Computer Vision*, pp. 513–520.

K. Wilson and N. Snavely (2014), Robust global translations with 1DSfM. In *ECCV 2014: 13th European Conference on Computer Vision*, part III, Vol. 8691 of Lecture Notes in Computer Science, Springer, pp. 61–75.

C. Wu (2007), SiftGPU: A GPU implementation of scale invariant feature transform (SIFT). http://cs.unc.edu/~ccwu/siftgpu/

C. Wu (2011), VisualSFM: A visual structure from motion system. http://ccwu.me/vsfm/

C. Wu, S. Agarwal, B. Curless and S. Seitz (2011), Multicore bundle adjustment. In *CVPR 2011: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3057–3064.

G. Younes, D. Asmar and E. Shammas (2016), A survey on non-filter-based monocular visual SLAM systems. arXiv.org:1607.00470

C. Zach, M. Klopschitz and M. Pollefeys (2010), Disambiguating visual relations using loop constraints. In *CVPR 2010: IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1426–1433.

Z. Zhang (1998), 'Determining the epipolar geometry and its uncertainty: A review', *Int. J. Comput. Vision* **27**, 161–195.

https://doi.org/10.1017/S096249291700006X Published online by Cambridge University Press