Robust Global Translations with 1DSfM

Kyle Wilson and Noah Snavely

Cornell University, Ithaca, NY, USA
{wilsonkl,snavely}@cs.cornell.edu

Abstract. We present a simple, effective method for solving structure from motion problems by averaging epipolar geometries. Based on recent successes in solving for global camera rotations using averaging schemes, we focus on the problem of solving for 3D camera translations given a network of noisy pairwise camera translation directions (or 3D point observations). To do this well, we have two main insights. First, we propose a method for removing outliers from problem instances by solving simpler low-dimensional subproblems, which we refer to as 1DSfM problems. Second, we present a simple, principled averaging scheme. We demonstrate this new method in the wild on Internet photo collections.

Keywords: Structure from Motion, translations problem, robust estimation.

1 Introduction

Recent work on the unstructured Structure from Motion (SfM) problem has had renewed interest in global methods. Unlike sequential approaches which build 3D models from photo collections by iteratively growing a small seed model, global (or batch) methods for SfM consider the entire problem at once. By doing this they avoid several disadvantages of sequential methods, which have tended to be costly, requiring a repeated nonlinear model refinement (bundle adjustment) to avoid errors. Also, unlike global methods, sequential SfM necessarily treats images unequally, where those considered first can have a disproportionate effect on the final model. In practice, this behavior can sometimes lead to cascading mistakes and can exacerbate the problem of drift.

However, global methods have difficulties of their own. A key problem is that reasoning about outliers is challenging. Techniques from sequential methods, such as filtering out measurements inconsistent with the current model at each step, are not directly applicable in a global setting. It is harder to reason *a priori* about which measurements are unreliable.

In this work, we present a new global SfM method; like other methods, we solve first for global camera rotations, then translations, given a set of pairwise epipolar geometries. As there has been significant progress on the rotations problem, we focus on translations, and offer two key insights. The first, which we call **1DSfM**, is a simple way to preprocess a problem instance to remove outlier measurements. 1DSfM is based on reducing a difficult problem to single-dimensional subproblems where inference becomes a more straightforward combinatorial computation. Under this 1D projection, a

D. Fleet et al. (Eds.): ECCV 2014, Part III, LNCS 8691, pp. 61–75, 2014.

translations problem becomes an instance of MINIMUM FEEDBACK ARC SET, a well studied graph problem. By solving for a 1D ordering, we recover information about which 3D measurements are likely inconsistent. Second, we describe a new, very simple solver for the translations problem. Surprisingly, we find that non-linear optimization with this solver—even with random initialization—works remarkably well, especially once outliers have been removed. Hence, our 1DSfM-based outlier removal technique goes hand in hand with our simple translations solver to achieve high-quality results.

We show the effectiveness of our two methods on a variety of landmark-scale Internet community photo collections, covering a range of sizes and scene types. Our code and data are available at `http://www.cs.cornell.edu/projects/1dsfm`.

2 Related Work

While some earlier SfM methods were global, such as factorization [20], most current large-scale SfM systems involve sequential reconstruction [19,2,9]. Sequential methods build models a few images at a time, often with bundle adjustment in between steps. However, there has been significant recent interest in revisiting global methods because of their potential for improved speed and decreased dependence on local decisions or image ordering. These methods often work by first estimating an initial set of camera poses (typically through use of estimated relative poses between pairs or triplets), followed by a global bundle adjustment to refine this initial solution. With a few exceptions (e.g. [12]), these methods first solve for camera rotations, and then camera translations.

Rotations. A number of methods have been proposed for solving for global rotations from pairwise estimates of relative rotations. Some methods formulate the problem as a linear system by relaxing constraints on rotation parameterizations [11,17,3]. Enqvist et al. [8] look for a best spanning tree of pairwise rotations to filter outliers in advance. Sinha et al. [18] use vanishing point estimates as an additional cue. More recently, Hartley et al. [13] as well as Chatterjee and Govindu [5] have presented robust l_1 methods based on the Lie algebraic structure of the manifold of rotations. Finally, Fredriksson and Olsson [10] present an approach based on primal and dual problems which can certify if a solution is globally optimal. We have found the method of Chatterjee and Govindu [5] particularly effective, and use it to produce input for our method.

Translations. Like the rotations problem, the translations problem is often formulated as computing global camera translations from pairwise ones. Some approaches to solving this problem are based on a linear system of cross product constraints [11,3]. Others use Second Order Cone Programming, based on the l_∞ norm [16,17]. Such methods require very careful attention to outliers. Brand et al. [4] use a spectral approach, but do not address outlier noise. Sinha et al. [18] robustly compute similarity transformations that align pairs of reconstructions, and then average over these transformations. Recently Jiang et al. [14] have formulated a linear constraint with geometric, rather than algebraic meaning, based on co-planarity in triplets of cameras. Finally, Crandall et al. [6] take a different approach to optimization, using a complex scheme involving a discrete Markov Random Field search and a continuous Levenberg-Marquardt refinement to robustly explore the solution space. Our translations solver optimizes an objective function that depends only on comparing measurement directions to model directions, as opposed to other methods [11,3] where the objective function is also a function of the distance between images. To avoid the resulting bias, Govindu proposes an iterative reweighting scheme [11], which is unnecessary in our approach. Jiang et al. discuss the importance of geometric vs. algebraic cost functions, as they minimize a value that has physical significance. In this sense our cost function is also geometric (but in the space of measurements, rather than in the solution space).

Handling Outliers. A key contribution of our work is a simple algorithm for removing outliers in a translations problem. Zach et al. [23] detect outlier epipolar geometries by looking at loop closure in graph cycles. Our method for outlier removal is similar in motivation, but by projecting into a single dimension we solve tractable subproblems that reduce to a simple combinatorial graph problem.

3 Problem Formulation

The gold standard method for structure from motion is bundle adjustment—the joint nonlinear refinement of camera and structure parameters [21]. However, bundle adjustment is a largely local search, and its success depends critically on initialization. Given a good initial guess, bundle adjustment can produce high quality solutions, but if the guess is bad, the optimization may fall into local minima far from the optimal solution. For this reason most SfM methods focus on creating a close-enough initialization which can then be refined with bundle adjustment; sequential (or incremental) SfM methods are one such approach that use repeated bundle adjustment on increasingly large problems to reach a good solution.

Initializing bundle adjustment involves estimating a rotation matrix and a position for each camera. In our notation, a rotation matrix $\mathbf{R}_i$ represents a mapping from world coordinates to camera coordinates, and a translation $\mathbf{t}_i$ represents a location in the world coordinate frame (in our work, we use "location" and "translation" interchangeably, in a slight abuse of terminology). As with other recent global methods, our input is a set of images V, and a network of computed epipolar geometries $(\hat{\mathbf{R}}_{ij}, \hat{\mathbf{t}}^l_{ij})$ between pairs (i, j) of overlapping images. (We will use a hat for epipolar geometries, to emphasize that they are our input measurements. We use a superscript l for relative translations between two cameras, which are defined in a local coordinate system.) These epipolar geometries are not available for all camera pairs, because not all pairs of images visually overlap. These inputs define a graph we call the epipolar geometry (EG) graph $G = (V, E)$ on a set of images V, where for every edge $(i, j) \in E$ we have a measurement $(\hat{\mathbf{R}}_{ij}, \hat{\mathbf{t}}^l_{ij})$. Given perfect measurements, global camera poses $(\mathbf{R}_i, \mathbf{t}_i)$ would satisfy

$$\hat{\mathbf{R}}_{ij} = \mathbf{R}_i^\top \mathbf{R}_j \tag{1}$$

$$\lambda_{ij} \hat{\mathbf{t}}^l_{ij} = \mathbf{R}_i^\top (\mathbf{t}_j - \mathbf{t}_i) \tag{2}$$

where λ_{ij}'s are unknown scaling factors (unique up to global gauge ambiguity).

Following a now-common approach [11,16,17,6,3,14], we separate the initialization into two stages: a rotations problem and a translations problem. These two together produce an initialization to a final bundle adjustment. Recent work has been successful in solving the rotations problem robustly [13,10,5]; we build on this work and focus on the translations problem. Given estimates $\mathbf{R}_i$ of camera rotation matrices, we can write our measurement of the direction from camera i to camera j as $\hat{\mathbf{t}}_{ij} = \mathbf{R}_i \hat{\mathbf{t}}^l_{ij}$, where $\hat{\mathbf{t}}_{ij}$ is a unit 3-vector (i.e., a point on the unit sphere) in the global coordinate system. Hence, the translations problem reduces to the following graph embedding problem:

$$\begin{array}{rl}
\text{Given:} & \text{Graph } G = (V, E) \\
 & \text{Measurements } \hat{\mathbf{t}}_{ij} : E \to S^2 \\
 & \text{Metric } d : S^2 \times S^2 \to \mathbb{R} \\
\text{Minimize:} & \displaystyle\sum_{(i,j) \in E} d\left(\hat{\mathbf{t}}_{ij}, \frac{\mathbf{t}_j - \mathbf{t}_i}{\|\mathbf{t}_j - \mathbf{t}_i\|}\right) \\
\text{over embeddings:} & \mathcal{T} : V \to \mathbb{R}^3 \qquad \text{i.e. } \mathcal{T} = \{\mathbf{t}_i | i \in V\}
\end{array}$$

Note that in this framework, the second endpoint j of an edge may be a point or a camera. Camera-point constraints can be important for achieving full scene coverage, and for avoiding degeneracies arising from collinear motion, an issue discussed in [14].

The formulation above does not specify the exact form of our objective function. It also excludes objective functions that depend on the distance between $\mathbf{t}_i$ and $\mathbf{t}_j$, rather than only the direction. These issues will both be discussed in Section 5.

Finally, this problem is made greatly more difficult by noise. We hope that most EGs will be approximately correct, but sometimes calculating EGs returns a wildly incorrect solution. For the translations problem we assume a mixed model of small variance inlier noise with a smaller fraction of outlier noise distributed uniformly over S^2.

4 Outlier Removal Using 1DSfM

By removing bad measurements in advance we can solve problems more accurately and reliably. In this section, we present a new method for identifying outlier measurements by projecting translations problems to 1-dimensional subproblems which we can solve more easily. Our approach is related to previous work [23] which detects outliers as measurements that cannot be consistently chained along cycles. However, there are usually many cycles to enumerate, and inferring erroneous measurements from bad cycles is difficult for large problems. Our method is based on many smaller, simpler inferences that are then aggregated. This makes outlier detection tractable even for large problems where [23] has difficulty.

The translations problem described above is a 3-dimensional embedding. One way to approach outlier detection is to try to first simplify this underlying problem. For instance, we could project the 3D problem onto a ground plane, resulting in a 2D graph embedding problem. In other words, we could ignore the z component of each measurement, and consider only the 2D projections: $\hat{\mathbf{t}}_{ij} \mapsto \hat{\mathbf{t}}_{ij} - \mathrm{proj}_{\hat{\mathbf{k}}} \hat{\mathbf{t}}_{ij}$, where $\hat{\mathbf{k}} = \langle 0, 0, 1 \rangle$. In this projected problem, we would need to assign an (x, y) pair to each vertex.

In our work, we take this idea a step further and project onto a *single* dimensional subspace. Consider projecting a translations problem onto the x-axis, as in the blue problem in Figure 1. Only the x component of each translations measurement is now

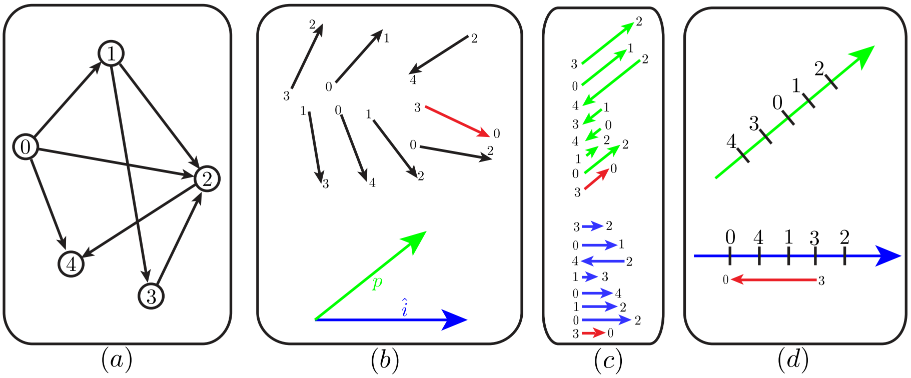


Fig. 1. A toy illustration of 1DSfM. Panel (a) is a good solution to a translations problem for reference. Panel (b) shows the translations problem input—a set of edges with orientations. One outlier edge has been added in red. We also show two directions for projection: $\hat{i}$ and p. Panel (c) contains only the projected translations problems, one for each projection direction. These problems are instances of MINIMUM FEEDBACK ARC SET. Finally, (d) contains good solutions to the 1D problems in (c). In the lower case, not all ordering constraints can be satisfied, due to the outlier edge. Note that outlier edges may be consistent in some subproblems but not in others.

relevant to the problem: $\hat{\mathbf{t}}_{ij} \cdot \langle 1, 0, 0 \rangle = x_{ij}$, and we need to assign an x-coordinate to each vertex. Recall that our pairwise translation measurements represent directions, but not distances. On the x-axis there are only two directions: left and right. Hence, an embedding is consistent with edge (i, j) if $x_{ij} > 0$ and i embeds to the left of j, and vice versa for $x_{ij} < 0$. Figure 1 panel (d) shows such an embedding. Note that in 1D, edge directions have become *ordering constraints*: all embeddings with the same ordering are equally consistent with our problem. Hence, this 1D problem is a combinatorial ordering problem, rather than a continuous optimization problem: we want to find a global ordering of the vertices that satisfies the pairwise orderings as well as possible. We can formulate this problem on a directed version of our graph G, as described below.

Figure 1 also illustrates projecting the same problem in a different direction (in green). Notice that the outlier shown in red is inconsistent in one projection direction, but not in another. To catch as many outliers as possible, we embed a graph in many 1D subspaces, each defined by a unit vector p. For each subproblem only the component of translations measurements $\hat{\mathbf{t}}_{ij}$ in the direction of p is relevant to the optimization: $\hat{\mathbf{t}}_{ij} \mapsto p \cdot \hat{\mathbf{t}}_{ij} = w_{ij}$. By regarding the pair $(i, j), w_{ij}$ as equivalent to $(j, i), -w_{ij}$, we can form a problem with directed edges with positive edge weights. Given a directed graph formed in this way, we try to find an ordering that satisfies as many of these pairwise constraints as possible; the edges that are inconsistent with this ordering are potential outliers. This is a well-studied problem in optimization called MINIMUM FEEDBACK ARC SET (MFAS). Unfortunately it is NP-complete, but there is a rich literature of approximation algorithms. We found that a variant of [7], as detailed in Algorithm 1, worked very well on our problems. This algorithm greedily builds an order from left to right. It always selects a next node that breaks no order constraints if possible. If not, it selects the next node to maximize a heuristic: $(1 + \deg_{out}(v))/(1 + \deg_{in}(v))$,

Algorithm 1. MFAS ordering

```
procedure MFAS($G = (V, E), w_{ij}$)                          ▷ Order vertices on a line
    $E_{dir} \leftarrow \{(i,j)|w_{ij} > 0\} \cup \{(j,i)|w_{ij} < 0\}$    ▷ Put in form of an MFAS problem
    $\pi = [\,]$                                              ▷ build a permutation here
    $V_{rem} = V$                                             ▷ unchosen vertices
    $G' \leftarrow (V_{rem}, E_{dir})$
    while $G' \neq \varnothing$ do
        $x \leftarrow \{v \in G' | v \text{ is source}\}$          ▷ select all sources first
        if $x = \varnothing$ then
            $x \leftarrow \text{argmax}_{v \in V_{rem}} \frac{1+\text{deg}_{out}(v)}{1+\text{deg}_{in}(v)}$    ▷ heuristic
        $\pi$.append($x$)
        $V_{rem} \leftarrow V_{rem} - x$
        $G' \leftarrow$ restrict($G', V_{rem}$)                ▷ restrict graph
```

Algorithm 2. Combinatorial Cleaning

```
procedure CLEAN($G = (V, E), \hat{t}, N, \tau$)               ▷ Remove outlier measurements from $E$
    $x_{ij} \leftarrow 0 \quad \forall (i,j) \in E$            ▷ Accumulator for broken edge weight
    for $k \leftarrow 1, N$ do
        $\hat{p} \leftarrow \text{RAND}(\hat{t})$              ▷ Sample $\hat{p}$ proportional to density of $\hat{t}$ on $S^2$
        $w_{ij} \leftarrow \hat{p} \cdot \hat{t}_{ij} \quad \forall (i,j) \in E$
        $\pi \leftarrow \text{MFAS}((V, E), w_{ij})$           ▷ Order vertices along direction $\hat{p}$
        for $(i,j) \in E$ do
            if sgn($\pi(j) - \pi(i)$) $\neq$ sgn($w_{ij}$) then
                $x_{ij} \leftarrow x_{ij} + |w_{ij}|$
    $E \leftarrow \{(i,j)\,|x_{ij}/N < \tau\}$
```

where $\text{deg}_{in}(v)$ and $\text{deg}_{out}(v)$ are the sum of weights of outgoing and incoming edges of node v, respectively. We found that this ratio heuristic performs much better on our problems than the heuristic used in [7] (namely, $\text{deg}_{out}(v) - \text{deg}_{in}(v)$).

Projection from 3D to 1D necessarily loses information. Bad measurements could be missed entirely by some choices of projection direction p. To identify outliers reliably we aggregate the results of solving 1D subproblems projected in many different directions. We use a kernel density estimator to sample these projection directions randomly, proportional to the density of directions of measurements in the input problem. We sample this way because outliers stand out most clearly in directions where many edges project with high weight; picking uncommon directions (like straight up) tends to have poor signal-to-noise ratio. For each direction, if an edge (i, j) is inconsistent with the ordering we compute, we accumulate the weight $|w_{ij}|$ on that edge. Edges that accumulate weight in many subproblems are inconsistent and probably bad. After running in N sampled directions we reject edges (i, j) which have accumulated more than a threshold $\tau \cdot N$ of weighted inconsistency. This process is summarized in Algorithm 2.

5 Solving the Translations Problem

Now that we have a cleaner set of pairwise relative translations, we use them to solve for a global set of translations. In order to make our translations problem concrete, we must

Table 1. Common distances on S^2

Name	Formula	Equivalent
Geodesic	$\angle(\mathbf{u}, \mathbf{v})$	θ
Cross Product	$\mathbf{u} \times \mathbf{v}$	$\sin\theta$
Inner Product	$1 - \mathbf{u} \cdot \mathbf{v}$	$2\sin^2\frac{\theta}{2}$
Squared Chordal Distance	$\|\mathbf{u} - \mathbf{v}\|^2$	$4\sin^2\frac{\theta}{2}$

first choose an objective function to minimize. After evaluating a number of metrics, we opted to use the sum of squared chordal distance:

$$err_{ch}(\mathcal{T}) = \sum_{(i,j)\in E} d_{ch}\left(\hat{\mathbf{t}}_{ij}, \frac{\mathbf{t}_j - \mathbf{t}_i}{\|\mathbf{t}_j - \mathbf{t}_i\|}\right)^2 \qquad (3)$$

$$d_{ch}(\mathbf{u}, \mathbf{v}) = \|\mathbf{u} - \mathbf{v}\|_2 \qquad (4)$$

This is a nonlinear least squares problem, with the nonlinearity coming from the division mapping vectors to directions. We minimize it using Levenberg-Marquardt, as implemented in the Ceres software package [1]. In general, nonlinear least squares problems are not guaranteed to have a single local minimum, so a good initialization is critical. Surprisingly, we find that with this distance metric, our problems generally converged well for our test datasets even from random initialization. Although the node orderings from 1DSfM could provide an initialization, we found this no more effective than randomization.[1]

Comparative Discussion. The SfM translations problem recovers coordinates in $\mathbb{R}^3$ from measurements on the sphere S^2. Previous work has proposed objective functions based on different combinations of these two spaces. For example, the cross product used in [11] maps $S^2 \times \mathbb{R}^3 \to \mathbb{R}$. This biases the problem as error is proportional to the length of the edges in a solution. To compensate, they use an iterative reweighting framework to divide out edge length. This framework approximates a cross product map $S^2 \times S^2 \to [-1, 1]$.

We avoid this bias by comparing measurements (on S^2) directly to edge directions for a solution (also on S^2). Table 1 shows several ways to measure the distance between two directions. A natural distance on a sphere is the geodesic (great circle) distance, so each distance is also given in terms of this angle θ. Note that the cross product has both parallel and antiparallel minima, which is undesirable. The inner product and the squared chordal distance are equivalent up to a constant, but the latter is a preferable formulation because it is a sum of squares.

While our 1DSfM method seeks to remove outlier measurements, we note that one could also handle outliers using robust cost functions. Crandall et al. [6] demonstrate the utility of robust cost functions for global SfM problems, but operated within a complex discrete optimization algorithm. In our case, within an continuous optimization framework, we have found that the choice of robust function is very important—Cauchy and

[1] Formally, Eq. 5 is undefined if ever $\mathbf{t}_j = \mathbf{t}_i$ for any edge. With a random initialization and natural problems this is exceedingly unlikely. However, in this case a random perturbation could allow the algorithm to continue.

threshold-based robust costs lead to poor convergence, but we find that a Huber loss to be very effective. Our results will show that a Huber loss can improve solution quality while retaining good convergence, and that the benefit is largely orthogonal to 1DSfM.

Convergence Properties. We now give a basis for confidence in minimizing our objective function. Consider two vectors $\mathbf{x}_0$ and $\mathbf{x}_1$ in $\mathbb{R}^3$, and a convex combination of them, $\mathbf{x}_\lambda = (1-\lambda)\mathbf{x}_0 + \lambda\mathbf{x}_1$. The chordal distance is not convex here (nor quasi-convex), but a related (weaker) inequality holds: [2]

$$d_{ch}\left(\frac{\mathbf{x}_\lambda}{\|\mathbf{x}_\lambda\|}, \hat{\mathbf{t}}\right)^2 \leq \max\left\{ d_{ch}\left(\frac{\mathbf{x}_0}{\|\mathbf{x}_0\|}, \hat{\mathbf{t}}\right)^2, d_{ch}\left(\frac{\mathbf{x}_1}{\|\mathbf{x}_1\|}, \hat{\mathbf{t}}\right)^2 \right\} \tag{5}$$

(In fact, this also holds without squares, so these results also apply to a robust L_1 cost.)

So if $\mathcal{T}_0$ and $\mathcal{T}_1$ are embeddings (maps taking each vertex to $\mathbb{R}^3$), and $\mathcal{T}_\lambda$ is a convex combination of them, then we can bound the objective function at $\mathcal{T}_\lambda$:

$$err_{ch}(\mathcal{T}_\lambda) = \sum_{(i,j)\in E} d_{ch}\left(\frac{\mathcal{T}_\lambda(j) - \mathcal{T}_\lambda(i)}{\|\mathcal{T}_\lambda(j) - \mathcal{T}_\lambda(i)\|}, \hat{\mathbf{t}}_{ij}\right)^2 \leq err_{ch}(\mathcal{T}_0) + err_{ch}(\mathcal{T}_1) \tag{6}$$

This means that in a noise-free problem ($err_{ch}(\mathcal{T}_0) = 0$) the error surface would be perfectly non-decreasing away from a global minimum $\mathcal{T}_0$ (though note that all solutions are only unique up to a global gauge, and ill-posed problems may have an even larger space of global minima). With small noise we are still guaranteed that the barrier between any solution and an optimum is no higher than the optimal value of the objective function. This bound is not necessarily tight, and is not achieved in natural problems. In practice, once most outliers have been removed (by 1DSfM) we consistently find good solutions.

6 Implementation

Solving for Rotations. To compute global rotations, we run Chatterjee and Govindu's rotations averaging method [5], with the parameters suggested in their paper.

Forming a Translations Problem. Given rotations, we form a translations problem with both camera-to-camera and camera-to-point edges. We find camera-to-camera edges to be crucial for accuracy and compute them from EGs. However, these camera-to-camera edges often have areas of sparse coverage—we find that popular parts of the scene are well represented, but less photographed areas can have many fewer measurements, resulting in reconstructions that can break apart into disconnected submodels.

To address this problem, we augment the translations problem with camera-to-point edges. We find that on their own these yield a noisy solution, but they increase scene coverage and connectedness. In addition, these edges are crucial for avoiding degeneracies when cameras are nearly collinear, as discussed in [14].

[2] To be precise, this follows if we assume that $\hat{\mathbf{t}}$ and the geodesic from $\mathbf{x}_0$ to $\mathbf{x}_1$ lie in open hemisphere. Since we think of $err_{ch}(\mathcal{T}_0)$ as small, this constraint is easy to satisfy.

We use only some of all possible camera-to-point edges, as they increase problem size, with diminishing returns. Similar to [6], to choose a subset of points to add to our problem we solve a simple graph covering problem: we greedily choose points that are visible to the most (as-yet-uncovered) cameras, until all cameras see k points. (We use $k = 6$ in our experiments.) Every camera-to-point edge in with this subset of points is included in our translations problem.

Cleaning with 1DSfM. We run 1DSfM on $N = 48$ random subproblems. We then remove all edges with accumulated inconsistency scores $\geq \tau \cdot N$ from the translations problem. We use $\tau = 0.10$ in our experiments.

Solving a Translations Problem. We minimize our sum of squared chordal distance objective function using Ceres Solver [1], a state-of-the-art nonlinear least squares package. We use default solver settings, except that we set the linear solver to be an iterative Schur method with Jacobi preconditioning. Additionally, we weight each constraint to prevent camera-to-point edges from dominating the problem, since there are many more of these than camera-to-camera edges. We set camera-to-camera (cc) edge weights to 1.0 and camera-to-point (cp) edges weights to $\alpha \cdot |cc \text{ edges}| \, / \, |cp \text{ edges}|$. In our experiments we use $\alpha = 0.5$, so cc edges contribute twice as much to the objective function as cp edges. For runs using a robust cost function, we use a Huber loss with width 0.1, implemented in Ceres. After solving the translations problem, we use this initialization, along with the camera rotations, to triangulate all points, and run a final bundle adjustment using the standard reprojection error.

7 Results

We evaluated our algorithm on realistic synthetic scenes, as well as a number of medium- to large-scale Internet datasets downloaded by geotag search from Flickr, as summarized in Table 2, and shown as point clouds in Figure 3. The **Notre Dame** dataset is publicly available online with Bundler [19].

1DSfM. We demonstrate that 1DSfM accurately identifies outliers in two ways. First, we tested on synthetic problems where the error on each edge (and its inlier/outlier status) is known. We observed that synthetic problems created with some common random graph models are easier than real problems, so we form our synthetic problems by using an existing reconstruction as a problem instance (reusing the epipolar graph structure and computing pairwise translations from the sequential SfM camera positions), and then adding known perturbations to every translation direction. We sampled a random 15% of edges, replacing them with translation directions sampled uniformly at random, and perturbed the rest of the translation directions with Gaussian noise with standard deviation 11.4 degrees. (We chose these numbers as representative of real problems we have observed.) Edges with error greater than 30 degrees were deemed to be the ground truth outliers for the purposes of analysis. We ran our 1DSfM algorithm on problem instances generated from four scenes—**Roman Forum**, **Tower of London**, **Ellis Island**, and **Notre Dame**. At our threshold of $\tau = 0.1$, we found that 1DSfM classified edges with a precision of 0.96 and an recall of 0.92 (averaged across the four datasets). This high classification accuracy gives us confidence in the method.

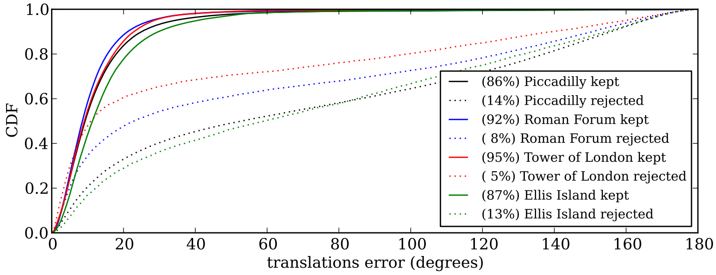


Fig. 2. Performance of 1DSfM at identifying outliers in real data. The x-axis is the error for each input translation direction. 1DSfM classifies each of these as accepted or rejected. The lines are cumulative distribution functions for both accepted edges (solid lines) and rejected edges (dotted lines) for four datasets. We see that the accepted edges have many more low residual edges, and the rejected edges contain many edges with much higher residuals.

We also demonstrate 1DSfM on real problems where ground truth is not known, but for which we can still compare translation directions from pairwise EGs to a reference reconstruction. Figure 2 shows cumulative distributions of errors in pairwise translation estimates, for edges deemed inliers and outliers by 1DSfM on four of our datasets. To approximate the inherent noisiness of the input epipolar geometries, we see how well they agree with a good sequential SfM model built using [19]. The residuals in each edge (the distance between the epipolar geometry translations direction and the translation direction computed from the reference sequential SfM model, measured with the geodesic distance) is closely tied to the noise in the input measurements. Our 1DSfM algorithm divides these input translation edges into a set to keep, and a set to discard. Figure 2 shows the distribution of residuals in these two sets. Notice that while most inlier edges have low residual (are not very noisy) the edges selected to be removed are much noisier. For example, we can see that the median error for rejected edges is around 80° for several datasets. 1DSfM removes a relatively small number of correct edges, but helps significantly by getting rid of most outlier edges.

Comparison to Sequential SfM. Because ground truth positions are usually unavailable for large-scale SfM problems, we show our method gives similar results to a sequential SfM system based on Bundler [19], but in much less time. Table 2 shows the similarity between these sequential SfM solutions and the results of several global algorithms, computed as mean and median distances between corresponding cameras between the two SfM models, across all of our datasets. The units in Table 2 are approximately in meters, as we use geotags associated with images in the collection to place each sequential SfM reconstruction in an approximate world coordinate frame, and use a RANSAC approach to compute the absolute orientation between a candidate reconstruction and the sequential SfM solution (using correspondences between camera centers).

In Table 2, we compare several variants of our method: with and without a final bundle adjustment (BA), and with four combinations of outlier treatments. We see that in all cases, we return a result with a median within several meters of the sequential SfM

Table 2. Comparison of several methods to a reference sequential SfM method based on [19]. Units are approximately in meters; sizes are the number of cameras in the largest component of the input EG graph. The methods are our translations solver combined with all four permutations of using 1DSfM and robust cost functions. The fifth column is a baseline method [11]. Results are given as N_{c}, the number of cameras reconstructed, $\bar{x}$, the average error, and $\widetilde{x}$, the median error, where by errors are the distances to corresponding cameras in [19]. Lower error is closer to the reference method. The lowest mean and median in each row are bolded, as well as two-way ties.

			without 1DSfM				with 1DSfM				Robust Loss						[11]	
			no BA	BA			no BA	BA			BA			1DSfM+BA			BA	
Name	Size	N_{c}	$\widetilde{x}$	N_{c}	$\widetilde{x}$	$\bar{x}$	$\widetilde{x}$	N_{c}	$\widetilde{x}$	$\bar{x}$	N_{c}	$\widetilde{x}$	$\bar{x}$	N_{c}	$\widetilde{x}$	$\bar{x}$	N_{c}	$\widetilde{x}$
Piccadilly	80	2152	3.2	1905	1.0	9e3	4.1	1932	0.6	**5e1**	1965	**0.3**	9e3	1956	0.7	7e2	1638	10
Union Square	300	789	9.9	700	3.3	3e3	5.6	702	3.5	5e2	699	3.2	2e2	710	3.4	**9e1**	521	10
Roman Forum	200	1084	6.9	973	1.5	3e4	6.1	981	0.3	4e1	1000	2.7	9e5	989	**0.2**	**3e0**	840	37
Vienna Cathedral	120	836	5.5	758	0.9	9e3	6.6	757	0.5	**8e3**	770	0.7	7e4	770	**0.4**	2e4	652	12
Piazza del Popolo	60	328	1.8	311	**1.2**	2e1	3.1	303	2.6	**4e0**	317	1.6	9e1	308	2.2	2e2	93	16
NYC Library	130	332	1.7	297	1.5	7e2	2.5	292	0.9	2e1	307	**0.2**	8e1	295	0.4	**1e0**	271	1.4
Alamo	70	577	1.0	528	0.2	7e3	1.1	521	0.3	**7e0**	541	0.2	7e5	529	0.3	2e7	422	2.4
Metropolis	200	341	6.0	282	0.5	1e3	9.9	288	1.2	**9e0**	292	0.6	3e1	291	0.5	7e1	240	18
Yorkminster	150	437	7.0	405	0.2	3e3	3.4	395	0.2	1e4	416	0.4	9e3	401	0.1	**5e2**	345	6.7
Montreal N.D.	30	450	0.9	431	0.2	4e3	2.5	425	0.9	1e0	431	**0.1**	**4e-1**	427	0.4	1e0	357	9.8
Tower of London	300	572	9.4	417	1.1	2e3	11	414	0.4	3e3	427	**0.2**	3e4	414	1.0	**4e1**	306	44
Ellis Island	180	227	4.1	211	0.4	4e0	3.7	213	0.4	4e1	213	0.3	3e0	214	0.3	3e0	203	8.0
Notre Dame	300	553	19	524	**0.7**	2e4	10	500	2.1	**7e0**	530	0.8	7e4	507	1.9	**7e0**	473	2.1

solution, and often a much smaller distance. In general, bundle adjustment significantly reduces error. The effect of 1DSfM shows up clearly in the average error, which usually is reduced by orders of magnitude. While we see that both robust cost functions and 1DSfM improve reconstructions, they are not interchangeable—rather, 1DSfM is able to greatly reduce average error, while robust cost functions usually increase it, while decreasing median error. These two approaches cope with outliers in complementary ways, and so we advocate using both 1DSfM and a Huber loss function (as mentioned earlier, we found that other, non-convex loss functions performed poorly).

Qualitatively, our reconstructions have high quality; visualizations of many of the results are shown in Figure 3. Finally, Figure 4 shows our largest reconstruction, **Trafalgar**, with 4591 images. This model was computed with 1DSfM and bundle adjustment. The cameras have a median error of about 0.60 meters compared to sequential SfM, and it took about 3.4 hours to run, compared to 8.1 hours for sequential SfM.

Table 3 shows timing information for the experiments in Table 2, comparing our method especially with sequential SfM. All experiments were run on a machine with two 2.53 GHz Intel Xeon E5540 quad core processors. Our method is always faster than sequential SfM, usually 2-4 times faster, with even bigger improvements on larger datasets such as **Piccadilly**. The majority of our time in each dataset is spent on bundle adjustment, although unlike sequential SfM we only need to do a single large bundle adjustment, rather than many repeated ones.

Comparison to [11]. We also compared our results to [11], which solves the translations problem by minimizing the cross product of solution translations with input pairwise translations. To avoid bias from the cross product, this linear method is wrapped in an iterative reweighting framework. We used our own SciPy implementation, on the same machine as the other trials. In a slight departure, we use only three rounds of reweighting rather than four, since with each round of reweighting the underlying linear system

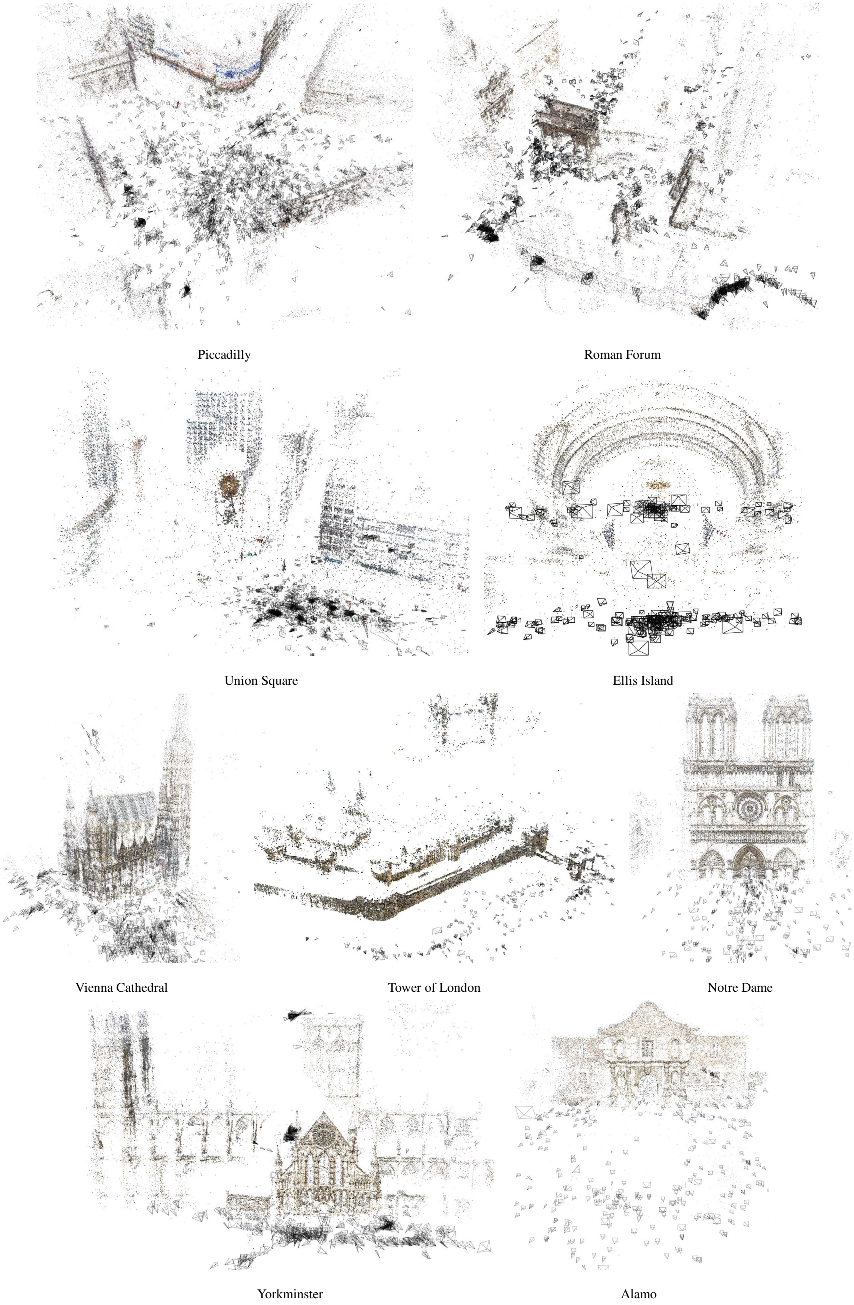
Table 3. Timing information, in seconds for the results in Table 2. Times are listed for solving for rotations with [5] (T_R), removing outliers with 1DSfM (T_O), running a translations problem solver (T_S), and for bundle adjustment (T_{BA}).

		without 1DSfM			with 1DSfM				using [11]			using [19]
Name	T_R	T_S	T_{BA}	Σ	T_O	T_S	T_{BA}	Σ	T_S	T_{BA}	Σ	T
Piccadily	570	177	3252	3999	122	366	2425	3483	9497	1046	11113	44369
Union Square	17	71	401	489	20	75	340	452	277	150	444	1244
Roman Forum	37	104	1733	1874	40	135	1245	1457	290	694	1021	4533
Vienna Cathedral	98	225	3611	3934	60	144	2837	3139	1282	893	2273	10276
Piazza del Popolo	14	28	213	255	9	35	191	249	98	26	138	1287
NYC Library	9	38	382	429	13	54	392	468	21	190	220	3807
Alamo	56	96	646	798	29	73	752	910	1039	308	1403	1654
Madrid Metropolis	15	32	224	271	8	20	201	244	57	67	139	1315
Yorkminster	11	60	955	1026	18	93	777	899	81	302	394	3225
Montreal Notre Dame	17	76	1043	1136	22	75	1135	1249	25	382	424	2710
Tower of London	9	52	750	811	14	55	606	648	17	238	264	1900
Ellis Island	12	17	276	305	7	13	139	171	7	108	127	1191
Notre Dame	53	152	2139	2344	42	59	1445	1599	299	841	1193	6154

becomes increasingly poorly conditioned. We evaluated [11] on translations problems produced by 1DSfM, reporting the results after bundle adjustment, since this combination gave the best results. Median error is reported in Table 2 and timing information in Table 3. While [11] is usually faster than our method (especially on smaller problems), the accuracy (and number of reconstructed cameras) greatly suffers.

Discussion. Our method has at its core a nonlinear optimization framework that we have found to be particularly effective, even with random initialization (once outliers are removed). Our analysis of convergence in Section 5 suggests reasons for this, but understanding fully the convergence properties of translations problems is still an interesting avenue for future work. As we noted previously, the same analysis extends to L_1 style robust cost functions as well. We believe our work points to nonlinear optimization being reconsidered as a tool for structure-from-motion beyond bundle adjusting a good solution. It is also instructive to contrast ours with other global methods. In particular, the recent linear method by Jiang et al. works on very different principles to ours, both in addressing outliers and in its efficient linear optimization framework built on triplets. Other recent methods use more sophisticated optimization methods (including discrete optimization) [6]. We believe a strength of our method is its simplicity—it relies on a well-studied combinatorial optimization problem, and a simple non-linear solver.

Limitations. Our method is based on averaging epipolar geometries to compute an accurate initialization. This works well when there are many EGs to reason about. However, sometimes EGs are sparse, such as when scenes are poorly connected. Averaging very few measurements may not be accurate. Figure 5 shows a failure case of our method. A correct reconstruction from [19] is on the left, and our broken solution is on the right. The scene has a central building with smaller domed buildings on each side. This scene is challenging because of the wide baseline between the buildings, and the similar appearance of the domes. There are few EGs that connect cameras which view different buildings. A second limitation is that our method does not reason about self-


Piccadilly

Roman Forum

Union Square

Ellis Island

Vienna Cathedral

Tower of London

Notre Dame

Yorkminster

Alamo

Fig. 3. Selected renders of models produced by our method

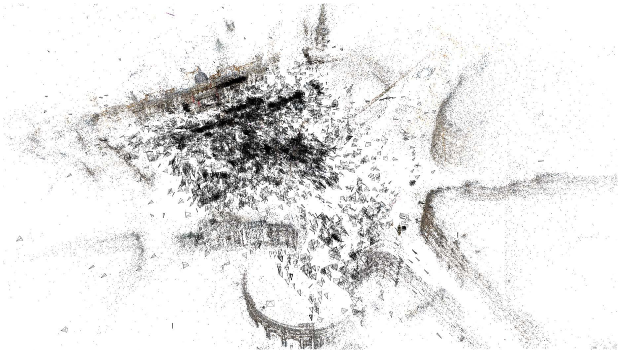


Fig. 4. A large reconstruction of Trafalgar Square containing 4597 images

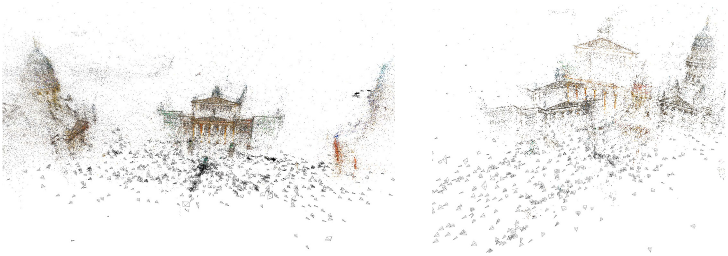


Fig. 5. (a) A correct model of **Gendarmenmarkt** from [19]. (b) A broken model by our method.

consistent outliers, such as those arising from ambiguous structures in the scene. To deal with these cases, SfM disambiguation methods could be used [23,15,22].

8 Conclusion

We presented a new method for solving the global SfM translations problem. Our method has two pieces: 1DSfM, a a method for removing outliers by solving 1-dimension ordering problems, and a simple translations solver based on squared chordal distance. Like other global methods, it treats images equally and runs faster than common sequential methods. Our method stands out by being particularly simple, and represents a different take on the problem from previous methods which focus on linear formulations.

We have demonstrated the effectiveness of our method on a range of datasets in the wild; these are available, along with code, at `http://www.cs.cornell.edu/projects/1dsfm`. We produce models comparable to existing sequential methods in much less time. In the future we hope to explore further ways of aggregating 1DSfM subproblems than simple summation, which could shed light on more complicated outliers, such as those arising from ambiguous scene structures.

Acknowledgements. This work was funded in part by NSF grants IIS-1149393 and IIS-0964027, and by support from the Intel Science and Technology Center for Visual Computing. We would also like to thank David Williamson and Robert Kleinberg for their valuable help.

References

1. Agarwal, S., Mierle, K., et al.: Ceres solver, `https://code.google.com/p/ceres-solver/`
2. Agarwal, S., Snavely, N., Simon, I., Seitz, S.M., Szeliski, R.: Building Rome in a day. In: ICCV (2009)
3. Arie-Nachimson, M., Shahar, S.Z., Kemelmacher-Shlizerman, I., Singer, A., Basri, R.: Global motion estimation from point matches. In: 3DIMPVT (2012)
4. Brand, M., Antone, M., Teller, S.: Spectral solution of large-scale extrinsic camera calibration as a graph embedding problem. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3022, pp. 262–273. Springer, Heidelberg (2004)
5. Chatterjee, A., Govindu, V.M.: Efficient and robust large-scale rotation averaging. In: ICCV (2013)
6. Crandall, D., Owens, A., Snavely, N., Huttenlocher, D.: Discrete-continuous optimization for large-scale structure from motion. In: CVPR (2011)
7. Eades, P., Lin, X., Smyth, W.F.: A fast and effective heuristic for the feedback arc set problem. In: Information Processing Letters (1993)
8. Enqvist, O., Kahl, F., Olsson, C.: Nonsequential structure from motion. In: OMNIVIS (2011)
9. Frahm, J.-M., et al.: Building Rome on a cloudless day. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 368–381. Springer, Heidelberg (2010)
10. Fredriksson, J., Olsson, C.: Simultaneous multiple rotation averaging using Lagrangian duality. In: Lee, K.M., Matsushita, Y., Rehg, J.M., Hu, Z. (eds.) ACCV 2012, Part III. LNCS, vol. 7726, pp. 245–258. Springer, Heidelberg (2013)
11. Govindu, V.M.: Combining two-view constraints for motion estimation. In: CVPR (2001)
12. Govindu, V.M.: Lie-algebraic averaging for globally consistent motion estimation. In: CVPR (2004)
13. Hartley, R., Aftab, K., Trumpf, J.: L_1 rotation averaging using the Weiszfeld algorithm. In: CVPR (2011)
14. Jiang, N., Cui, Z., Tan, P.: A global linear method for camera pose registration. In: ICCV (2013)
15. Jiang, N., Tan, P., Cheong, L.: Seeing double without confusion: Structure-from-motion in highly ambiguous scenes. In: CVPR (2012)
16. Kahl, F.: Multiple view geometry and the L_∞-norm. In: ICCV (2005)
17. Martinec, D., Pajdla, T.: Robust rotation and translation estimation in multiview reconstruction. In: CVPR (2007)
18. Sinha, S.N., Steedly, D., Szeliski, R.: A multi-stage linear approach to structure from motion. In: Kutulakos, K.N. (ed.) ECCV 2010 Workshops, Part II. LNCS, vol. 6554, pp. 267–281. Springer, Heidelberg (2012)
19. Snavely, N., Seitz, S., Szeliski, R.: Photo tourism: Exploring photo collections in 3D. In: SIGGRAPH (2006)
20. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: A factorization method. In: IJCV (1992)
21. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment—a modern synthesis. In: Triggs, B., Zisserman, A., Szeliski, R. (eds.) Vision Algorithms 1999. LNCS, vol. 1883, pp. 298–372. Springer, Heidelberg (2000)
22. Wilson, K., Snavely, N.: Network principles for SfM: Disambiguating repeated structures with local context. In: ICCV (2013)
23. Zach, C., Klopschitz, M., Pollefeys, M.: Disambiguating visual relationships using loop constraints. In: CVPR (2010)