Noname manuscript No.
(will be inserted by the editor)

Simultaneous Camera Pose and Correspondence Estimation with Motion Coherence

Wen-Yan Lin · Loong-Fah Cheong · Ping Tan · Guo Dong · Siying Liu

Received: date / Accepted: date

Abstract Traditionally, the camera pose recovery problem has been formulated as one of estimating the optimal camera pose given a set of point correspondences. This critically depends on the accuracy of the point correspondences and would have problems in dealing with ambiguous features such as edge contours and high visual clutter. Joint estimation of camera pose and correspondence attempts to improve performance by explicitly acknowledging the chicken and egg nature of the pose and correspondence problem. However, such joint approaches for the two-view problem are still few and even then, they face problems when scenes contain largely edge cues with few corners, due to the fact that epipolar geometry only provides a "soft" point to line constraint. Viewed from the perspective of point set registration, the point matching process can be regarded as the registration of points while preserving their relative positions (i.e. preserving scene coherence). By demanding that the point set should be transformed coherently across views, this framework leverages on higher level perceptual information such as the shape of the contour. While thus potentially allowing registration of non-unique edge points, the registration framework in its traditional form is subject to substantial point localization error and is thus not suitable for estimating camera pose. In this paper, we introduce an algorithm which jointly estimates camera pose and correspondence within a point set registration framework based on motion coherence, with the camera pose helping to localize the edge registration, while the "ambiguous" edge information helps to guide camera pose computation. The algorithm can compute camera pose over large displacements and by utilizing the non-unique edge points can recover camera pose from what were previously regarded as feature-impoverished SfM scenes. Our algorithm is also sufficiently flexible to incorporate high dimensional feature descriptors and works well on traditional SfM scenes with adequate numbers of unique corners.

Keywords Structure from Motion · Registration

1 Introduction

The process of recovering 3-D structure from multiple images of the same scene is known in the vision community as the Structure from Motion (SfM) problem. One central issue that must be addressed in solving SfM is camera pose recovery. Traditionally, the camera pose recovery problem has been formulated as one of estimating the optimal camera pose given a set of point correspondences. Such approach includes, among many others, improved linear estimation [22,43], bundle adjustment [53] as well as globally optimal estimators [12,26]. However, despite many advances in matching techniques [3,21,35], obtaining correspondences across two images remains a non-trivial problem and contains a strong underlying assumption that the features are

This work is partially supported by project grant NRF2007IDM-IDM002-069 on "Life Spaces" from the IDM Project Office, Media Development Authority of Singapore and DSO grant R263-000-400-592.

First, Second and Third Author
Electrical and Computer Engineering Department,
National University of Singapore
4 Engineering Drive 3, Singapore 117576
E-mail: g0600124, eleclf, ptan@nus.edu.sg

Fourth Author
DSO National Laboratories,
20 Science Park Drive,
Singapore 118230
E-mail: gdong@dso.org.sg

Fifth Author
Institute for Infocomm Research,
1 Fusionopolis Way. #21-01 Connexis, South Tower.
Singapore 138632
E-mail: sliu@i2r.a-star.edu.sg

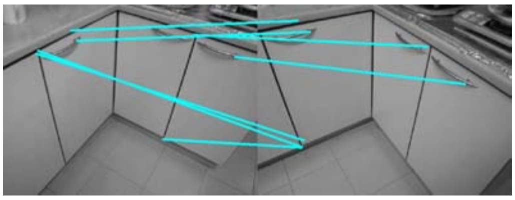


Fig. 1 This scene illustrates the difficulty in obtaining reliable matches when there are few corners. The correspondences are the results of a SIFT matcher. There are insufficient corners available for matching, with most of the few corners available suffering from ambiguity. Pose recovery on these scenes would be substantially easier if we could use the clear contour cues present.

sufficiently distinct to enable unique point to point correspondence. This limits camera pose recovery to well textured scenes with abundant corner features. In this paper, we seek to design an algorithm which can incorporate ambiguous features such as edge points into the camera pose recovery process. This allows pose recovery on more challenging SfM scenes where there are few corners; such scenes are particularly common in man-made environment [56, 57], one example of which is illustrated in Figure 1. Our algorithm is, however, not limited to such scenes. Natural scenes where the visual features are highly similar or whose extraction is non-repeatable across large viewpoint change can also benefit from our approach.

While correspondence is needed to obtain camera pose, knowledge of camera pose also facilitates point correspondence. In recent years, a number of works [9, 18, 30, 36, 47] have proposed joint pose and correspondence algorithms (JPC) which explicitly acknowledge the chicken and egg nature of the pose and correspondence problem. Rather than choosing a camera pose in accordance with a pre-defined set of matches, these algorithms choose camera pose on the basis of whether the feature points can find a correspondence along the associated epipolar line. This permits the utilization of non-unique features to contribute to camera pose computation. Note that we should distinguish such JPC works from other joint estimation works such as 2D image or 3D surface registration [5, 11, 31, 59] using say, the Iterative Closest Point (ICP) technique. These registration works invariably involve a global transformation that is parameterized by a few variables (such as the affine parameters) and provides a well-defined mapping from point to point. This one-to-one mapping means the global parameters automatically preserves the relative alignment of features and largely accounts for the success in solving the registration. In contrast, in the JPC algorithms, the 3D camera pose does not define a point to point correspondence but rather a point to epipolar line relationship on the 2D image plane. This additional ambiguity means a much greater degree of freedom and associated problem complexity. More importantly for our problem scenario where the features are highly ambiguous, it also means that the epipolar constraint alone is insufficient to resolve the ambiguity, even with the JPC approach. For example, if the feature points consisted of edge pixels that form a long connected contour, an epipolar line in any direction will eventually intersect with the contour. Thus, a JPC algorithm will have difficulty choosing a correct camera pose.

Despite such apparent ambiguity, we note that the motion-induced deformation of a 2D contours' shape contains clear perceptual cues as to the relative camera pose. One possible reason that humans can infer the camera pose might be that they perceive the contour points as a collective entity in motion (i.e. the law of shared common fate), rather than as independently moving individual points. This motivates us to impose a coherent motion constraint on the feature point displacements such that the displacements approximately preserve the overall shape of these points; in other words, points close to one another should move coherently [58].

While general non-rigid registration algorithms such as [7, 42] are generally able to preserve the overall shape of a point set, they are not designed for point-to-point correspondence and suffer from the aperture problem. As was shown in our preliminary work [32], individual contour points are poorly localized using the registration algorithm proposed in [42]. The registration is not consistent with any epipolar geometry and, hence, is not useful for obtaining camera pose.

In this paper, we propose jointly estimating the camera pose and point correspondence while enforcing a coherent motion constraint. Such joint estimation scheme is complex because the goodness of any point match depends not only on the camera pose and its local descriptor, but also on the matching position allocated to all other image points. The complexity is further increased because the smooth coherent motion of a contour is essentially a continuous concept, but we wish to work on discrete point sets containing possibly both corners and edge information. We adapt for this purpose the Coherent Point Drift framework of [42], which overlaid a continuous displacement field over the sparse point set, and regularized the displacement field to achieve motion coherence. The resultant scheme can compute camera pose using "ambiguous" features such as edge points (as well as the conventional corner points). It also removes the localization uncertainty of the edge point correspondence from using registration algorithm. This is illustrated in Figure 2. To our knowledge, this is the first attempt to integrate motion coherence, correspondence over a sparse point set and camera pose estimation into a common framework. The result makes a big difference in the perceived difficulty of a SfM scene. Our experiment showed that our algorithm can work well across large viewpoint changes, on scenes which primarily consist of long edges and few corners, as well as natural scenes with high visual clutter.

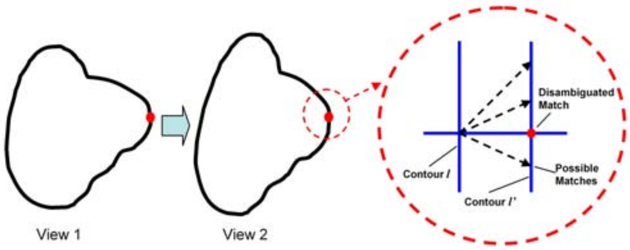


Fig. 2 The dotted region represents the localization uncertainty present in the matching provided by the registration algorithm. On the right, the horizontal epipolar line allows the localization of the contour point.

1.1 Related works

The core concept of using an iterative refinement of pose and correspondence has a long and rich history in SfM. Examples include RANSAC-flavored algorithms [15,20,39], and the Joint Pose and Correspondence/Flow algorithms [36,30, 9,47,18,44,49,55]. Many of these are landmark works which greatly improve SfM's stability in previously difficult scenes.

A large number of JPC works can be classified as working on discrete features like corners. The optimality of a specific pose is evaluated on the basis of whether its associated epipolar constraint permits good correspondence. A variety of methods are then proposed to minimize the cost function, such as the Radon transforms in [36], line matching in the Fourier space [30], or an EM-based search of the space [9]. The problem of searching the solution space can also be simplified and made more robust (though at the expense of generality) by introducing restrictions on anticipated scene type or camera motion. As such, a number of these works focus on the affine camera [47,30], while others like [18] restricted the camera to motion on a plane, thus limiting potential camera pose space to a 2-dimensional region. Also related to the JPC scheme is [17], where although pre-computed correspondence is used, the correspondences' photometric properties are used to help restrict camera pose. Our work also focuses on using pre-detected features and edges, however, unlike the previously discussed works, we incorporate a smoothness term, thus permitting the handling of ambiguous edge features and their associated aperture problem over large displacements.

Conceptually, our work is similar to flow based JPC algorithms such as [49,44,55]. They overcome the aperture problem by finding an optical flow that is consistent with a camera motion. While this approach can be extended to wider baselines by applying a point set registration algorithm as initialization, such an algorithm would be inelegant and is likely to suffer from large amount of noise caused by approximating a large displacement by flow. Our approach handles large displacement naturally. It also handles the problem of disconnected point sets and isolated corners more naturally than that of optical flow formulation and would be especially useful in incorporating recently proposed edge descriptors [38]. Lastly, our approach can incorporate high-dimensional feature descriptors which give greater robustness to photometric noise.

There are many other works that jointly estimate a global transformation between two sets of points and the point correspondence between them, but they differ from our work in some important aspects. Some of these involve multiple frames [10,28,41], where an initial 3D map was built from say, five-point stereo [43]. Subsequent camera poses were tracked using local bundle adjustment over the N most recent camera poses, and features are constantly added to allow the 3D map to grow in the SLAM style. In other works, the 3D models are available a priori (e.g. from a CAD model) [8,27,40,13]. In contrast, our joint estimation is carried out over two frames in the absence of any 3D model or initial map. Other joint estimation works [5,25,31,46,59] involve aligning two sets of points which are related by some simple transformations defining a point to point mapping. The one to one mapping automatically preserves the relative alignment of the features within the point set without having a need for an additional coherence constraint. Our work differs in that the epipolar geometry does not enforce a one-to-one mapping. Instead, the unknown depth of the feature points means that the camera pose provides a point to epipolar line constraint. It also means that an additional coherence term is needed to enforce a greater coherence of shape, leading to a significantly more complex problem formulation.

For multiple views, it is also possible to make use of structure from lines algorithms to overcome the aperture problem [2,14,23,50]. Interested readers might like to peruse other works dealing with various aspects of curve / line reconstruction [1,6,24,51,56,57] as well as the merger of intensity and edge information [37,45,54].

2 Formulation

In this paper, the problem addressed is the recovery of cameras' relative pose (i.e. orientation and position) given two different views of a static scene. The formulation emphasizes generality, allowing easy adaptation for different inputs such as corners and edges. Edges are simply described by point sets obtained by sampling the edge map of the image.

2.1 Definitions

Each feature point takes the form of a D dimensional feature vector,

$$\left[x \; y \; r \; g \; b \ldots \right]^T_{1 \times D},$$

with x and y being image coordinates, while the remaining optional dimensions can incorporate other local descriptors such as color, curvature, etc. We are given two point sets. A **base** point set $\mathbf{B_0}_{M\times D} = [b_{01}, ..., b_{0M}]^T$ describing M feature points in the base image and a **target** point set $\mathbf{T_0}_{N\times D} = [t_{01}, ..., t_{0N}]^T$ describing N feature points in the target image. b_{0i}, t_{0i} are D dimensional point vectors of the form given above.

We define another matrix $\mathbf{B}_{M\times D} = [b_1, ..., b_M]^T$ which is the evolved version of $\mathbf{B_0}$. We seek to evolve $\mathbf{B}$ until it is aligned to the target point set $\mathbf{T_0}_{N\times D}$, while still preserving the coherence of $\mathbf{B_0}$ (that is, the overall 2D geometric relationships between points in $\mathbf{B_0}$ should be preserved as much as possible). The evolution of $\mathbf{B}$ consists of changing only the image coordinates (first two entries) of the b_i vectors. The remaining entries are held constant to reflect the brightness/ feature constancy assumption. When attempting to align the evolving base set $\mathbf{B}$ to the target set $\mathbf{T_0}$, we try to ensure that the resulting mapping of the image coordinates of b_{0i} to b_i are consistent with that of a moving camera viewing a static scene (i.e. abide by some epipolar constraint).

As many equations only involve the first two dimensions of b_{0i}, b_i, to simplify our notation, we define them as the sub-vectors β_{0i}, β_i respectively. We further denote the first two columns of $\mathbf{B}_0$ and $\mathbf{B}$ by $\mathfrak{B}_0$ and $\mathfrak{B}$, which are $M \times 2$ matrices formed by β_{0i} and β_i. As $\mathfrak{B}_\mathbf{0}$ and $\mathfrak{B}$ uniquely define $\mathbf{B_0}$ and $\mathbf{B}$ respectively in our case, the matrices can often be used interchangeably in probabilities and function declarations. The constancy of much of the b_i vector also means that the algorithm's run time is largely independent of the size of D. Hence one can apply high dimensional descriptors on the contour points with little additional cost.

2.2 Problem formulation

We seek an aligned base set $\mathbf{B}$ and the associated motion of an uncalibrated camera $\mathbf{F}$ (for calibrated cameras, one could parameterize $\mathbf{F}$ using the rotation and translation parameters without changing the formulation), which has maximum likelihood given the original base and target point sets $\mathbf{B_0}$ and $\mathbf{T_0}$ respectively. Mathematically, this can be expressed as maximizing $P(\mathbf{B}, \mathbf{F}|\mathbf{B_0}, \mathbf{T_0})$. Using Bayes' rule, this can be formulated as,

$$\begin{aligned} P(\mathbf{B}, \mathbf{F}|\mathbf{B_0}, \mathbf{T_0}) &= \frac{P(\mathbf{T_0}, \mathbf{B}|\mathbf{F}, \mathbf{B_0})P(\mathbf{F}, \mathbf{B_0})}{P(\mathbf{B_0}, \mathbf{T_0})} \\ &= \frac{P(\mathbf{T_0}, \mathbf{B}|\mathbf{F}, \mathbf{B_0})P(\mathbf{F}|\mathbf{B_0})P(\mathbf{B_0})}{P(\mathbf{B_0}, \mathbf{T_0})} \end{aligned}$$

It is clear that the likelihoods $P(\mathbf{B_0}), P(\mathbf{B_0}, \mathbf{T_0})$ are constants with respect to the minimization variables $\mathbf{F}, \mathbf{B}$. Furthermore, if we assume a uniform (un-informative) prior for the motion, it makes sense to assign $P(\mathbf{F}|\mathbf{B_0})$ to be a constant[1]. This allows us to simplify the probabilistic expression into

$$\begin{aligned} P(\mathbf{B}, \mathbf{F}|\mathbf{B_0}, \mathbf{T_0}) &\propto P(\mathbf{T_0}, \mathbf{B}|\mathbf{F}, \mathbf{B_0}) \\ &= P(\mathbf{B}|\mathbf{F}, \mathbf{B_0})P(\mathbf{T_0}|\mathbf{B}, \mathbf{F}, \mathbf{B_0}). \end{aligned} \quad (1)$$

Observe that by expressing our formulation in terms of a warping from a base image to a target image, we treat the information from the two views in an asymmetrical manner. A symmetrical formulation may be able to better handle spurious feature and validate whether the algorithm has converged to an adequate minimum. However, the resultant scheme will be complex and is beyond the scope of this paper.

We first study the term $P(\mathbf{B}|\mathbf{F}, \mathbf{B_0})$. Given camera pose $\mathbf{F}$ and assuming independent isotropic Gaussian noise of standard deviation σ_b, the evolving base point set $\mathbf{B}$ has an associated probability given by the "improper" (the non-essential scale is dropped for simplicity) distribution

$$P(\mathbf{B}|\mathbf{F}, \mathbf{B_0}) = P(\mathfrak{B}|\mathbf{F}, \mathbf{B_0}) = e^{-\lambda \Psi(\mathfrak{B})} \prod_{i=1}^{M} g(d_i, \sigma_b). \quad (2)$$

where $g(z, \sigma) = e^{-\frac{\|z\|^2}{2\sigma^2}}$ is a Gaussian function. The first, $e^{-\lambda \Psi(\mathfrak{B})}$ term is a coherence term which we discuss in Section 2.3, while the second term contains the epipolar constraint, with d_i denoting distance from the epipolar line, with the detailed discussion in Section 2.4.

2.3 Coherence term

The first exponent in equation (2) contains the regularization term $\Psi(\mathfrak{B})$ with λ controlling the relative importance of this regularization term.

Recall that we desire to enforce smoothness over a discrete point set whose points are sparsely distributed, a rather difficult operation to perform. One option is to directly penalize any deviation in the relative position of points considered as neighbors. Such an approach fits naturally into the discrete point set problem and is amenable to graph based minimization [48,52]. However, because only the first order smoothness is imposed, it tends to penalize all deviations in relative position, rather than penalizing discontinuous changes in shape much more heavily than smooth deformation in shape caused by viewpoint changes. In other words, such first-order smoothness does not supply enough coherence of shape.

To overcome the aforementioned difficulties, we define a fictitious continuous field over the sparse point set and call

[1] An intuitive explanation for a uniform prior is that a camera can move to any position in the 3D world and similarly have any calibration parameters.

it the displacement field or velocity field (in this paper, the terms velocity and displacement are used loosely and do not imply any small motion approximation for the former). We utilize the motion coherence framework of [58] in which higher order of smoothness is enforced on the velocity field. The smoothness is imposed mathematically by regularization in the Fourier domain of the velocity field. Our scheme has a number of advantages:

1. By imposing higher-order smoothness, it permits smooth changes in relative position that nevertheless maintains coherence in shape, rather than penalizing all changes. In fact, [58] explicitly showed that for isolated features, a smoothing operator with only first-order derivatives does not supply enough smoothness for a well-posed solution.
2. The formulation of this fictitious velocity field acts as a unifying principle for all types of motion information (isolated features, contours, brightness constancy constraint). It allows us to integrate the information provided by isolated features and contours, and yet does not require the declaration of a specific region of support when deciding which points are neighbors that should influence each others' motion.
3. While the interaction of the velocity field falls off with distance and is thus local, we obtain a resultant interaction between the isolated features that is nonlocal. This is desirable on account of the Gestalt principle. On the other hand, when there is local motion information that suggests discontinuous change in the velocity field, the rapidly falling off local interaction of the velocity field will ensure that it will be the locally measured data that are most respected, thus allowing discontinuous change in the velocity field. Preservation of such discontinuous changes is further aided by additional mechanisms introduced in the regularization scheme (more of that, later).

We define $v(.)$ as this 2D velocity field function. The velocity field covers the entire image, and at image locations β_{0i} where feature points exist, it must be consistent with the feature points' motion. Mathematically, this means that they obey the constraint

$$\beta_i = v(\beta_{0i}) + \beta_{0i}. \tag{3}$$

$\Psi(\mathfrak{B})$ is defined in the Fourier domain to regularize the smoothness of the velocity field function $v(.)$:

$$\Psi(\mathfrak{B}) = \min_{v'(s)} \left(\int_{\Re^2} \frac{|v'(s)|^2}{g'(s) + \kappa'(s)} ds \right), \tag{4}$$

where $v'(s)$ is the Fourier transform of the velocity field $v(.)$ which satisfies equation (3) and $g'(s)$ is the Fourier transform of a Gaussian smoothing function. The Gaussian function has a spatial standard deviation of γ which controls the amount of coherence desired of the velocity field. Without the $\kappa'(s)$ term, the above smoothness function follows the motion coherence form proposed in [58] and has been used in general regularization theory [19]; it was also subsequently adopted in the contour registration work of [42]. Such definition allows us to impose a continuous coherent motion field over the motion of a discrete point set specified by equation (3). Suppressing the high frequency components of the velocity field ensures that adjacent contour points have similar motion tendencies, thus preserving the overall 2D geometric relationships between points in $\mathbf{B_0}$. However, the Gaussian function drops off very sharply away from the mean, greatly penalizing the high frequency terms. In SfM where there may be occlusion and sharp velocity changes, such a penalty function can be overly restrictive. As such, we introduce the additional $\kappa'(s)$ term, which should have limited spatial support and hence wide frequency support. In this paper, spatial support is taken to be less than the smallest separation between any two points in $\mathbf{B_0}$. Given such limited spatial support, the exact form of the function κ is immaterial. We simply specify that $\kappa(.)$ must have the property:

$$\kappa(\beta_{0i} - \beta_{0j}) = \begin{cases} k, & i = j \\ 0, & i \neq j \end{cases} \tag{5}$$

where k is some pre-determined constant.

2.4 Epipolar term

The second term in equation (2) contains the epipolar constraint defined by camera pose, $\mathbf{F}$. As mentioned earlier, we desire that the image coordinate pairs β_{0i}, β_i, to be consistent with $\mathbf{F}$. Hence, d_i is the perpendicular distance of the point β_i from the epipolar line defined by point β_{0i} and pose $\mathbf{F}$, with a cap at ζ. Observe that since β_{0i} is a fixed point of unknown depth, d_i is the geometric error [23] associated with $\beta_{0i}, \beta_i, \mathbf{F}$, with an additional capping function. The capping function basically expresses the fact that the Gaussian noise error model is only valid for inlier points, while there exist a number of randomly distributed outlier points which result in much thicker tails than are commonly assumed by the Gaussian distribution.

Practically, such robust functions allow outliers to be removed from consideration by paying a certain fixed penalty. In this regards, its function is similar to statistical form of RANSAC [53]. Formally, the capped geometric distance can be written as

$$d_i = \min(\left\| l_i^T(\beta_i - r_i) \right\|, \zeta) \tag{6}$$

where r_i is a two dimensional vector representing any point on the epipolar line. l_i is a two dimensional unit vector perpendicular to the epipolar line defined by $\mathbf{F}$ and β_{0i}. ζ is the maximum deviation of a point from the epipolar line, before it is considered an outlier. As our point sets often contain huge numbers of outliers, we usually set ζ to a very low value of 0.01 (the distance is defined in the normalized image space after Hartley's normalization [22]).

2.5 Registration term and overall cost function

We now consider $P(\mathbf{T_0}|\mathbf{B},\mathbf{F},\mathbf{B_0})$ in equation (1). Since $\mathbf{T_0}$ is independent of the ancestors $\mathbf{F}$ and $\mathbf{B_0}$ given the immediate parent $\mathbf{B}$, this probability can be simplified to just the confidence measure of $\mathbf{T_0}$ given $\mathbf{B}$. Note that the $\mathbf{T_0}$ and $\mathbf{B}$ contain a mix of descriptor and coordinate terms. We let each b_i be the D dimensional centroid of an equivariant Gaussian function with standard deviation σ_t (we assume that the data has been pre-normalized, the normalization weights being given in section 3.3). The following forms the Gaussian mixture probability of $\mathbf{T_0}$:

$$P(\mathbf{T_0}|\mathbf{B},\mathbf{F},\mathbf{B_0}) = \prod_{j=1}^{N}\sum_{i=1}^{M} g(t_{0j}-b_i,\sigma_t). \tag{7}$$

This is the registration error term which includes both geometric and intensity information for the entire set of features but does not force a strict one-to-one feature correspondence. Initially, $\mathbf{B}$ is not necessarily close to $\mathbf{T_0}$, thus making the above probability very small. However, using the Expectation Maximization (EM) algorithm, we use these initial, low probabilities to better align $\mathbf{B}$ with $\mathbf{T_0}$. Note that we use the term EM loosely to describe the general minimization style although the exact mechanism is slightly unconventional.

Substituting equations (2) and (7) into (1) and taking the negative log of the resultant probability, our problem becomes one of finding the $\mathbf{F}$ and $\mathfrak{B}$ which maximize the probability in equation (1), or equivalently, minimize $A(\mathbf{B},\mathbf{F})$, where

$$A(\mathbf{B},\mathbf{F}) = -\sum_{j=1}^{N} log \sum_{i=1}^{M} g(t_{0j}-b_i,\sigma_t) + \sum_{i=1}^{M}\frac{d_i^2}{2\sigma_b^2} + \lambda\Psi(\mathfrak{B}). \tag{8}$$

The first term in $A(\mathbf{B},\mathbf{F})$ measures how well the evolving point set $\mathbf{B}$ is registered to the target point set $\mathbf{T_0}$. The second term measures whether the evolving point set $\mathbf{B}$ adheres to the epipolar constraint. Finally, the third term ensures that the point set $\mathbf{B}$ evolves in a manner that approximately preserves the coherence of $\mathbf{B_0}$.

3 Joint estimation of correspondence and pose

We seek the $\mathfrak{B}$ and $\mathbf{F}$ which optimize equation (8) (recall that $\mathfrak{B}$ is the first two columns of $\mathbf{B}$). Observe that this is a constrained minimization but as the l_i, r_i terms in the geometric distance d_i have a non-linear relationship with the camera pose $\mathbf{F}$ and image point β_{0i}, as well as due to the presence of the regularization term, it precludes other more straightforward minimization techniques. Using a method similar to expectation maximization, we minimize $A(\mathbf{B},\mathbf{F})$ by alternately updating $\mathfrak{B}$ and $\mathbf{F}$. The procedure is described in the following subsections.

3.1 Updating registration, $\mathfrak{B}$

In this subsection, we hold the camera pose $\mathbf{F}^{old}$ constant while updating $\mathfrak{B}$. This results in a $\mathfrak{B}^{new}$ whose associated evolving base point set $\mathbf{B}^{new}$ is better aligned to the target point set $\mathbf{T_0}$, while preserving the point set's coherence and respecting the epipolar lines defined by the camera pose $\mathbf{F}^{old}$. The new registration $\mathfrak{B}^{new}$ can be computed from the $M\times 2$ linear equations in equation (13).

Here we provide the derivations. We define

$$\begin{aligned}\phi_{ij}(b_i,t_{0j}) &= g(t_{0j}-b_i,\sigma_t)\\ \overline{\phi_{ij}}(\mathbf{B},t_{0j}) &= \frac{\phi_{ij}(b_i,t_{0j})}{\sum_z \phi_{zj}(b_z,t_{0j})}.\end{aligned} \tag{9}$$

For more robust correspondence with occlusion, we use a robust version of $\overline{\phi_{ij}}(\mathbf{B},t_{0j})$ in equation (9). This is given by $\overline{\phi_{ij}}(\mathbf{B},t_{0j}) = \frac{\phi_{ij}(b_i,t_{0j})}{\sum_z \phi_{zj}(b_z,t_{0j})+2\mu\pi\sigma_t^2}$. The second, $2\mu\pi\sigma_t^2$ denominator term provides a thickening of the tail compared to those of the Gaussian. The idea is similar to the robust implementation of the regularization in equation (6).

Using Jensen's inequality and observing that the maximum value of d_i is ζ, we can write the inequality

$$\begin{aligned}&A(\mathbf{B}^{new},\mathbf{F}^{old}) - A(\mathbf{B}^{old},\mathbf{F}^{old})\\ &\leq -\sum_{j=1}^{N}\sum_{i=1}^{M}\overline{\phi_{ij}}(\mathbf{B}^{old},t_{0j}) log\frac{\phi_{ij}(b_i^{new},t_{0j})}{\phi_{ij}(b_i^{old},t_{0j})}\\ &\quad+\sum_{i\in inlier}\frac{(d_i^{new})^2-(d_i^{old})^2}{2\sigma_b^2}\\ &\quad+\lambda\left(\Psi(\mathfrak{B}^{new})-\Psi(\mathfrak{B}^{old})\right)\\ &=\Delta A(\mathbf{B}^{new},\mathbf{B}^{old},\mathbf{F}^{old}).\end{aligned} \tag{10}$$

where a point i is an inlier if $d_i^{old} < \zeta$.
Observing from equation (10) that $\Delta A(\mathbf{B}^{old},\mathbf{B}^{old},\mathbf{F}^{old}) = 0$, the $\mathbf{B}^{new}$ which minimizes $\Delta A(\mathbf{B}^{new},\mathbf{B}^{old},\mathbf{F}^{old})$ will ensure that

$$A(\mathbf{B}^{new},\mathbf{F}^{old}) \leq A(\mathbf{B}^{old},\mathbf{F}^{old})$$

since the worst $A(\mathbf{B}^{new},\mathbf{F}^{old})$ can do is to take on the value of $A(\mathbf{B}^{old},\mathbf{F}^{old})$.

Dropping all the terms in $\Delta A(\mathbf{B}^{new}, \mathbf{B}^{old}, \mathbf{F}^{old})$ which are independent of $\mathfrak{B}^{new}$, we obtain a simplified cost function

$$
\begin{aligned}
Q =& \frac{1}{2} \sum_{j=1}^{N} \sum_{i=1}^{M} \overline{\phi_{ij}}(\mathbf{B}^{old}, t_{0j}) \frac{\|t_{0j} - b_i^{new}\|^2}{\sigma_t^2} \\
&+ \sum_{i \in inlier} \frac{(d_i^{new})^2}{2\sigma_b^2} + \lambda \Psi(\mathfrak{B}^{new}).
\end{aligned} \tag{11}
$$

Using a proof similar to that in [42], we show in the Appendix that the regularization term $\Psi(\mathfrak{B})$ at the minima of $A(\mathbf{B}, \mathbf{F})$ is related to $\mathfrak{B}$ and $\mathfrak{B}_\mathbf{0}$ by

$$
\Psi(\mathfrak{B}) = tr(\Gamma \mathbf{G}^{-1} \Gamma^T), \tag{12}
$$

where $\mathbf{G}$ is a $M \times M$ matrix with its (i, j) entry given by $\mathbf{G}(i, j) = g(\beta_{0i} - \beta_{0j}, \gamma) + k\delta_{ij}$ (δ_{ij} being the Kronecker delta), $\Gamma = (\mathfrak{B} - \mathfrak{B}_\mathbf{0})^T$, and $tr(.)$ represents the trace of a matrix. Substituting the above expression of $\Psi(\mathfrak{B})$ into Q and taking partial differentiation of Q with respect to each element of $\mathfrak{B}^{new}$, we can construct the matrix $\frac{\partial Q}{\partial \mathfrak{B}^{new}}$, where each entry is $\frac{\partial Q}{\partial \mathfrak{B}^{new}(i,j)}$. The conditions needed for achieving the minimum of Q can be obtained by setting all the entries of this matrix to zero:

$$
\begin{aligned}
\frac{\partial Q}{\partial \mathfrak{B}^{new}} = \begin{bmatrix} c_1 \; c_2 \; \ldots \; c_{M-1} \; c_M \end{bmatrix} + 2\lambda \Gamma^{new} \mathbf{G}^{-1} &= \mathbf{0}_{2 \times M} \\
\mathbf{C} + 2\lambda \Gamma^{new} \mathbf{G}^{-1} &= \mathbf{0}_{2 \times M} \\
\mathbf{C}\mathbf{G} + 2\lambda \Gamma^{new} &= \mathbf{0}_{2 \times M}
\end{aligned} \tag{13}
$$

Here, the column vector c_i is computed as

$$
\begin{aligned}
c_i =& \sum_{j=1}^{N} \overline{\phi_{ij}}(\mathbf{B}^{old}, t_{0j}) \left(\frac{\beta_i^{new} - \hat{t}_{0j}}{\sigma_t^2} \right) \\
&+ \begin{cases} \frac{\mathbf{q_i}^{old}(\beta_i^{new} - r_i^{old})}{\sigma_b^2} & i \in \text{inlier} \\ \mathbf{0}_{2 \times 1} & \text{otherwise} \end{cases},
\end{aligned}
$$

where $\mathbf{q_i}_{2 \times 2}$ is a 2×2 matrix given by $\mathbf{q_i}_{2 \times 2} = (l_i)(l_i^T)$, $\hat{t}_{0j}$ stands for the truncated vector of t_{0j} with the latter's first two elements, and the definitions of l_i, r_i are as given in equation (6). Equation (13) produces $M \times 2$ linear equations which can be solved to obtain $\mathfrak{B}^{new}$.

Observe that the minimization step in equation (13)—in particular, the computation of c_i—is in keeping with the spirit of the outlier rejection scheme discussed in equation (6): "outliers" are no longer over-penalized by the camera pose but they remain incorporated into the overall registration framework.

3.2 Updating camera pose, $\mathbf{F}$

We now update the camera pose on the basis of the new correspondence set $\mathfrak{B}^{new}$,$\mathfrak{B}_\mathbf{0}$. Replacing $\mathbf{B}$ in equation (8) with $\mathbf{B}^{new}$ and holding it constant, we seek to minimize the cost function $A(\mathbf{B}^{new}, \mathbf{F}^{new})$ with respect to only $\mathbf{F}^{new}$. Only the middle term in $A(\mathbf{B}, \mathbf{F})$ depends on $\mathbf{F}$. Using the definition of the geometric distance d_i in equation (6), we minimize the simplified cost function

$$
\sum_{i=1}^{M} \min \left(\left\| (l_i^{new})^T (\beta_i^{new} - r_i^{new}) \right\|^2, \zeta^2 \right) \tag{14}
$$

with β_i^{new} being the image coordinates of the point set $\mathfrak{B}^{new}$.

Observe that the problem of finding the $\mathbf{F}^{new}$ which in turn produces l_i^{new} and r_i^{new} that minimize the above cost function can be formulated as a bundle adjustment problem [53] with camera pose $\mathbf{F}$ initialized to $\mathbf{F}^{old}$.

After these two steps, $\mathfrak{B}^{old}$, $\mathbf{F}^{old}$ are replaced with $\mathfrak{B}^{new}$, $\mathbf{F}^{new}$ and the algorithm returns to the first step in section 3.1. The process is iterated until convergence as the evolving base set $\mathbf{B}$ registers itself to the target set $\mathbf{T_0}$.

3.3 Initialization and iteration

Hartley normalization is performed on the image coordinates of both point sets, thus pre-registering their centroids and setting the image coordinates to have unit variance. In this paper, SIFT [35] feature descriptors were also attached to the points. These descriptors are normalized to have magnitudes of σ_t of equation (7).

For initialization of the correspondence, we use SIFT flow [33] to give initial values of $\mathfrak{B}^{new}$. However, SIFT flow is not used to initialize the camera pose. As can be seen from equation (8), setting l_i to zero for the first EM iteration will cause the algorithm to ignore the epipolar constraint during this first iteration. Once $\mathfrak{B}^{new}$ is calculated, $\mathbf{F}^{new}$ can be calculated from $\mathfrak{B}^{new}$ and $\mathfrak{B}_\mathbf{0}$, after which $\mathfrak{B}^{old}$, $\mathbf{F}^{old}$ are replaced with $\mathfrak{B}^{new}$, $\mathbf{F}^{new}$. Normal EM resumes with l_i restored, and the process is iterated until convergence.

For stability, we set σ_t, σ_b to artificially large values, then steadily anneal them smaller. This corresponds to the increased accuracy expected of the camera pose estimate and the point correspondence. A summary of the algorithm is given in figure 3.

4 System implementation

In this section, we consider how one might build a complete SfM system using our proposed joint estimation framework. To do this, we must address issues such as point set acquisition, occlusion detection and initialization under real world conditions.

```
Input: Point sets, $\mathbf{B_0}$, $\mathbf{T_0}$
Initialize $\sigma_t, \sigma_b$;
Initialize $\mathfrak{B}^{old}$ as $\mathfrak{B}_\mathbf{0}$, $l_i$ to zero vector;
while $\sigma_t, \sigma_b$ above threshold do
    while No convergence do
        Use eqn (9) to evaluate $\phi_{ij}(b_i^{old}, t_{0j})$ from $\mathfrak{B}^{old}$, $\mathbf{F}^{old}$;
        Use eqn (13) to determine $\mathfrak{B}^{new}$ from $\phi_{ij}(b_i^{old}, t_{0j})$;
        Use bundle adjustment to obtain $\mathbf{F}^{new}$ from $\mathfrak{B}^{new}$ and $\mathfrak{B}_\mathbf{0}$;
        Replace $\mathfrak{B}^{old}, \mathbf{F}^{old}$ with $\mathfrak{B}^{new}, \mathbf{F}^{new}$;
    end
    Anneal $\sigma_t = \alpha\sigma_t$, $\sigma_b = \alpha\sigma_b$, where $\alpha = 0.97$.
end
```

Fig. 3 Algorithm

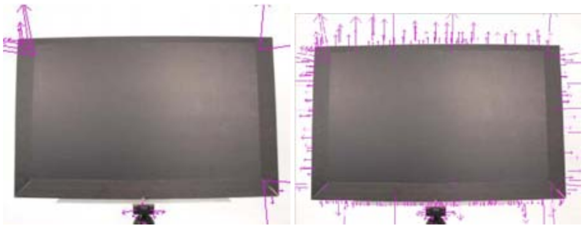


Fig. 4 Left to right: Output of SIFT feature detector with and without its cornerness function.

The first step of any such system has to be the identification of point sets in both images. As our algorithm is capable of utilizing non-unique features such as edges, we do not wish to use a corner detector, which would reject all edge-like features. Edge detectors would provide edge information; however, they often detect many spurious edges [54]. In order to overcome these problems, we detect features following the seminal SIFT algorithm [35]. However, as we are not interested in uniqueness, we disabled the cornerness term which otherwise would remove feature points that are considered too edge-like. The result appears to resemble that of a rather sparse but robust edge detector as illustrated in figure 4 but will also provide corner information when available. The descriptors that come with the SIFT detector also contribute greatly to stability.

The next issue is one of initialization and occlusion detection. What we need at this stage is not a well localized image registration but a crude initialization and a general idea of which sections of the image are occluded (feature points in the occluded regions need to be removed from the point sets $\mathbf{B_0}$ and $\mathbf{T_0}$). For these purposes, we utilize the dense SIFT flow algorithm to give us a crude mapping. Occluded regions are defined as regions where the SIFT flow is inconsistent, i.e. point A in image 1 maps to point B in image 2, however, point B does not map back to anywhere near point A. At very large baselines, the occlusion detector may declare the entire image as occlusion. In such situations the occlusion mask is discarded. (Note that a more sophisticated form of occlusion detection can be obtained in [4].) The complete pipeline for camera pose recovery is shown in figure 5.

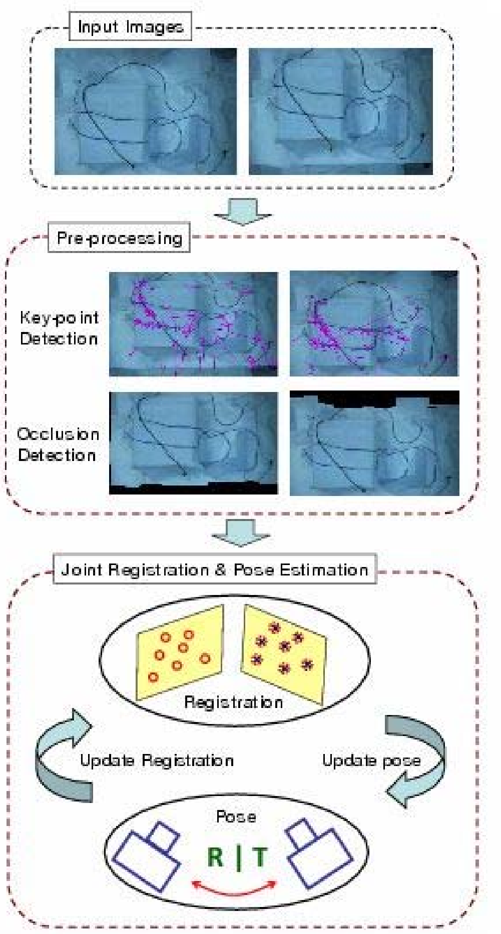


Fig. 5 Pose computation pipeline. Left to right: Input images, key-point detection, occlusion detection (with occluded pixels set to zero) and recovered camera pose.

5 Experiments and Evaluation

We run a series of real and simulated experiments to evaluate our algorithm, with errors reported as deviations from ground truth rotation and translation. All parameters reported are with respect to the Hartley normalized coordinates. All images are evaluated at a resolution of 640×480.

The rotational error $\widetilde{R}$ refers to the rotation angle in degree needed to align the reference frame of the computed pose to that of the true pose. The translational error $\widetilde{T}$ is the angle in degree between the computed translation and the ground truth translation. Although both the rotational and translational errors are given in degrees, in general, for typical camera and scene configuration, a large rotational error is more serious than a translational error of similar magnitude.

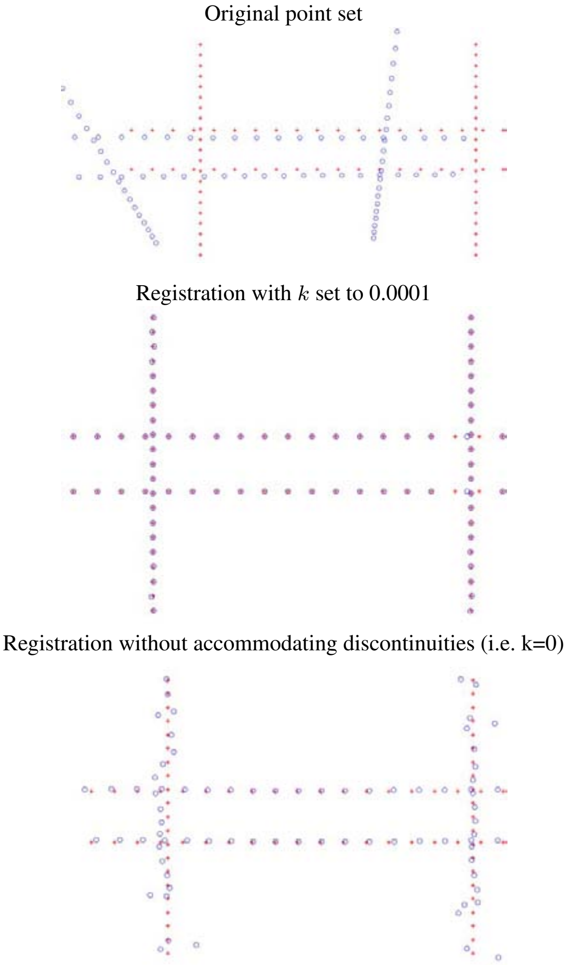
We test our system on a wide range of scene types and baselines. These include many "non-traditional" SfM scenes in which there are few/ no distinct corners available for matching, such as natural vegetation scenes where there is a large amount of self occlusion and thus spurious corners, architectural scenes where the available corners are very repetitive as well as more traditional SfM scenes. This is followed by a systematic evaluation of our algorithm's handling of increasing baseline. For most scenes, ground truth camera pose is obtained by manually obtaining point correspondences until the computed camera pose is stable. An exception is made for the last two images in figure 7, where the extremely textureless scenes were taken using linear rail with known motion. A calibrated camera was used for all these tests.

To give the reader a general feel for the scenes' difficulty, our results are benchmarked against that of a traditional SfM technique. Correspondences are obtained using [35]. Camera pose is obtained using the five point algorithm [43] together with outlier removal by the RANSAC implementation in [29], the outliers rejection threshold being set at a Sampson distance of 0.001. The RANSAC step is followed by a bundle adjustment using the implementation of [34] to minimize the reprojection error.

The same set of parameters are used throughout the entire experiments. The two Gaussian parameters σ_b and σ_t in equations (2) and (7) are given an initial value of $\sigma_t = \sigma_b = 0.1$. They are decreased using the annealing parameter $\alpha = 0.97$. The occlusion handling parameter μ in equation (9) is set to 0.5, while the epipolar outlier handling parameter ζ in equation (6) is set to 0.01. λ, which controls the relative weight given to the smoothness function, is set to 1. k, the degree of tolerance for high frequency components in equation (5), was set to 0.0001, while γ, the standard deviation of the Gaussian smoothness function, was set to 1. The algorithm can handle approximately 1500 SIFT features in 5 minutes.

5.1 Evaluation

We evaluate our algorithm on a variety of real and simulated scenes. In the simulated scene in figure 6, we illustrate our system's performance over depth discontinuities and the role of the discontinuity parameter k in equation (5). It shows that our algorithm can handle depth discontinuities and the pose computed is robust to the smoothness perturbations that the discontinuities induce. This is also illustrated in a number of real images of trees in figure 8 and a bicycle scene in figure 9. For the outdoor scenes, the baseline is usually a few meters. For the indoor scenes where objects are closer to the camera, the baseline is typically half a meter.


Original point set

Registration with k set to 0.0001

Registration without accommodating discontinuities (i.e. k=0)

Fig. 6 The vertical bars have a different depth and color (color not shown in results) from the horizontal bars. As the camera moves, the depth discontinuity causes the vertical bars to slide over the horizontal one. Setting the high frequency tolerance parameter k to 0.0001, the system retains both the smoothness constraint while accommodating the discontinuities. While there are some correspondence errors, our system is sufficiently robust to ensure that there is negligible error in the overall pose estimation. Using the standard motion coherence, where $k = 0$, the conflict between registration, smoothness and epipolar geometry cannot be resolved. The resultant pose estimate suffers, with a translational and rotation errors of 13.5^o and 5^o respectively.

In figure 7, we investigate real images of scenes with sparsely distributed corners. Errors in the recovered camera parameters are reported below the images. "Ours" indicates the errors obtained by our algorithm, "SIFT flow" those obtained by running the five point algorithm and bundle adjustment on SIFT flow as correspondence input and finally, "Traditional" those obtained by running the five point algorithm with RANSAC and bundle adjustment on SIFT matches as correspondence input (traditional here refers to the dependence on unique features such as corners). In some scenes, SIFT matching returns too few matches for the traditional algorithm to give a pose estimate. In such circumstances, the pose error is given as Not Applicable (NA).

The first two test images are of buildings. As in many man-made structures, lines and edges are the predominant cues present. The problem of identifying matches needed for traditional SfM is compounded by the wide baseline. By relaxing the uniqueness requirement, our algorithm can utilize a much greater amount of information compared to the traditional approach, leading to a stable camera pose recovery. The third and fourth scenes consist of extremely sparsely distributed sets of corners. Here the primary SfM cue is the edge information. Our algorithm can utilize this edge information to convert an information-impoverished scene with very few point matches into an information-rich scene. This allows it to circumvent the difficulties faced by the traditional SfM algorithms.

In figure 8, we further our investigation on scenes which contain a large number of non-unique corners. This is true for the floor image, where the grid pattern tiling results in multiple corners with nearly identical feature descriptor. It also occurs in natural vegetation scenes, where the leaves form many repetitive features. For plants, the problem is made more severe because the extensive self occlusion caused by the interlocking of leaves and branches further degrades potential corner descriptors. Hence, despite the large number of corners available (nearly 1000 for some of the images), there are few SIFT matches on the foliage. For the floor scene, jointly estimating the correspondence and pose allows the handling of non-unique features and the subsequent pose recovery. For the plant images, our algorithm can ignore the noise in the degraded feature descriptors and utilize the tree trunks and their outlines to obtain a camera pose estimate. We also illustrate a failure case in the last column of figure 8. With most of the feature descriptors badly perturbed by self occlusion, the primary SfM cue lies in the edge information which in this case is the extremal contour of the plant. Unlike polyhedral objects, the extremal contour of the plant is view-dependent (i.e. the points on the plants that participate in generating the contour are view-dependent). This dependency effect is especially significant when the displacement is quite large (at smaller displacements our algorithm can handle this scene).

Finally, in figure 9 we evaluate our algorithm on traditional SfM scenes with adequate number of unique features. This shows that our algorithm also works well when the primary cue lies in disconnected but discriminative corner information. Although some scenes contain significant depth discontinuities, our algorithm can produce the same accuracy in the camera pose estimate when compared to the results of the traditional SfM algorithms.

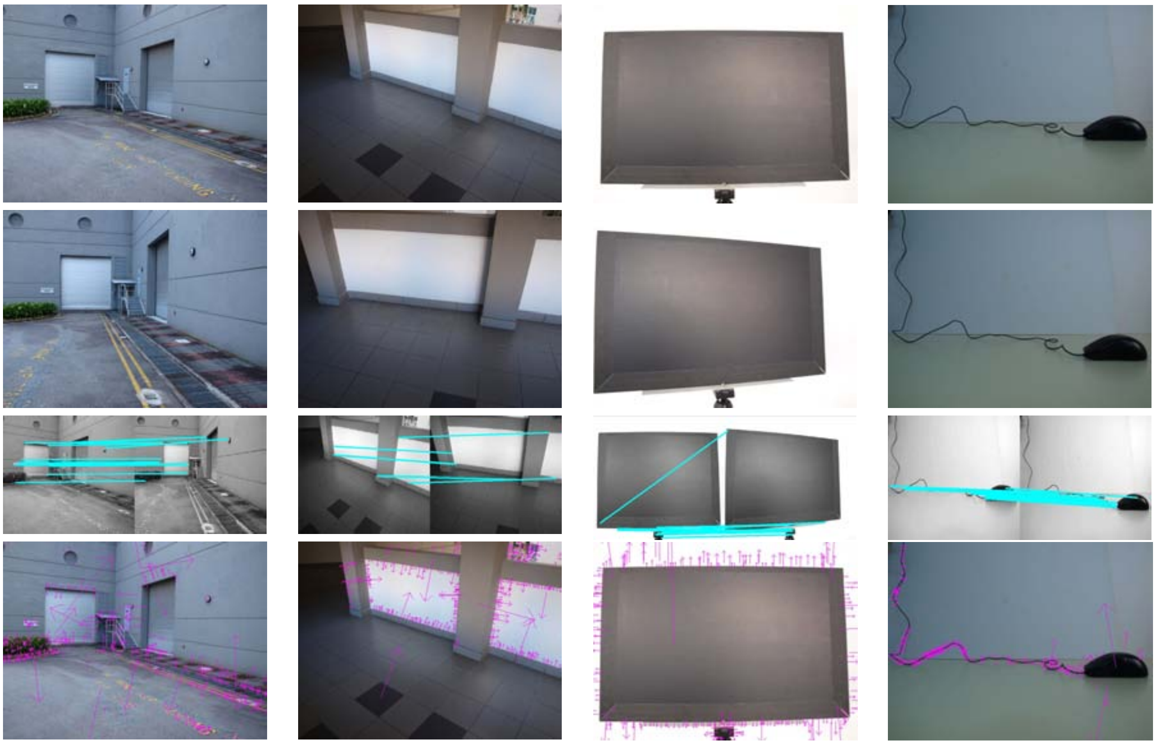
5.2 Performance with increasing baseline

In figure 10, we investigate our algorithm's behavior with increasing baseline. The sequences consist of a moving camera fixated upon a scene and are arranged in increasing baseline and thus level of difficulty. The color-coded depth maps obtained by reconstructing the scene using PMVS [16] are also included. The first sequence is a traditional, well textured SfM scene. The baseline is fairly large, with the camera rotating through 33.9 degrees while fixated on the table. Our algorithm gives a stable estimate of camera pose for all images in that sequence, achieving comparable performance with the traditional approach, and slightly outperforming it for the case of the widest baseline. The second sequence is of a moderately difficult scene where our algorithm outperforms the traditional approach by remaining stable over the entire sequence. This enhanced stability is the result of our algorithm being able to utilize the edge features provided by the door frame, while the traditional approach is limited to the tightly clustered features on the posters, giving it a small effective field of view. Finally, the last sequence shows a very difficult scene. There are very few feature matches (the point matches from the second image pair are shown in figure 1) and by the third image of the sequence, there are insufficient matches for a traditional SfM algorithm to make a pose estimate. Furthermore, the baseline is slightly larger than that shown in the previous two scenes, with a maximum camera rotation of 35.9 degrees about the object of interest. Although the performance of our algorithm at larger baselines degrades, an estimate of the camera pose and the depth can still be recovered at very large baselines.

5.3 Unresolved issues and Discussion

Throughout this paper, we have emphasized our algorithm's ability to utilize more information than traditional SfM algorithms. However, we should caution that unless properly weighted, more information is not necessarily better. This is illustrated in figure 11, where an undulating cloth surface means that the edge information is subject to a great deal of “occlusion” noise, caused by the extremal contours varying with viewpoint changes. inconsistent edge detection. Despite the large amount of occlusion, our algorithm could still return a fairly good estimate; however, re-running our algorithm using only corner information improves the results. This indicates that it is the inclusion of “noisy” information without proper weighting that degrades somewhat the performance of our algorithm. We note that unique corner matches can be better incorporated into our algorithm by allowing these point matches to influence the σ_t values in our Gaussian mixture. A principled fusion of these different sources of match information, together with a well thought-out data weighting scheme would be of great practical value and remains to be properly addressed.

While our algorithm cannot attain the global minimum and more research in that direction is necessary, we would



	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$
SIFT flow	15.9177	32.7802	4.8950	5.5900	13.589	81.2057	2.0970	27.8590
Traditional	14.5585	78.5412	12.8954	9.3199	15.0885	57.1229	17.7057	80.8453
Ours	0.8944	2.8339	4.2943	7.5999	1.6161	12.5875	0.9923	11.5929

Fig. 7 We show a number of scenes where there are few corners and correspondingly few matches. The correspondences obtained from SIFT matching [35] are shown in the third row. The matches that exist are also poorly distributed, with the majority of matches being clustered in a small region. The fourth row shows the SIFT points used by our algorithm. By relaxing the need for unique correspondence, we can use a much richer and better distributed point set, which in turn permits a better recovery of the camera pose. The pose errors are reported below the images (see the text for the meanings of $\widetilde{R}$ and $\widetilde{T}$).

like to make some final remarks on the stability of our algorithm against local minima, whether arising from the inherent ambiguity of the SfM problem, or caused by errors in the initialization. Referring to figures 7, 8, and 10, it can be seen that both "SIFT flow" and "Traditional" sometimes returned a translation estimate that was almost 90 degrees off the correct solution. This is caused by the well known bias of the translation estimate towards the center of the image (the true translation is lateral in these sequences), which becomes more acute when the feature matches are insufficient or of poor quality. Our algorithm suffers less from these well known local minima of SfM because we can use ambiguous edge features in these circumstances. While initialization with SIFT flow helps reduce the local minima problem, it can be seen from our results that we can converge to a correct solution even when the original SIFT flow initialization is fairly erroneous. This is especially obvious in the sequences with varying baseline in figure 10, where our algorithm degrades gracefully with increasing displacement induced noise and worsening SIFT flow initialization.

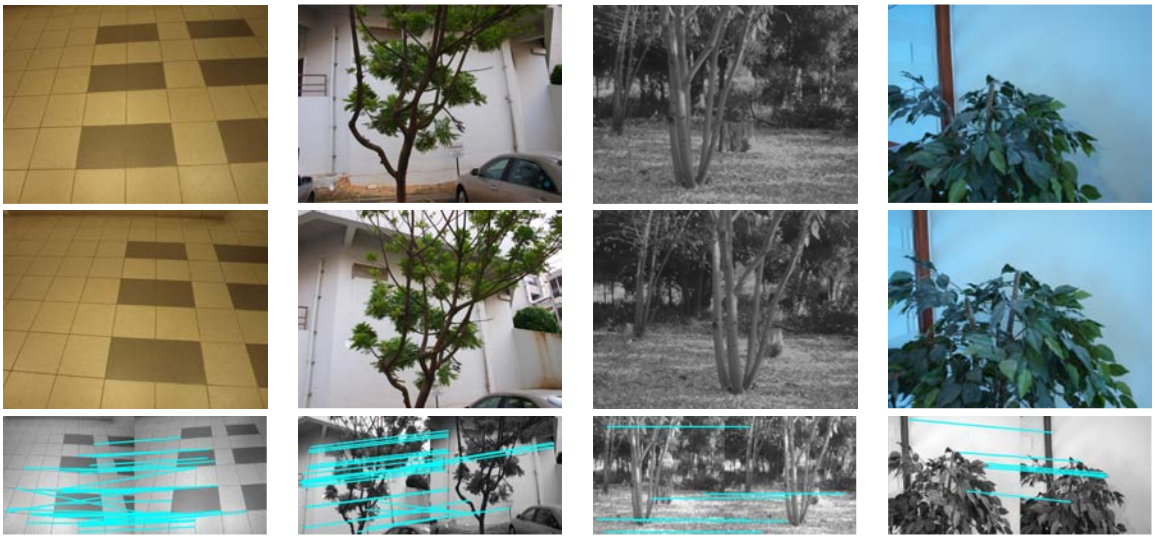
6 Conclusion

In this paper we have extended the point registration framework to handle the two-frame structure from motion problem. Integrating the motion coherence constraint into the joint camera pose and matching algorithm provides a principled means of incorporating feature points with non-unique descriptors. This in turn allows us to recover camera pose from previously difficult SfM scenes where edges are the dominant cues and point features are unreliable.

While the results obtained so far are promising, there is also much scope for further improvements in terms of improving the initialization, incorporation of multiple views, proper weighting of cues, as well as basic improvement to the point registration mechanism.

7 Acknowledgments

We are deeply appreciative of the discussions with Hiew Litt Teen and Bing Jain.



	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$
SIFT flow	2.0097	27.8590	7.6931	3.6912	1.3447	11.5193	15.3635	88.9147
Traditional	9.3170	56.4661	6.5460	86.6666	66.5418	89.6218	NA	NA
Ours	1.2401	8.6582	1.5239	13.5200	0.8103	9.3550	48.6037	35.5356

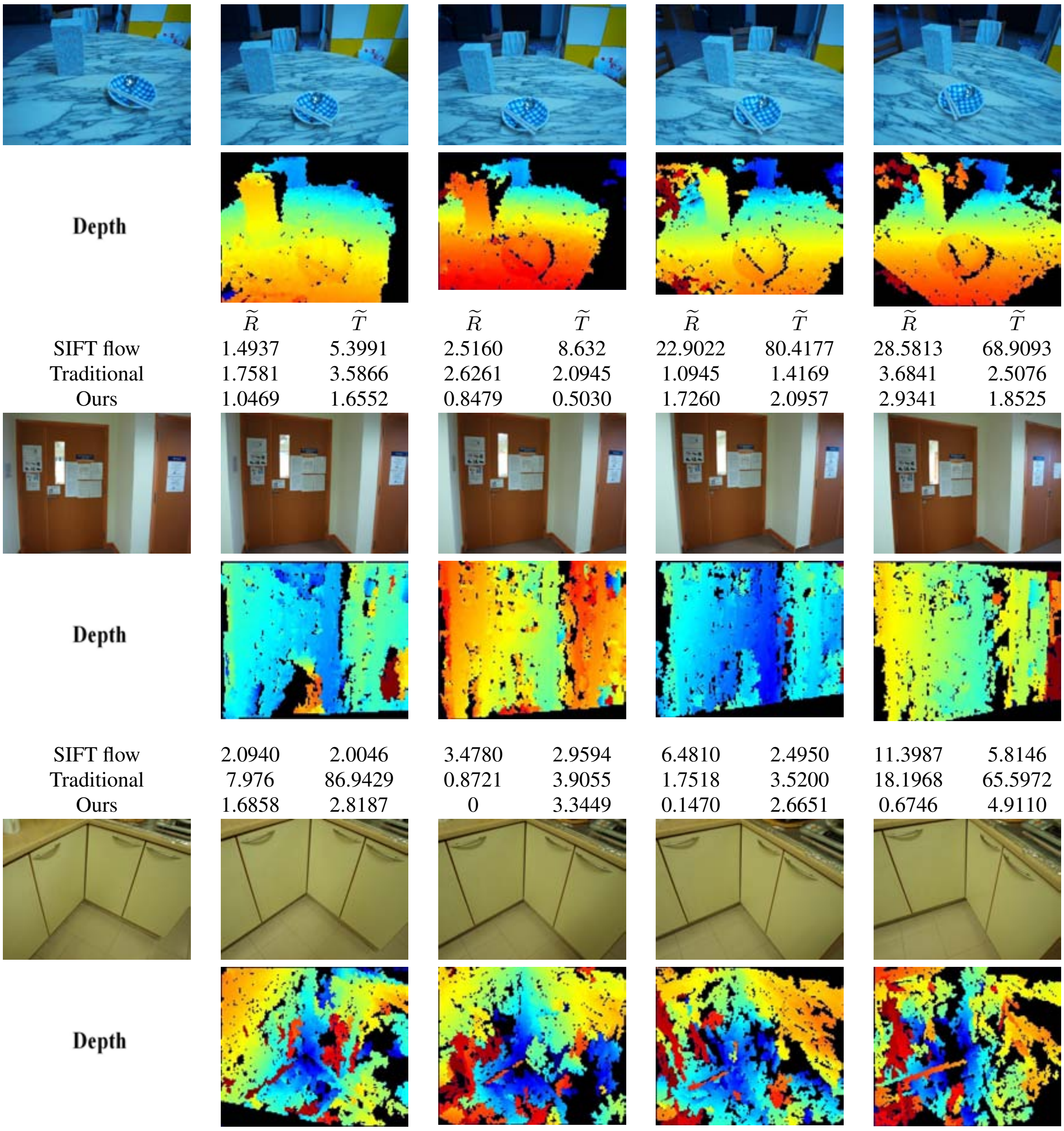
Fig. 8 Here we experiment on images where corners are plentiful (some of the tree images have over 1000 features detected) but unique matching remains challenging. This lack of uniqueness is due to the strong repetitive pattern. For the plant images, the problem is compounded by the interlocking leaves which induce self-occlusion and corresponding feature degradation. For the floor image, our algorithm can utilize the non-unique SIFT feature to recover camera pose, while for the tree images, we can utilize the features lying along the trees branches. The final image shows a failure case where the stem is hidden by the foliage and the problem is further compounded by a view-dependent extremal contour.

	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$
SIFT flow	1.13	0.63	0.38	1.53	5.92	14.7	24.51	85.50	1.83	3.89	14.12	27.08
Traditional	0.93	1.51	0.17	0.60	3.20	3.95	0.79	2.26	0.80	0.83	0.96	1.01
Ours	0.30	0.38	0.35	0.60	0.12	0.46	0.13	0.24	0.88	0.85	0.50	0.13

Fig. 9 Evaluating our algorithm on a traditional structure from motion sequence from Christoph Strecha. Camera pose is computed between adjacent image pairs. Observe that our algorithm also performs well on well-textured structure from motion scenes.

References

1. A. Bartoli and P. Sturm. The 3d line motion matrix and alignment of line reconstructions. *In Proc. of Computer Vision and Pattern Recognition*, 2001.
2. A. Bartoli and P. Sturm. Structure-from-motion using lines: Representation, triangulation, and bundle adjustment. *Computer Vision and Image Understanding*, 100:416–441, 2005.
3. H. Bay, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. *In Proc. of European Conference on Computer Vision*, 2006.
4. V.G Bellile, A. Bartoli and P.Sayd. Deformable Surface Augmentation in spite of Self-Occlusions. *In Proc. of International Symposium on Mixed and Augmented Reality*, 2007.
5. P. Besl and N. MacKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.
6. S.Z.Chang. Epipolar parameterization for reconstructing 3d rigid curve. *Pattern Recognition*, 30:1817–1827(11), 1997.
7. H. Chui and A. Rangarajan. A new algorithm for non-rigid point matching. *In Proc. of Computer Vision and Pattern Recognition*, 2000.
8. P. David, D. Dementhon, R. Duraiswami, and H. Samet. Simultaneous pose and correspondence determination using line features. *International Journal of Computer Vision*, 2:424–431, 2002.
9. F. Dellaert, S. Seitz, C. Thorpe, and S. Thurn. Structure from motion without correspondence. *In Proc. of Computer Vision and Pattern Recognition*, 2000.
10. C. Engels, H. Stewenius, and D. Nister. Bundle adjustment rules. In *Photogrammetric Computer Vision*, 2006.
11. O. Enqvist, and F. Kahl. Robust Optimal Pose Estimation. *European Conference on Computer Vision*, 2008.
12. O. Enqvist, and F. Kahl. Two view geometry estimation with outliers. *British Conference on Machine Vision*, 2009.
13. O. Enqvist, K. Josephson and F. Khal. Optimal Correspondance from Pairwise Constraints. *International Conference on Computer Vision*, 2009.
14. O. Faugeras and B. Mourrain. On the geometry and algebra of the point and line correspondences between n images. *In Proc. of International Conference on Computer Vision*, 1995.
15. M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. of the ACM*, 24:381–395, 1981.
16. Y. Furukawa and J. Ponce. Accurate, dense, and robust multi-view stereopsis. *In Proc. of Conference on Computer Vision and Pattern Recognition*, 2007.

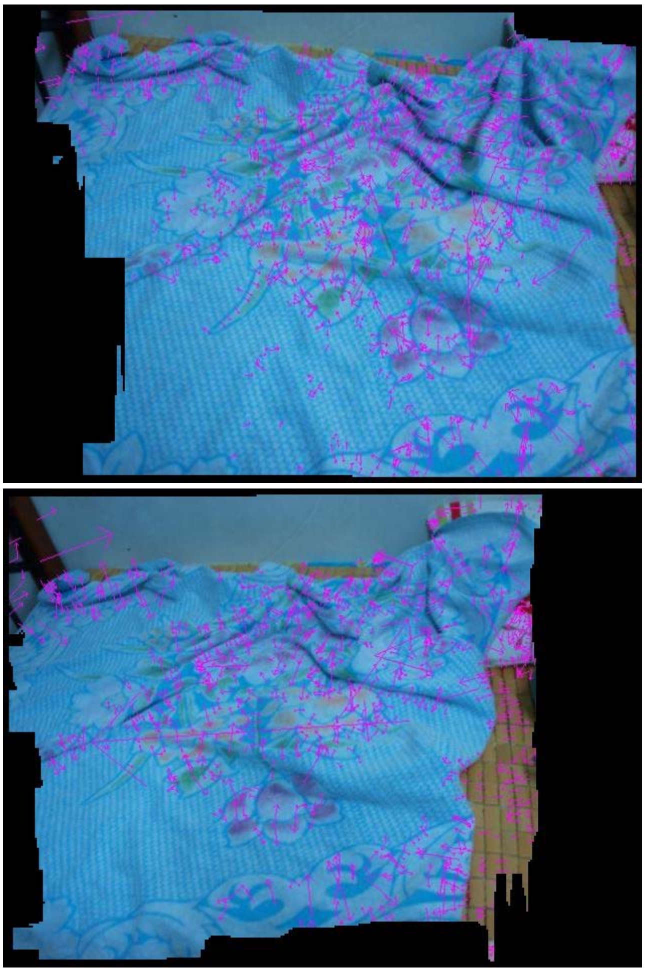


	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$	$\widetilde{R}$	$\widetilde{T}$
SIFT flow	1.4937	5.3991	2.5160	8.632	22.9022	80.4177	28.5813	68.9093
Traditional	1.7581	3.5866	2.6261	2.0945	1.0945	1.4169	3.6841	2.5076
Ours	1.0469	1.6552	0.8479	0.5030	1.7260	2.0957	2.9341	1.8525

SIFT flow	2.0940	2.0046	3.4780	2.9594	6.4810	2.4950	11.3987	5.8146
Traditional	7.976	86.9429	0.8721	3.9055	1.7518	3.5200	18.1968	65.5972
Ours	1.6858	2.8187	0	3.3449	0.1470	2.6651	0.6746	4.9110

SIFT flow	3.3724	9.0112	15.3820	48.9364	22.4894	49.2501	53.7316	57.2713
Traditional	8.7717	10.4135	18.2654	83.1424	NA	NA	84.6166	53.1542
Ours	2.6484	4.8031	4.0713	2.2964	7.1120	2.6457	12.1119	12.5768

Fig. 10 Sequences in increasing order of difficulty. Camera pose is with respect to the base image on the extreme left, with each scenes color coded depth map below it. Warm colors representing near depths and cold colors far depths. Observe that in scenes with sparse sets of corners, our algorithm has greater stability over large baselines, compared to the traditional approach. In the second image sequence, our algorithm also exhibits an advantage over the traditional SfM in handling wider baselines. This is because our algorithm allows the utilization of the entire door contour, rather than focusing on the tightly clustered feature points available on the poster. In the third image sequence where point to point feature matching is extremely difficult (in the fourth image of this sequence, there are insufficient matches to make an estimate of the camera pose using traditional methods), our algorithm still remains stable. Although the baseline is wider than the previous two scenes, our algorithm deteriorates gracefully.



	$\widetilde{R}$	$\widetilde{T}$
SIFT flow	5.8690	6.1868
Traditional	1.3211	1.1139
Ours	1.4134	3.7060
Ours*	1.3083	2.5192

Fig. 11 Computed point sets on two images of a textured cloth. This image is easy for traditional SfM. However, our point set recovery faces large amount of "self-occlusion" caused by the extremal contours on the blanket varying under viewpoint changes. Under Ours*, we applied our algorithm using only traditional SIFT corner features. The results improve significantly, showing that when there is abundant high quality corner information present, including more noisy edge information can have a negative impact on performance. This scene also illustrates our algorithm ability to give a reasonable estimate despite large amount of noise and occlusion.

17. P. Georgel, S. Benhimane , and N. Navab. A Unified Approach Combining Photometric and Geometric Information for Pose Estimation. In *British Machine Vision Conference*, 2008.
18. P. Georgel, A. Bartoli, and N. Navab. Simultaneous In-Plane Motion Estimation and Point Matching Using Geometrical Cues Only In *Workshop on Motion and Video Computing*, pages 1–7, 2009.
19. F. Girosi, M. Jones, and T. Poggio. Regularization theory and neural networks architectures. *Neural Computation*, vol. 7, no. 2, pp. 219–269, 1995.
20. L. Goshen and I. Shimshoni. Balanced exploration and exploitation model search for efficient epipolar geometry estimation. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 30(3):1230–1242, 2008.
21. C. Harris and M. Stephens. A combined corner and edge detector. *In Proc. Fourth Alvery Vision Conference*, 1988.
22. R. Hartley. In Defense of the Eight-Point Algorithm. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 19(6):580–593, 1997.
23. R. Hartley and A. Zisserman. Multiple view geometry in computer vision. *Cambridge University Press*, 2000.
24. S. Hou, Ramani, and Karthik. Structure-oriented contour representation and matching for engineering shapes. *Comput. Aided Des.*, 40(1):94–108, 2008.
25. H. Jiang and S.X Yu. Linear solution to scale and rotation invariant object matching. In *Proc. Computer Vision and Pattern Recognition*,2474-2481,2009.
26. F. Kahl, S. Agarwal, M.K. Chandraker, D. Kriegman, S. Belongie. Practical Global Optimization for Multiview Geometry. *International Journal of Computer Vision*, 79: 271–284, 2008.
27. G. Klein and T. Drummond. Robust visual tracking for non-instrumented augmented reality. In *International Symposium on Mixed and Augmented Reality*, page 113, 2003.
28. G. Klein and D. Murray. Parallel tracking and mapping for small AR workspaces. In *International Symposium on Mixed and Augmented Reality*, pages 1–10, 2007.
29. P. D. Kovesi. MATLAB and Octave functions for computer vision and image processing. School of Computer Science & Software Engineering, The University of Western Australia. Available from: <http://www.csse.uwa.edu.au/∼pk/research/matlabfns/>.
30. S. Lehman, A. P. Bradley, I. V. L. Clarkson, J. Williams, and P. J. Kootsookos. Correspondence-free determination of the affine fundamental matrix. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1):82–96, 2007.
31. H. Li and R. Hartley. The 3d-3d registration problem revisited. In *International Conference on Computer Vision*, pages 1–8, 2007.
32. W.Y. Lin, G. Dong, P. Tan, L.F. Cheong, C.H. Yan. Simultaneous Camera Pose and Correspondence Estimation in Cornerless Images. *In Proc. International Conference on Computer Vision*, 2009.
33. C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. T. Freeman. Sift flow: Dense correspondence across different scenes. *In Proc. of European Conference on Computer Vision*, 2008.
34. M. I. A. Lourakis and A. A. Argyros. SBA: A Software Package for Generic Sparse Bundle Adjustment. *ACM Trans. Math. Software*, 36(1):1–30, 2009. Available from: < http://www.ics.forth.gr/ lourakis/sba>.
35. D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Intl. J. of Computer Vision*, 60(2):91–110, 2004.
36. A. Makadia, C. Geyer, and K. Daniilidis. Correspondence-free structure from motion. *International Journal of Computer Vision*, 75(3):311 – 327, 2007.
37. L. Masson, F. Jurie, and M. Dhome. Contour/texture approach for visual tracking. In *Scandinavian conference on Image analysis*, pages 661–668, 2003.
38. J. Meltzer and S. Soatto. Edge descriptors for robust wide-baseline correspondence. *In Proc. of Conference on Computer Vision and Pattern Recognition*, 2008.
39. L. Moisan and B. Stival. A probabilistic criterion to detect rigid point matches between two images and estimate the fundamental matrix. In *International Journal of Computer Vision*, 57(3):201–218, 2004.
40. M.Moreno-Noguer, V.Lepetit, and P.Fua. Pose priors for simultaneously solving alignment and correspondence. *In Proc. of European Conference on Computer Vision*, 2008.
41. E. Mouragnon, F. Dekeyser, P. Sayd, M. Lhuillier, and M. Dhome. Real time localization and 3d reconstruction. In *Proc. IEEE Intl. Conference on Computer Vision and Pattern Recognition*,pages 363–370, 2006.
42. A. Myronenko, X. Song, and M. Carreira-Perpinan. Non-rigid point set registration: Coherent point drift. *Advances in Neural Information Processing Systems (NIPS)*, 2007.

43. D. Nister. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):756-770, 2004.
44. T. Papadopoulo and O. Faugeras. Computing structure and motion of general 3d rigid curves from monocular sequences of perspective images. *In Proc. of European Conference on Computer Vision*, 1996.
45. M. Pressigout and E. Marchand. A model free hybrid algorithm for real time tracking. In *International Conference on Image Processing*, 2005.
46. A. Rangarajan, H. Chui, and F. Bookstein. The softassign Procrustes matching algorithm. In *International Conference on Information Processing in Medical Imaging*, pages 29–42, 1997.
47. O. Ricardo, C. Joao, and X. Joao. Contour point tracking by enforcement of rigidity constraints. *3DIM : International Conference on 3-D Digital Imaging and Modeling*, 2005.
48. C. Schellewald and C. Schnrr Probabilistic Subgraph Matching Based on Convex Relaxation. *Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2005.
49. Y. Sheikh, A. Hakeem, and M. Shah. On the direct estimation of the fundamental matrix. *In Proc. of Computer Vision and Image Processing*, 2007.
50. M. E. Spetsakis and J. Aloimonos. Motion and structure from point and line matches. *In Proc. of International Conference on Computer Vision*, 1987.
51. R. Szeliski and R. Weiss. Robust shape recovery from occluding contours using a linear smoother. *International Journal of Computer Vision*, 28(1):141–165, 1993.
52. L. Torresani, S. Kolmogorov and C. Rother. Feature Correspondence via Graph Matching: Models and Global Optimization. *European Conference of Computer Vision*, 2008.
53. B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment - a modern synthesis. *Vision Algorithms: Theory and Practise*, 1999.
54. L. Vacchetti, V. Lepetit, and P. Fua. Combining edge and texture information for real-time accurate 3d camera tracking. In *International Symposium on Mixed and Augmented Reality*, pages 48–57, 2004.
55. L. Valgaerts, A. Bruhn, and J. Weickert. A variational model for the joint recovery of the fundamental matrix and the optical flow. *In Proc. of Pattern Recognition*, 2008.
56. K.-Y. K. Wong and R. Cipolla. Structure and motion estimation from apparent contours under circular motion. *Image and Vision Computing*, (5–6):441–448, 2001.
57. K.-Y. K. Wong and R. Cipolla. Structure and motion from silhouettes. *In Proc. International Conference on Computer Vision*, 2001.
58. A. L. Yuille and N. M. Grzywacz. A mathematical analysis of the motion coherence theory. *International Journal of Computer Vision*, 3:155–175(2), 1988.
59. Z. Zhang. Iterative point matching for registration of free-form curves. In *International Journal of Computer Vision*, volume 13, number 2, pages 119–152, 2004.

8 Appendix

This appendix deals with how the smoothness function $\Psi(\mathfrak{B})$ can be simplified into a more tractable form for the minimization process. In particular, we want to show that at the minima of $A(\mathbf{B},\mathbf{F})$, $\Psi(\mathfrak{B})$ is related to $\mathfrak{B}$ and $\mathfrak{B}_\mathbf{0}$ by $\Psi(\mathfrak{B}) = tr(\Gamma \mathbf{G}^{-1}\Gamma^T)$.

At the minima, the derivative of equation (8) with respect to the velocity field expressed in the Fourier domain $v'(.)$ must be zero. Hence, utilizing the Fourier transform relation, $v(\beta_{0i}) = \int_{\Re^2} v'(s)e^{2\pi k<\beta_{0i},s>}ds$, we obtain the constraint

$$\begin{aligned}
&\frac{\partial A(v',\mathbf{F})}{\partial v'(z)} = \\
&-\sum_{j=1}^{N}\frac{\sum_{i=1}^{M}\left(\frac{1}{\sigma_t^2}(\beta_i-\hat{t}_{0j})\right)g(t_{0j}-b_i,\sigma_t)\int_{\Re^2}\frac{\partial v'(s)}{\partial v'(z)}e^{2\pi k<\beta_{0i},s>}ds}{\sum_{i=1}^{M}g(t_{0j}-b_i,\sigma_t)} \\
&+\sum_{i\in inlier}\frac{1}{\sigma_b^2}l_i l_i^T(\beta_i-r_i)\int_{\Re^2}\frac{\partial v'(s)}{\partial v'(z)}e^{2\pi k<\beta_{0i},s>}ds \\
&+\lambda\int_{\Re^2}\frac{\partial}{\partial v'(z)}\frac{|v'(s)|^2}{g'(s)+\kappa'(s)}ds \\
&=-\sum_{j=1}^{N}\frac{\sum_{i=1}^{M}\left(\frac{1}{\sigma_t^2}(\beta_i-\hat{t}_{0j})\right)g(t_{0j}-b_i,\sigma_t)e^{2\pi k<\beta_{0i},z>}}{\sum_{i=1}^{M}g(t_{0j}-b_i,\sigma_t)} \\
&+\sum_{i\in inlier}\frac{1}{\sigma_b^2}l_i l_i^T(\beta_i-r_i)e^{2\pi k<\beta_{0i},z>}+2\lambda\frac{v'(-z)}{g'(z)+\kappa'(z)}, \\
&=0_{2\times 1}
\end{aligned} \tag{15}$$

where $\hat{t}_{0j}$ denotes a two dimensional vector made of the first two elements of t_{0j}.

Simplifying equation (15), we obtain

$$-2\lambda\sum_{i=1}^{M}w_i e^{2\pi k<\beta_{0i},z>}+2\lambda\frac{v'(-z)}{g'(z)+\kappa'(z)}=0$$

where the two dimensional vectors w_i act as placeholders for the more complicated terms in (15).

Substituting z with $-z$ into the preceding equation and making some minor rearrangements, we have

$$v'(z) = (g'(-z)+\kappa'(-z))\sum_{i=1}^{M}w_i e^{-2\pi k<\beta_{0i},z>}. \tag{16}$$

where the two dimensional vectors, w_i, can be considered as weights which parameterize the velocity field.

Using the inverse Fourier transform relation

$$\begin{aligned}
&\int_{\Re^2} w_i^T w_j(g'(z)+\kappa'(z))e^{+2\pi k<\beta_{0j}-\beta_{0i},z>}dz \\
&\quad = w_i^T w_j\left(g(\beta_{0j}-\beta_{0i},\gamma)+\kappa(\beta_{0j}-\beta_{0i})\right),
\end{aligned}$$

and equation (16), we can rewrite the regularization term of equation (8) as

$$\begin{aligned}\varPsi(\mathfrak{B}) &= \int_{\Re^2} \frac{(v'(z))^T(v'(z))^*}{g'(z)+\kappa'(s)}dz \\ &= \int_{\Re^2} \frac{(g'(z)+\kappa'(s))^2 \sum_{i=1}^{M}\sum_{j=1}^{M} w_i^T w_j e^{+2\pi k<\beta_{0j}-\beta_{0i},z>}}{g'(z)+\kappa'(s)}dz \\ &= \sum_{i=1}^{M}\sum_{j=1}^{M}\int_{\Re^2} w_i^T w_j (g'(z)+\kappa'(s)) e^{+2\pi k<\beta_{0j}-\beta_{0i},z>}dz \\ &= tr(\mathbf{W^T G W}),\end{aligned} \tag{17}$$

where * represents the complex conjugate operation, $tr(.)$ represents the trace of a matrix, and

$$\begin{aligned}&\mathbf{W}_{M\times 2} = [w_1, ..., w_M]^T, \\ &\mathbf{G}(i,j) = g(\beta_{0i}-\beta_{0j},\gamma)+\kappa(\beta_{0i}-\beta_{0j}).\end{aligned}$$

If, as in the main text, one takes $\kappa(.)$ to be a function with spatial support less than the smallest separation between two feature points in $\mathbf{B_0}$, the above expression for $\mathbf{G}(i,j)$ can be simplified into

$$\mathbf{G}(i,j) = \begin{cases} g(\beta_{0i}-\beta_{0j},\gamma)+k, & i=j \\ g(\beta_{0i}-\beta_{0j},\gamma), & i\neq j \end{cases} \tag{18}$$

where k is some pre-determined constant.

Lastly, taking the inverse Fourier transform of equation (16), we obtain

$$\begin{aligned}v(z) &= (g(z,\gamma)+\kappa(z)) * \sum_{i=1}^{M} w_i \delta(z-\beta_{0i}) \\ &= \sum_{i=1}^{M} w_i (g(z-\beta_{0i},\gamma)+\kappa(z-\beta_{0i})).\end{aligned}$$

where δ is the Dirac delta. Hence,

$$\mathfrak{B}-\mathfrak{B}_\mathbf{0} = \mathbf{GW}. \tag{19}$$

Substituting equation (19) into (17), we see that the regularization term $\varPsi(\mathfrak{B})$, has the simplified form used in the main text

$$\varPsi(\mathfrak{B}) = tr(\mathbf{W^T G W}) = tr((\mathfrak{B}-\mathfrak{B}_\mathbf{0})^T\mathbf{G}^{-1}(\mathfrak{B}-\mathfrak{B}_\mathbf{0})). \tag{20}$$