Non-Rigid Structure from Locally Rigid Motion

by

Jonathan Taylor

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Graduate Department of Computer Science
University of Toronto

Abstract

Non-Rigid Structure from Locally Rigid Motion

Jonathan Taylor
Doctor of Philosophy
Graduate Department of Computer Science
University of Toronto
2014

The non-rigid structure from motion problem typically involves recovering the 3D trajectories of a set of scene points, from their corresponding image trajectories. In this thesis, the assumption of locally-rigid motion is used to regularize this otherwise under-constrained problem. The key idea is that even when a scene undergoes complex global deformations, the trajectories of local triplets of scene points can often be approximated by the vertices of a rigidly moving triangle. This intuition informs our bottom-up reconstruction procedure, which discovers such triplets through a hypothesis and test framework. To this end, a rigid triangle model is fit to the proposed image trajectories and evaluated using a procedure that we call 3-SFM. The recovered triangle models are then integrated into a global solution, by resolving their orthographic depth flip and translation ambiguities. Lastly, we consider using this solution to initialize an energy based model, subject to a set of soft isometric constraints, in order to allow each observation to constrain the global scene structure. Results on several sequences, both our own and from related work, suggest that these models are applicable in diverse and challenging scenes, such as those including multiple deforming bodies.

Contents

Chapter 1

Introduction

A fundamental problem in Computer Vision is to reconstruct the three dimensional structure of a dynamic scene from a stream of two dimensional image data, such as that depicted in Figure 1.1. Although the human visual system can use stereo streams to triangulate depth, it is clear that humans can deal perfectly well with such mono streams. Furthermore, it has been observed [30] that humans can often perceive the three dimensional structure of a scene from just the two dimensional projections of a sparse set of 3D points moving in that scene. Therefore, instead of inferring scene structure directly from the dense arrays of pixel values in an image stream, the standard approach is to work with a sparse set of image trajectories representing the noisy projections of a deforming set of 3D scene points. These trajectories can be obtained in a variety of ways, such as tracking local image features [29] or by sampling from a dense 2D deformation field [50] as illustrated in Figure 1.2. Regardless of how these trajectories were obtained, they are generally considered to be input to these algorithms. Thus, the goal is to formulate and fit a model of both the scene structure and projection properties that can explain these trajectories, while elucidating the true properties of the scene.

This paradigm, when formalized as a problem to be solved computationally, is called

Figure 1.1: Three frames of the the `PAPER` sequence from [44].

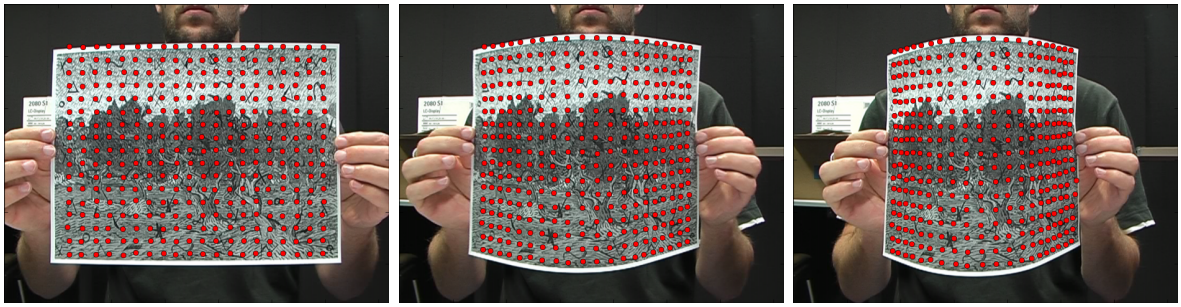

Figure 1.2: Three frames of the PAPER sequence with image trajectories from [50] overlaid.

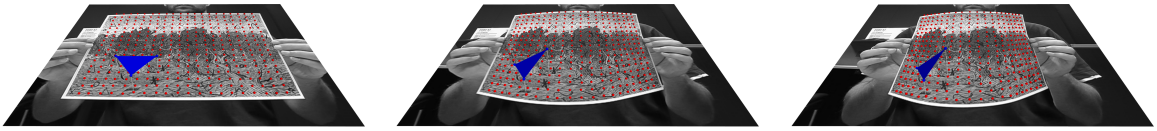

Figure 1.3: Three frames of the PAPER sequence with a local triplet of trajectories modelled as the projection of the vertices of a rigidly moving triangle.

the structure from motion (SFM) problem. When the scene is assumed to be static (or rigid), a rich theory detailing the required minimal configurations of points and views for scene reconstruction [7, 51] is available, as well as efficient reconstruction algorithms [51]. When the scene, however, is allowed to deform arbitrarily, the problem becomes highly ambiguous and new constraints need to be added to regularize the problem. Indeed, research in this area is generally a two-step process that begins with formulating a set of assumptions capable of yielding enough constraints to disambiguate the solution. Only then can a method be developed to leverage these constraints in reconstructing the scene from its projected two dimensional motion. These methods can then be said to be solutions to the non-rigid structure from motion (NRSFM) problem.

This thesis explores such problems by appealing to the assumption of *local rigidity*, which conjects that scenes undergoing complex global deformations can often be approximated locally by rigid motion models. For example, the paper being bent is undergoing an extreme global transformation. Nonetheless, at any point on the paper, the motion of nearby (local) points can be approximated, by a plane being rotated around the vertical axis. This thesis exploits this assumption to formulate and fit three different models to non-rigid scenes:

- A local model of the motion of three trajectories as that of the orthographically projected vertices of a rigidly moving triangle. This is illustrated in Figure 1.3.
- A piecewise local model of the motion of a set of trajectories as that of the averaged projections of a loosely connected set of these local rigid triangle models. This is

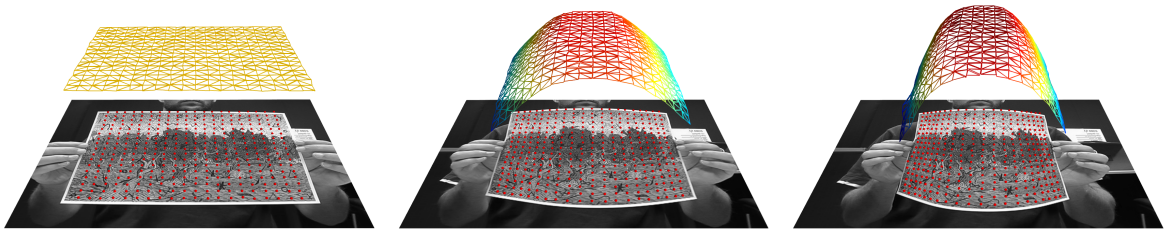

Figure 1.4: Three frames of the PAPER sequence modelled globally as a set of rigid triangles encouraged to align with their neighbours.

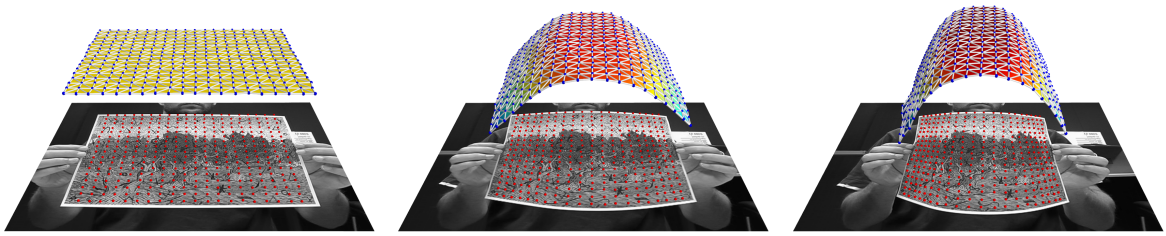

Figure 1.5: Three frames of the PAPER sequence modelled globally as a 3D point cloud (the blue points) encouraged to satisfy a set of pairwise isometric constraints (the white lines).

illustrated in Figure 1.4.

- A global model of the motion of a set of trajectories as the projection of a deforming 3D point set, weakly constrained to satisfy a set of pairwise isometric constraints. This is illustrated in Figure 1.5.

The global point cloud model is perhaps the most obvious way to formulate the local rigidity assumption, however, its natural formulation as an energy minimization problem leads to an energy landscape containing a plethora of local minima, making optimization extremely difficult. We can, however, use the piecewise global model to provide an initial guess to a procedure searching for a good local minima. We find, in turn, that for the piecewise local model, another hard optimization problem can be avoided by performing a bottom up reconstruction, in which the local three point rigid models are fit individually, a discrete labelling problem is solved to resolve orthographic depth flip ambiguities and a linear system is solved to tie these models together. Thus, a major contribution of this thesis is the decomposition of the global model fitting process into a series of tractable steps.

Although not used in recent decades, the general paradigm we follow is very old. Indeed, in his original work on the problem, Ullman [57, 58] suggests grouping points

in quadruplets, testing for rigidity, solving 4-point rigid structure from motion, and combining the results. Our piecewise model can therefore be thought of as a modern re-interpretation of Ullmans original scheme, applied to general non-rigid scenes and made even more local with three points instead of four.

This piecewise model, first proposed in [49], became part of a resurgence in interest in modelling complex scenes with simpler local models such as those modelling rigid planar [13] or quadratic deformation [18]. In contrast to our use of a minimal model though, these other approaches generally opt to combine local models using a substantially higher number of points. One could argue that this will cause our triangle models to be more susceptible to noise and their integration into a global structure more difficult due to the lack of significant inter-model overlap. We have found, however, that this is outweighed by the flexibility and ease of optimization that these minimal models provide. Further, by upgrading to a global model, some of these deficiencies can be ameliorated.

Further, the assumption that our global model can be regularized by a set of pairwise isometric constraints lies in stark contrast to the majority of methods that make sweeping assumptions about the global spatio-temporal properties of a scene. These include (1) deformations that span a low-dimensional shape space [52, 11, 54, 37, 16]; (2) trajectories that span a low-dimensional motion space [3]; (3) textured meshes with a regularized shape and low-order deformation [46, 47, 45, 59], or a known template shape [39]; and (4) scenes composed of rigid bodies moving independently [14, 56, 61] or in articulated configurations [63, 64]. Naturally, when these assumptions hold a lot of leverage can be gained from the constraints that they provide, but we argue that the local rigidity assumption is applicable in a diverse set of alternative scenarios in which these others are not.

In contrast to the global methods that do incorporate an isometric regularization, our work avoids many of the problems that come with assuming a known template [6] or a surface model [59]. The recent work of [60], however, is an exception as their formulation closely resembles our global point cloud model, and thus provides another route to its optimization. Their approach, however, requires ad-hoc assumptions to be made in order to identify the pairwise isometric constraints and heuristics to be used in order to perform a difficult global optimization. Our bottom up reconstruction bypasses these difficulties, although naturally admits its own. That work should therefore be seen as complementary to ours, and further serves to validate the difficulty in optimizing such a model.

The remainder of this document is organized as follows:

- Chapter 2 introduces the mathematics of the orthographic Non-Rigid Structure from Motion problem and the various solutions proposed in the literature. This

includes predominant work in the area involving generalizations of the classical rigid factorization approach [51] to the non-rigid case [11]. As well, we describe recent work with close ties to our approach due to the use of either piecewise models or local isometric regularizers [60].

- Chapter 3 presents 3-SFM as a method for fitting a model of three point rigid motion under orthography. This includes a direct method for recovering this model up to its fundamental ambiguities in the noiseless case. A discussion of this method's bias in the presence of noise and how this can be ameliorated using a bundle adjustment like refinement is then presented. The chapter is concluded with a discussion of the method's ability to test for the rigidity of three points in a scene.

- Chapter 4 presents Locally Rigid Motion as a technique for aggregating local rigid models into a piecewise rigid model of global non-rigid motion. This is organized as a recipe that extracts local models using either 3-SFM or a planar template, and resolves their orthographic ambiguities to form a global solution. A discussion of the performance of this algorithm on a variety of datasets is included.

- Chapter 5 presents our global model of a 3D point cloud deforming under the regularization of a set of weak isometric constraints. It is demonstrated how gradient based optimization can find a good local minimum using results from Chapter 4 as an initialization.

- Chapter 6 concludes by summarizing the work here and indicating potential avenues for further work.

Chapter 2

An Overview of Orthographic Structure from Motion

In this chapter, we introduce the necessary mathematics to precisely define the orthographic NRSFM problem. As the most popular approaches to solving this problem are a set of methods based on low-rank matrix factorization, we spend a good deal of the chapter describing these. In particular, we demonstrate how the traditional rigid factorization technique has been generalized in various manners to deal with the non-rigid scenes. We then discuss two alternative sets of approaches that are most comparable to the work here. These are piecewise models that relate to the work in Chapter 4, and deforming point cloud models regularized by isometric constraints that relate to the work in Chapter 5. We generally exclude discussion of perspective models, except when needed to provide appropriate context.

2.1 Mathematical Formulation

Despite there being a variety of ways to formulate the NRSFM problem, we strive to present a single unified notation and formulation, both here and throughout the rest of the document. In general, we assume that the world consists of a deforming scene in which N scene points are orthographically imaged in F frames. In frame f, we denote the position of the n'th scene point as $s_{fn} \in \mathbb{R}^3$ and outline the orthographic imaging procedure that generates the corresponding image point $w_{fn} \in \mathbb{R}^2$. This procedure first applies a rigid transformation, consisting of a rotation $\mathcal{R}_f \in SO(3)$ and a translation $t_f \in \mathbb{R}^3$, that generates the point $p_{fn} \in \mathbb{R}^3$ in the camera's coordinate frame as

$$p_{fn} = \mathcal{R}_f s_{fn} + t_f \ . \tag{2.1}$$

This co-incides with a camera located at $-\mathcal{R}_f t_f$ with a coordinate system specified by the rows of $\mathcal{R}_f$. We assume an orthographic projection model (with no scaling) and thus write the projected image point $w_{fn} \in \mathbb{R}^2$ as

$$w_{fn} = \Pi p_{fn} \tag{2.2}$$

where $\Pi = \left[\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{smallmatrix}\right]$ is an orthographic projection along the Z-axis of the camera. These two equations then simplify to

$$w_{fn} = R_f s_{fn} + h_f \tag{2.3}$$

where $R_f = \Pi \mathcal{R}_f$ is just a combined rotation and orthographic projection and $h_f = \Pi t_f$ is just the 2D translation in the image plane.

In practice, the n'th scene point will be associated with a trajectory of observed image points $\{w'_{fn}\}_{f=1}^{F}$ obtained by tracking feature points [36, 29], sampling trajectories from a distortion field [50], or artificially projecting 3D motion capture data. Regardless, in each frame f, the true projection w_{fn} will not coincide with these observations and thus we write

$$w'_{fn} = w_{fn} + \epsilon_{fn} \ , \tag{2.4}$$

where $\epsilon \in \mathbb{R}^2$ is the error in measuring w_{fn}. It is often assumed that these errors are all independent samples taken from a single Gaussian distribution. In practice, the nature of these errors depends on the measurement procedure used and can deviate greatly from this assumption.

The goal of any (non-rigid orthographic) structure from motion algorithm is to recover information about the underlying scene $\{\{s_{fn}\}_{n=1}^{N}\}_{f=1}^{F}$ and camera motions $\{(\mathcal{R}_f, t_f)\}_{f=1}^{F}$ from these observations $\{\{w'_{fn}\}_{n=1}^{N}\}_{f=1}^{F}$. It is easy to see that this problem is completely unconstrained in the general case, as a wide variety of combinations of scene points, camera orientations and noise models can account equally well for these observations. It is thus imperative to make some assumptions about these elements in order to constrain the problem. Even then, the solution might contain gauge ambiguities such as having an unspecified global coordinate system.

2.1.1 Rigid Model

To illustrate this, we consider the very restricted assumption that the scene is fixed. Equivalently, by utilizing an object centred coordinate frame, we can consider a rigid object as stationary. The situation is similar to that detailed in equations (2.1) and (2.3)

except that now, the scene points are fixed, so we drop the temporal index f from the scene points. This modifies these orthographic projection equations to result in

$$w_{fn} = \Pi(\mathcal{R}_f s_n + t_f) \tag{2.5}$$
$$= R_f s_n + h_f \ . \tag{2.6}$$

If the camera orientations are known beforehand, then 2.6, is a linear constraint on these points. In the noiseless case, the problem is simply that of triangulation and for $F \geq 2$ it is trivial to solve this system and extract the points. If these assumptions are relaxed to admit a Gaussian noise model, then one can still form a linear system of equations and solve this using linear least squares to obtain the statistically optimal result.

If on the other hand, we do not know the camera positions and orientations then the situation becomes more difficult. It is then generally only possible, as we will demonstrate in the next section, to recover the optimal solution when there are no observational errors and $F \geq 3$. When a noise model is considered, one can use this non-optimal solution to initialize a bundle adjustment procedure [55] to try to correct the model by minimizing

$$\min_{\theta} \sum_{f=1}^{F} \sum_{n=1}^{N} \rho_{fn}(\|R_f(\theta)s_n(\theta) + h_f(\theta) - w'_{fn}\|) \ , \tag{2.7}$$

where θ parameterizes the various scene components and $\rho_{fn}(\epsilon)$ encodes the noise model (e.g. $\rho_{fn}(\epsilon) = \epsilon^2$ for i.i.d. Gaussian imaging noise). Unfortunately, such local methods are generally only able to find a local minimum of this function, and thus not the optimal solution. Even when an optimal solution is recovered, there are a number of equally optimal solutions that exist. In fact, every solution to the problem has a number of ambiguities.

The first is an ambiguous global coordinate system. This can be seen as we can apply any rotation $\mathcal{R}$ to the scene points $s'_n = \mathcal{R}s_n$ as long as new cameras rotations specified by $\mathcal{R}'_f = \mathcal{R}_f\mathcal{R}^{-1}$ undoes the effect as

$$\Pi(\mathcal{R}'_f s'_n + t_f) = \Pi(\mathcal{R}_f\mathcal{R}^{-1}\mathcal{R}s_n + t_f) = \Pi(\mathcal{R}_f s_n + t_f) = w_{fn} \ . \tag{2.8}$$

The second is a per-frame ambiguity in the depth of the points in camera coordinates. That is, we can set $t'_f = t_f + \delta \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$ for any $\delta \in \mathbb{R}$ and then,

$$\Pi(\mathcal{R}_f s_n + t'_f) = \Pi\mathcal{R}_f s_n + \Pi t_f + \delta\Pi \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T = R_f s_n + h_f = w_{fn} \ . \tag{2.9}$$

The final ambiguity is referred to as the Necker ambiguity [23] and corresponds to a reflection $\mathcal{B}$ of the world system $s'_n = \mathcal{B}s_n$ where

$$\mathcal{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{2.10}$$

is a flip in depth, along with a corresponding change in rotation $\mathcal{R}'_f = \mathcal{B}\mathcal{R}_f\mathcal{B}^{-1}$. Then we still have that

$$\Pi(\mathcal{R}'_f s'_n + t_f) = \Pi\mathcal{B}\mathcal{R}_f\mathcal{B}^{-1}\mathcal{B}s_n + h_f = R_f\mathcal{B}^{-1}\mathcal{B}s_n + h_f = R_f s_n + h_f = w_{fn} \ . \tag{2.11}$$

To see that $\mathcal{R}'_f$ is a rotation, notice that

$$\mathcal{R}'^T_f\mathcal{R}'_f = \mathcal{B}^T\mathcal{R}_f^T\mathcal{B}^T\mathcal{B}\mathcal{R}_f\mathcal{B} = \mathcal{B}^T\mathcal{R}_f^T\mathcal{R}_f\mathcal{B} = \mathcal{B}^T\mathcal{B} = I \tag{2.12}$$

and

$$\det(\mathcal{R}'_f) = \det(\mathcal{B})\det(\mathcal{R}_f)\det(\mathcal{B}) = (-1)\cdot 1\cdot(-1) = 1 \ . \tag{2.13}$$

2.2 Factorization Methods

Factorization methods encompass a widely successful set of approaches to solving the NRSFM problem. The commonality between each such method, is that the underlying models result in the noiseless image projections living in a low dimensional, linear space. In order to simplify the presentation of these models, we first need to take two short detours.

Centred Model. It turns out that, in the context of factorization approaches, it is often more convenient to work with what we call a "centred" model

$$\hat{w}_{fn} = R_f\hat{s}_{fn} \ , \tag{2.14}$$

in which there is no translation vector. To see why this is helpful, let the average image point be

$$\bar{w}_f = \frac{\sum_{n=1}^{N} w_{fn}}{N} \ , \tag{2.15}$$

and $\hat{w}_{fn} = w_{fn} - \bar{w}_f$ denote a centred image point. If we can find a solution to the centred problem, then we can solve the original problem by simply setting $h_f = \bar{w}_f$ because

$$w_{fn} = \hat{w}_{fn} + \bar{w}_f = R_f \hat{s}_{fn} + h_f \ . \tag{2.16}$$

Conversely, the original model always has an equivalent centred model. To see this, let

$$\bar{s}_f = \frac{\sum_{n=1}^{N} s_{fn}}{N} \ . \tag{2.17}$$

be the average scene point. Then we can see that the average image point is simply

$$\bar{w}_f = \frac{\sum_{n=1}^{N} w_{fn}}{N} = R_f \frac{\sum_{n=1}^{N} s_{fn}}{N} + \frac{\sum_{n=1}^{N} R_f t_f}{N} = R_f \bar{s}_f + h_f \ , \tag{2.18}$$

the projection of the average scene point. Therefore the centred image points $\hat{w}_{fn} = w_{fn} - \bar{w}_f$ and centred scene points $\hat{s}_{fn} = s_{fn} - \bar{s}_f$ satisfy

$$\hat{w}_{fn} = w_{fn} - \bar{w}_f = R_f s_{fn} + R_f t_f - R_f \bar{s}_f - R_f t_f = R_f (s_{fn} - \bar{s}_f) = R_f \hat{s}_{fn} \tag{2.19}$$

which is a centred model, with rotations consistent with the original model.

Matrix Formulation. It is also useful, in the description of these algorithms, to reformulate the centred imaging equations (i.e., (2.1) and (2.3) with no translation) in matrix form. This will allow them to be expressed for all N points and F frames simultaneously. We thus arrange the N 3D scene locations for frame f into the columns of a matrix $S_f \in \mathbb{R}^{3\times N}$ and likewise the corresponding image points as the columns of $W_f \in \mathbb{R}^{2\times N}$. This allows us to write the projection for all N points simultaneously as

$$W_f = \Pi \mathcal{R}_f S_f \ . \tag{2.20}$$

This can be extended further so that equations for multiple frames can be written simultaneously. For this, it is useful to introduce a notational convenience in which a matrix A is written as $A_{M\times N}$ to indicate that the matrix has M rows and N columns. Using this notation, we can specify matrices that include all of the above terms for all frames simultaneously as

$$\mathbf{W}_{2F\times N} = \begin{bmatrix} W_1 \\ \vdots \\ W_F \end{bmatrix} \ , \tag{2.21}$$

$$\mathbf{S}_{3F\times N} = \begin{bmatrix} S_1 \\ \vdots \\ S_F \end{bmatrix} , \tag{2.22}$$

and

$$\mathbf{D}_{3F\times 3F} = \begin{bmatrix} \mathcal{R}_1 & 0 & \cdots & 0 \\ 0 & \mathcal{R}_2 & & \\ \vdots & & \ddots & \\ 0 & & & \mathcal{R}_F \end{bmatrix} \tag{2.23}$$

We can now write all of the imaging equations simultaneously as

$$\mathbf{W} = \Phi\mathbf{D}\mathbf{S} \tag{2.24}$$

where $\Phi_{2F\times 3F} = \mathbf{I}_{F\times F} \otimes \Pi_{2\times 3}$. This reduces, as before, to

$$\mathbf{W} = \mathbf{R}\mathbf{S} \tag{2.25}$$

where $\mathbf{R} = \Phi\mathbf{D} \in \mathbb{R}^{2F\times 3F}$.

Factorization Approach. Factorization methods all fundamentally rely on the estimation of the image motion $\mathbf{W}$ by a low rank J factorization

$$\mathbf{W}_{2F\times N} = \mathbf{M}_{2F\times J}\mathbf{B}_{J\times N} , \tag{2.26}$$

where $\mathbf{M}$ and $\mathbf{B}$ may be forced to respect some problem specific constraints. $\mathbf{B}$ in some sense defines the model of the structure, and $\mathbf{M}$ defines the image formation process. As we will see, the choice of how these matrices are parameterized, how these parameters are set and what constraints are enforced in these matrices implicitly define the fundamental differences between the different factorization models available.

Once a particular model is chosen, an optimization must occur in order to fit the model to the image data, perhaps by minimizing reprojection error or some other quantity. The choice of how to proceed with the minimization and what to minimize is considerably nuanced with closed form solutions being available in only some circumstances, and local minima posing a problem for methods involving bundle adjustment or other non-convex optimizations. Indeed, it is often difficult to interpret claims that good performance has validated the use of a particular model, when it is not clear whether the model itself is contributing to the superior performance or just the optimization.

In this section, we focus on the various models that have been formulated in the factorization paradigm and isolate a few of the more principled ways to fit these models.

2.2.1 Rigid Case

The natural way to proceed is to describe the original factorization algorithm [51] which assumes that the scene is completely rigid as described in Section 2.1.1. For a centred model, with no noise, the motion matrix $\mathbf{W}$ will be of rank-3. To see why this is, we first set the columns of a matrix $\mathbf{B}_{3\times N}$ to be the fixed 3D coordinates of the N points. By stacking each of the rotational projection $R_f = \Pi\mathcal{R}_f$ into a $2F \times 3$ motion matrix $\mathbf{M}_{2F\times 3}$ we obtain the factorization $\mathbf{W}_{2F\times N} = \mathbf{M}_{2F\times 3}\mathbf{B}_{3\times N}$ similar to equation 2.26 showing that under this model, $\mathbf{W}$ has rank 3.

Recovery

Note that a rank-3 factorization of $\mathbf{W}$ is easily provided by $\hat{\mathbf{M}} = U_{:,1:3}$ and $\hat{\mathbf{B}} = D_{1:3,1:3}(V_{:,1:3})^T$ where UDV^T is the Singular Value Decomposition (SVD) of $\mathbf{W}$. Unfortunately, this does not respect the constraints defined by the model of the motion matrix $\mathbf{M}$, mainly that each pair of rows $M_f = R_f$ corresponding to frame f are orthornormal vectors. That said, both $\mathbf{M}$ and $\hat{\mathbf{M}}$ span the rank 3 column space of $\mathbf{W}$ and thus there must be a non-singular linear transformation Q such that $\mathbf{M} = \hat{\mathbf{M}}Q$ and $\mathbf{B} = Q^{-1}\hat{\mathbf{B}}$. To find this matrix Q we can use M_f, the two rows of $\mathbf{M}$ associated with frame f, to write:

$$\hat{M}_f Q Q^T \hat{M}_f^T = M_f M_f^T = I_2 \tag{2.27}$$

which gives us three linear constraints on the entries of the symmetric matrix $G = QQ^T$. With enough frames we can solve for G and factor it to the form QQ^T recovering the model $\mathbf{M} = \hat{\mathbf{M}}Q$ and $\mathbf{B} = Q^{-1}\hat{\mathbf{B}}$. This step is referred to as the euclidean upgrade.

In the presence of noise, we can still use SVD to obtain a rank-3 approximation $\mathbf{W}^* = \hat{\mathbf{M}}\hat{\mathbf{B}}$ to $\mathbf{W}$. It is not true, however, that the columns of $\hat{\mathbf{M}}$ span the column space of $\mathbf{W}$, only that of $\mathbf{W}^*$ and thus there is no guarantee of a rectification transformation Q even existing. That said, for any non-singular matrix Q, one can transform the SVD solutions via $\tilde{\mathbf{M}} = \hat{\mathbf{M}}Q$ and $\tilde{\mathbf{B}} = Q^{-1}\hat{\mathbf{B}}$ to yield $\tilde{\mathbf{M}}\tilde{\mathbf{B}} = \hat{\mathbf{M}}QQ^{-1}\hat{\mathbf{B}} = \hat{\mathbf{M}}\hat{\mathbf{B}} = \mathbf{W}^*$, an equally good rank-3 approximation $\mathbf{W}$. Therefore, we can still proceed by finding the matrix Q that satisfies each of the above orthogonality constraints (2.27) in the least squares sense as long as the recovered G can actually be factored.

Although, this is the advocated methodology in the literature, one should note that the orthogonality constraints will not have been exactly enforced in the presence of noise.

When the approximate factorization is performed, some of the noise will have leaked into the camera model and the structure matrix after the upgrade. This will leave the pairs of rows in the $\tilde{\mathbf{M}}$ merely representing the affine cameras that are the closest to satisfying the orthonormality constraints but that can also exactly reproduce the rank-3 approximation $\mathbf{W}^*$. A model that exactly satisfies those constraints (2.27) can be recovered by orthonormalizing each row pair either via Gram-Schmidt or a convex optimization [38]. This, however, only corrects the camera models and does not repair any damage that has been caused in the structure $\tilde{\mathbf{B}}$. Therefore, it would be ideal to perform a final bundle adjustment akin to minimizing reprojection error (i.e., Equation (2.7)), while respecting the model's constraints.

2.2.2 Shape and Trajectory Basis Representation

In [11] the factorization approach was adapted to the NRSFM paradigm by assuming that the the scene point set in each frame, $S_f \in \mathbb{R}^{3\times N}$, is a linear combination

$$S_f = \sum_{k=1}^{K} c_{fk} B_k \tag{2.28}$$

of K basis shapes $B_1, ..., B_K \in \mathbb{R}^{3\times N}$. This is a generalization of the rigid factorization method and reduces to it when $K = 1$. What seemed like an elegant and straightforward generalization, however, turned out to present very subtle difficulties to model recovery that took almost a decade to characterize. Regardless, the model itself has proven immensely useful and has spawned a wide variety of generalizations.

To create a matrix formulation corresponding to this model, we define a matrix of shape coefficients

$$\mathbf{C}_{F\times K} = \begin{bmatrix} c_{11} & \cdots & c_{1K} \\ \vdots & \ddots & \vdots \\ c_{F1} & \cdots & c_{FK} \end{bmatrix} \tag{2.29}$$

and a corresponding stacked shape matrix

$$\mathbf{B}_{3K\times N} = \begin{bmatrix} B_1 \\ \vdots \\ B_k \end{bmatrix} . \tag{2.30}$$

This allows us to write the observation matrix $\mathbf{W}$ as

$$\mathbf{W} = \mathbf{R}(\mathbf{C} \otimes \mathbf{I}_3)\mathbf{B} \ . \tag{2.31}$$

If we let $\mathbf{M}_{3F\times 3K} = \mathbf{R}(\mathbf{C} \otimes \mathbf{I}_3)$, we see that we again have a similar equation $\mathbf{W}_{2F\times N} = \mathbf{M}_{2F\times 3K}\mathbf{B}_{3K\times N}$ to that of equation (2.26), and thus $\mathbf{W}$ can be seen to be of rank-$3K$. One can again use SVD to recover a rank-$3K$ approximation $\hat{\mathbf{M}}\hat{\mathbf{B}}$, but the question of how to best "upgrade" the matrix $\hat{\mathbf{M}}$ under these assumptions has been the subject of great debate [62, 9, 10, 2]. In particular, one is looking for a non-singular $3K \times 3K$ matrix Q such that $\mathbf{M} = \hat{\mathbf{M}}Q$ or at least as close as possible to satisfying the constraints implicit in $\mathbf{M}$. The simple algorithm proposed by [11] was shown [9, 62] to not employ enough constraints to always extract the correct solution, without some sort of additional regularizing priors. Later work [10, 2, 16] then demonstrated that such regularization is not needed if a previously ignored rank constraint was incorporated. It was suggested that the main difficulty in fitting such a model was that of optimization, as the objective that must be minimized has many local minima [2]. In [16], however, a convex relaxation of this rank constraint was employed to obtain a solution that appears to always produce the correct solution in the noise-free case.

Throughout this duration of distress over how to proceed with the upgrade of a rank-$3K$ matrix factorization to a K basis shape model, a variety of alternative approaches were proposed. These not only helped bypass the upgrading issues, but also improved the practical performance by adding constraints through priors or fitting the model in a different way. For example, [54] used a rigid initialization followed by coordinate descent over the parameters. By parameterizing the rotations explicitly using the exponential map the need to perform an upgrade was avoided. In contrast, [38] avoids the exponential map and instead projects their solution to the manifold of rotations at each step.

A variety of methods assume deformation modes are added to a mean shape so that they can be more readily regularized. In [5, 21] the modes are fit incrementally to the remaining unexplained variance. In [53, 52] a Gaussian prior is used on the shape coefficients $\mathbf{C}_f \sim N(0, \mathbf{I}_K)$ that drive them to zero when not needed. The parameters are then estimated via Expectation-Maximization (EM) by alternately estimating the coefficients and then maximizing the likelihood of the data matrix $\mathbf{W}$. They also consider the possibility of a temporal smoother via $\mathbf{C}_{f+1} = \Phi\mathbf{C}_f + N(0, \Sigma)$ where $\Phi_{K\times K}$ and $\Sigma_{K\times K}$ are additional variables defining the temporal smoothness of the sequence fit by the EM framework. In [37, 5], temporal and spatial smoothness priors take the form of additional terms in a minimization framework. In [41], the shape coefficients are recovered first as

the resultant vectors from a multidimensional scaling framework.

2.2.3 Trajectory Basis Models

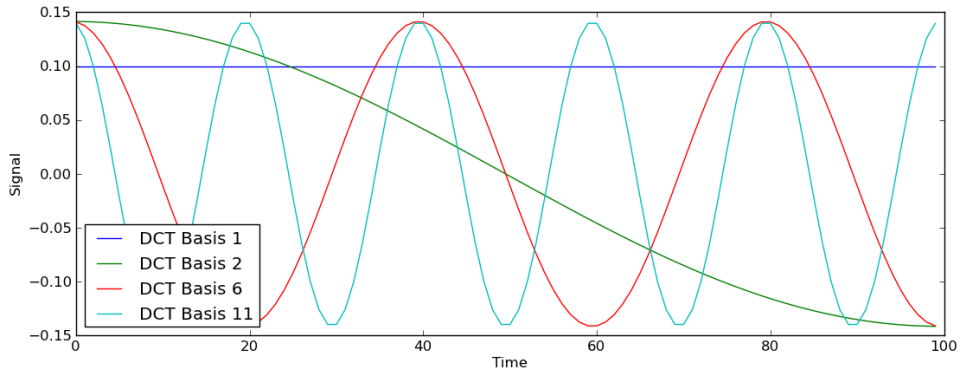

Figure 2.1: Some representative vectors from the DCT basis.

There are a vast amount of variables that need to be estimated in the general shape basis model. However, the success of temporal smoothness priors [52, 1] that implicitly add constraints to the model hint that many of these variables are redundant in video sequences. In this light, [4, 3] proposed to fix the column space using a basis of temporal trajectories. To this end, they utilize the discrete cosine transform (DCT) as shown in Figure 2.1 and defined[1] by the continuous functional

$$\omega_d(f) = \frac{\sigma_d}{\sqrt{F}} \cos\left(\frac{\pi(2f-1)(d-1)}{2F}\right) \tag{2.32}$$

where $f \in [1, F]$, $\sigma_1 = 1$ and $\sigma_d = 2$ for $d \geq 2$. This functional is discretely sampled in each frame $f \in \{1, ..., F\}$ as $\Omega_{fd} = \omega_d(f)$. One can then write the trajectory of point n as

$$S_{fn} = \sum_{k=1}^{K} \Omega_{fk} B_{kn} \tag{2.33}$$

where $B_{kn} \in \mathbb{R}^3$ specifies the amount of the X, Y and Z signal to add in from the k'th DCT trajectory for the n'th point. In matrix form, this is just

$$\mathbf{S}_{3F\times N} = (\Omega_{F\times K} \otimes I_{3\times 3})\mathbf{B}_{3K\times N} \ . \tag{2.34}$$

Note that this takes the same form as the shape basis model, except now $\mathbf{C} = \Omega$ is a

[1]This definition is from [22]

fixed matrix containing these low frequency smooth DCT vectors.

In this framework, the upgrade step is considerably easier as the extra constraints are shown to be enough [2] to avoid the difficulties detailed earlier. Regardless, the major advantage of this model is that many degrees of freedom have been removed by fixing the trajectory basis. This is much easier to do in trajectory space than shape space because a basis like the DCT can describe a wide variety of common motions. This is explored empirically in [3] by doing a principle component analysis of motion capture data and noting the similarity between the principle trajectory vectors and the low frequency DCT basis vectors.

Shape Trajectory Approach. A further generalization [22] of the trajectory basis method constructs the K trajectories as a linear combination of a potentially much wider spectrum $D > K$ of DCT basis vectors

$$\mathbf{C}_{F\times K} = \Omega_{F\times D}\mathbf{X}_{D\times K} \ . \tag{2.35}$$

It was demonstrated that this is quite helpful when there are high frequency deformations occurring in a sequence. Indeed, the truncated basis here $\Omega_{F\times D}$ contains $D-K$ higher frequencies than the corresponding standard trajectory basis model. The coefficients $\mathbf{X}_{D\times K}$ determine how to combine these into a new set of trajectories $\mathbf{C}_{F\times K}$ for the model to use while the factorization of $\mathbf{W}$ still remains constrained and of rank $3K$. Of course if $\mathbf{X} = \begin{bmatrix}\mathbf{I}_K \\ \mathbf{0}\end{bmatrix}$ is fixed, then this reduces to the standard trajectory approach. This is, indeed, exactly how they obtain an initialization to a non-linear optimizer to fit the model.

Kernel Shape Trajectory Approach. In order to deal with some of the difficulties these methods have modelling nonlinear deformation, the model presented in [20] uses the kernel trick. Points in a very low dimensional space are mapped into a higher dimensional shape space to select among a set of K of implicit basis shapes. This is done by specifying the coefficients using a kernel matrix $\mathbf{C}_{F\times K} = \mathbf{K}_{F\times K}$ with entries

$$\mathbf{K}_{fk} = \kappa(c_f, b_k) = e^{-\gamma\|c_f - b_k\|^2} \tag{2.36}$$

where c_f and b_k exist in a very low dimensional shape coordinate space $\mathbb{R}^H$. The points $b_1, \cdots, b_K$ are associated with the K shapes $B_1, ..., B_k$ that make up $\mathbf{B}$. Given the shape coordinate c_f, the kernel function $\kappa(c_f, b_k)$ then specifies how much of shape B_k to add

into frame f. Indeed, the estimated 3D point set S_f in some frame f can be recovered as

$$S_f = \sum_{k=1}^{K} \kappa(c_f, b_k) B_k \ . \tag{2.37}$$

This means that the b_k's and c_f's which need to be estimated can in practice exist in a much lower dimension $H << K$ than the $3 \times K$ dimensional shape space because they only interact through the κ function.

The number of parameters is reduced even further by again assuming that the shape coordinate trajectory in $\mathbb{R}^H$ itself is smooth and can be modelled as

$$c_f^T = \begin{bmatrix} \omega_1(f) & ... & \omega_D(f) \end{bmatrix} \mathbf{X}_{D \times H} \ . \tag{2.38}$$

This is very similar to (2.35) but now note that the coefficient matrix $\mathbf{X}$ has only $H << K$ columns that need to be estimated. Furthermore, the points $b_1, ..., b_K$ in the shape coordinate space are assumed to lie on this continuous trajectory that these F samples represent, and thus a single continuous variable $t_k \in [1, F]$ is used to specify each as

$$b_k^T(t_k) = \begin{bmatrix} \omega_1(t_k) & ... & \omega_D(t_k) \end{bmatrix} \mathbf{X}_{D \times H} \ . \tag{2.39}$$

Figure 2.2 illustrates this situation in the common setting of $H = 2$.

This leaves the kernel matrix $\mathbf{K}$ parametrized by only $DH + K + 1$ parameters to estimate[2]. In practice D is often set to $0.1F$ and H was set to 2 yielding around $0.2F + K$ parameters as opposed to the FK parameters in the coefficient matrix of the unconstrained shape model.

2.2.4 Locally Linear Manifold

All of the methods discussed above have assumed that the space of shape deformation across time is a linear subspace of finite dimension. In [40], this assumption is relaxed as to require only that the shape in each frame lie on a locally linear manifold of finite dimension. In essence, the model reduces to a rigid shape model for each frame $w_f = R_f B_f$, under the constraint that the shape B_f lies on that manifold and deforms smoothly in time. The method relies heavily on a heuristic initialization that finds clusters $C_1, ..., C_L$ of frames that give a valid rigid interpretation $W_f \approx R_f \hat{S}_l$, $\forall f \in C_l$. The shape in each frame is then initialized via $B_f = \hat{S}_l$ before optimization. Unfortunately, this model

[2]This includes the DH entries in $\mathbf{X}$, the K t_k's and γ from equation 2.36.

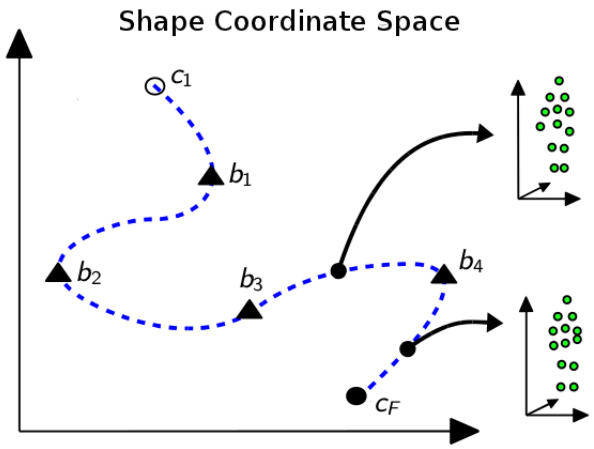

Figure 2.2: A two dimensional shape coordinate space. Figure adapted from [20].

has many parameters, seems brittle due to the awkward initialization via clustering and appears to be largely eclipsed by more recent methods.

2.2.5 Quadratic Deformation Model

The quadratic deformation model [19] is similar to the shape basis model as it assumes that the point set has a finite set of deformation modes present in $\mathbf{B}$. The difference is that here it has a special form in which it is parametrized by a fixed rest shape.

$$B_1 = \begin{bmatrix} X_1 & \cdots & X_N \\ Y_1 & \cdots & Y_N \\ Z_1 & \cdots & Z_N \end{bmatrix} . \tag{2.40}$$

Its structure matrix $\mathbf{B}$ has the form

$$\mathbf{B}(B_1) = \begin{bmatrix} B_1 \\ B_2(B_1) \\ B_3(B_1) \end{bmatrix} \tag{2.41}$$

where

$$B_2(B_1) = \begin{bmatrix} X_1^2 & \cdots & X_N^2 \\ Y_1^2 & \cdots & Y_N^2 \\ Z_1^2 & \cdots & Z_N^2 \end{bmatrix} \tag{2.42}$$

and

$$B_3(B_1) = \begin{bmatrix} X_1Y_1 & \cdots & X_NY_N \\ Y_1Z_1 & \cdots & Y_NZ_N \\ Z_1X_1 & \cdots & Z_NX_N \end{bmatrix} . \tag{2.43}$$

Each mode is transformed through a full set of unconstrained linear transformations in each frame so that the full model is written as

$$\mathbf{W}_{2F\times N} = \mathbf{D}_{2F\times 3F} \begin{bmatrix} \Gamma_{3F\times 3} & \Omega_{3F\times 3} & \Lambda_{3F\times 3} \end{bmatrix} \mathbf{B}_{9\times N}$$

This specifies $3F$ nearly general[3] quadratic mappings from the 3-dimensional rest shape space and specifies the per-frame pose and deformation. It is shown that such a formulation can produce a wide range of natural deformations. In particular, Γ provides a per-frame linear transformation of the rest shape B_1 such as stretching, Ω provides bending through the quadratic terms in B_2 and Λ provides twisting through the cross terms in B_3.

A serious drawback of this method though is that it assumes that the rest shape B_1 is fixed, centred and axis aligned. They suggest that this can be done through a rigid factorization of a short sequence of the target object at rest. It is then suggested to initialize Γ with F identity matrices and $\Omega = \Lambda = \mathbf{0}$ as this corresponds to the same transformation provided by the rigid factorization model. A bundle adjustment is performed to refine the rotations and learn the deformation coefficients.

2.3 Piecewise Models

The models that have been discussed so far try to model the entire set of point tracks simultaneously in a global model. Another option is to fit simpler models to smaller local sets of image trajectories, and then combine these into a global model. These piecewise models have the advantage that there may be common constraints on local deformation that apply to a diverse set of sequences, despite their global deformations being drastically different. These models, however, must decide how to divide the set of N trajectories, which local model to fit to each piece, and how to stitch these models together.

The approach presented in Chapter 4, and based on the work [49], chooses to use rigid triangle models to explain local triplets of trajectories. This has the advantage that the local rigidity assumption leveraged here is more likely to be valid with just three points.

[3] No constant term is available.

The planar nature of these triangles, however, admits a per-frame depth flip ambiguity, that is challenging to resolve given the limited number of points two models can share. We refer the reader to Chapter 4 for additional details.

An alternative and more standard approach is to instead model a modest number of points in a local neighbourhood. The intuition is that reasonably flexible models can be fit to relatively large but local subsets of trajectories. Further, the sets of trajectories attached to each model can be allowed to have a large intersection making it easier to integrate these models together. The local models that have been considered have been restricted to planar [59, 13] and quadratic deformation models [18] but of course a wide variety of SFM models could be used.

With these larger models, however, how to best divide the trajectories into subsets is still an open question. In [13], it is required that the user provide a reference frame and region of interest in which points are not occluded. In [18], a coarse rigid fit of the entire point set was used to embed the tracks in three dimensions. The bounding box of this set was then subdivided with each subdivision being used to initialize a local quadratic deformation model. In [43], an alternation is performed between model fitting and model assignment, allowing local models to grow to include inlier points or shrink by removing outliers.

2.4 Point Clouds with Pairwise Isometric Constraints

The above models attempt to explicitly restrict local deformation by fitting low capacity models to local image trajectories. An attractive alternative is to instead form a global model of all image trajectories that penalizes violations of the local deformation assumptions. The local deformation assumption that this thesis focuses on is local rigidity, which can be encoded using isometric constraints between two points as is done in Chapter 5. A similar, but uniquely different formulation is proposed in [60]. For completeness, we briefly describe the essence of this approach, but a more in depth explanation is deferred to Section 5.1.5.

In [60], they first acquire a set of pairs of points $\mathcal{L} \subseteq \{1, ..., N\}^2$ that should remain at a fixed distance to each other. They then parameterize their model with a vector θ which includes the assumed interpoint distance $L_{nm}(\theta)$ that a pair of points $(n, m) \in \mathcal{L}$ should remain. For the n'th scene point, they also dedicate a component of θ to its depth in each frame f, and thus one can write $z_{fn}(\theta)$ to indicate this dependence. They do not, however, parameterize the x and y components of the point, and instead constrain that

point to lie on the back projected ray through $w'_{fn} = \begin{bmatrix} x'_{fn} \\ y'_{fn} \end{bmatrix}$. They can thus write

$$p_{fn}(\theta) = \begin{bmatrix} x'_{fn} \\ y'_{fn} \\ z_{fn}(\theta) \end{bmatrix} . \tag{2.44}$$

They then formulate an energy based model

$$E(\theta; \mathcal{L}) = \sum_{f=1}^{F} \sum_{(i,j)\in\mathcal{L}} |\|p_{fi}(\theta) - p_{fj}(\theta)\| - L_{ij}(\theta)| \tag{2.45}$$

which they desire to minimize.

They choose to view the energy (5.14) as that of a Markov Random Field in which each term represents a potential over a clique of three variables. In order to optimize this energy, they choose to use a discrete optimization strategy in which, at each step, a proposal solution θ' is "fused" [35] to the current solution θ_k in such a way such that the resulting solution $\theta_{k+1} = FUSE(\theta_k, \theta')$ does not increase the energy (i.e., that $E(\theta_{k+1}; \mathcal{L}) \leq E(\theta_k; \mathcal{L})$ and $E(\theta_{k+1}; \mathcal{L}) \leq E(\theta'; \mathcal{L})$).

Chapter 3

Three Point Rigid Structure from Motion

In this chapter, we consider what can be said about 3 rigid scene points that have been orthographically imaged from F viewpoints as illustrated in Figure 3.1. Previous work has focused on the exact case, where it is assumed that no noise has entered the imaging process [7, 24]. That work has aimed to identify, given a set of image projections, whether such a rigid configuration exists to explain those projections and if so, how many such configurations exist. Although the proofs of these configurations' existence are often constructive in nature, attention is not generally paid to producing practical algorithms for their recovery. Indeed, they only consider explaining the projections in at most four views whereas a practical estimation procedure will need to integrate information from multiple frames or views. This is of course not surprising given that they consider only the noiseless case, in which four views effectively determine the intrinsic geometry of the scene. Adding another view will either agree with this geometry or invalidate it, as their is no allowance for the observations to deviate from the model. In contrast, we admit the possibility of noise, where the integration of observations from multiple views is likely essential to reliably estimating scene geometry.

This chapter begins by mathematically formalizing the orthographic three point rigid model and the unique depth flip ambiguities that it admits. We demonstrate that the projections in each new view yield a linear constraint on the coordinate free geometry of the scene, providing a straightforward method for recovering the rigid model in the noise free case. In the presence of noise, linear least squares provides a natural route to fitting a model that averages error from multiple frames. As this model cannot exactly reproduce the image observations, a Gaussian noise model is assumed and each viewpoint (exterior orientation) is optimized to maximize the likelihood under this model. This,

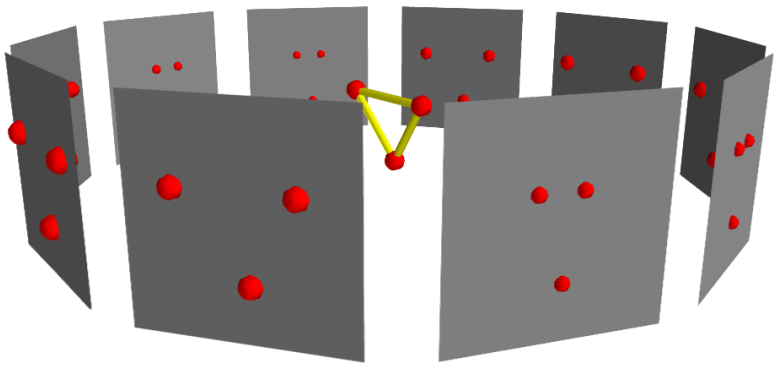

Figure 3.1: An illustration of the orthographic three point rigid model. The three points can be seen as a triangle that is imaged from a variety of viewpoints. The location of these points in the world, and the orientation and position of the viewpoints specify the model.

however, is not the maximum likelihood solution as the intrinsic geometry has been held fixed. We therefore use this as an initialization for a bundle-adjustment like procedure that simultaneously optimizes this geometry. Evidence on synthetic data indicates that this nearly always provides the optimal solution in the presence of reasonable amounts of Gaussian noise. We also demonstrate that a simple prior on the link lengths can be used to regularize the solution in degenerate cases caused by a lack of viewpoint variation. Finally we explore the procedure's ability to quantify the rigidity of three points.

3.1 Formulation

We can consider the three point rigid model to be a clear specialization of the more general rigid model described in Section 2.1.1 where now $N = 3$. This again requires the modification of the general orthographic projection equations (2.1) and (2.3) to utilize scene points that are fixed. We thus drop the temporal index f from the scene points in those equations (i.e., $s_{fn} = s_n$ for any f). Written explicitly, these equations are then

$$w_{fn} = \Pi(\mathcal{R}_f s_n + t_f) \tag{3.1}$$

and

$$w_{fn} = R_f s_n + h_f \tag{3.2}$$

replicated for each frame $f \in \{1, .., F\}$ and each point index $n \in \{1, 2, 3\}$.

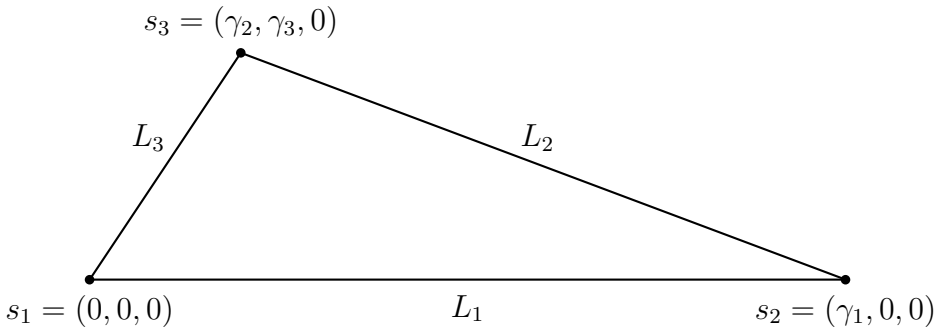

Figure 3.2: The geometry of a triangle has three degrees of freedom. There is a one to one mapping between the labelled lengths $L \in \mathcal{T}_L \subseteq \mathbb{R}^3$ and our assumed parameterization $\gamma \in \mathcal{T}_\gamma \subseteq \mathcal{T}_\gamma^{\pm}$. In addition, there is a second $\gamma^{\pm} \in \mathcal{T}_\gamma^{\pm}$ that corresponds to a triangle reflected across the x axis, that does not exist in $\mathcal{T}_\gamma$.

3.1.1 Parameterization as a Triangle

With only three fixed world points, it is natural to think of these points as the labelled vertices of a triangle in 3D that is imaged from multiple viewpoints as illustrated in Figure 3.1. As the world coordinate frame is ambiguous, it is natural to consider as equivalent any two such triangles that are related by a 3D rotation and translation. We will show that there is a one-to-one correspondence between these equivalence classes of triangles and their labelled triangle lengths $L_i = \|s_{i \bmod 3+1} - s_i\|$ for $i \in \{1, 2, 3\}$. The space of such lengths is

$$\mathcal{T}_L = \{(L_1, L_2, L_3) : L_1, L_2, L_3 > 0; L_1 + L_2 + L_3 - \max_i L_i > \max_i L_i\} \subseteq \mathbb{R}^3 \tag{3.3}$$

where the last constraint is just the triangle inequality. Despite this being a somewhat obvious result, its development here is justified as we introduce notation that will be used later on. Further, the following sub-result will be useful in understanding an argument about "counting" rigid interpretations put forth in Section 3.2.3.

Result 1. *Up to a rigid rotation and translation in the x-y plane there exists two possible triangles in the x-y plane with lengths $L \in \mathcal{T}_L$. These triangles take the form $s_1(L) = (0, 0, 0)$, $s_2(L) = (\gamma_1(L), 0, 0)$ and $s_3(L) = (\gamma_2(L), \gamma_3^{\pm}(L), 0)$ with*

$$\gamma_1(L) = L_1 > 0 \tag{3.4}$$

$$\gamma_2(L) = \frac{1}{2L_1}(L_3^2 - L_2^2 + L_1^2) > 0 \tag{3.5}$$

$$\gamma_3^{\pm}(L) = \pm\sqrt{L_3^2 - \gamma_2^2(L)} \neq 0 \ . \tag{3.6}$$

Proof. The geometry of this is illustrated in Figure 3.2. For any triangle in the x-y plane with labelled lengths L, it is clear that a rigid in-plane rotation and translation puts s_1 at the origin and s_2 at $(\gamma_1(L), 0, 0)$ with $\gamma_1(L) > 0$, so assume that is done. Now s_3 must be in the $x-y$ plane at distances L_2 and L_3 to points s_2 and s_3 respectively. The points of intersection of the two circles that these points define are at the two possible values of $s_3(L)$ detailed above. This creates two triangles, one above the x-axis and one below, with counter-clockwise ordering of their points as $(1, 2, 3)$ and $(1, 3, 2)$ when observed from the positive z-direction. Any in-plane rigid transformation, will preserve such an ordering and thus these triangles cannot be equivalent up to such a transformation. □

For reasons that will become clear in Section 3.2.3, where we discuss how related work may be counting rigid interpretations, one might then consider the space of such triangles as

$$\mathcal{T}_\gamma^{\pm} = \{(\gamma \in \mathbb{R}^3 : \gamma_1 > 0, \gamma_3 \neq 0\} \subseteq \mathbb{R}^3 \ . \tag{3.7}$$

For our purposes, however, we require this straightforward corollary.

Corollary 1. *Up to 3D rigid rotations and translations there exists a unique triangle with lengths $L \in \mathcal{T}_L$. This triangle takes the form $s_1(L) = (0, 0, 0)$, $s_2(L) = (\gamma_1(L), 0, 0)$ and $s_3(L) = (\gamma_2(L), \gamma_3(L), 0)$ with*

$$\gamma_3(L) = \sqrt{L_3^2 - \gamma_2^2(L)} \ . \tag{3.8}$$

Proof. Given any such triangle, there exists a rigid transformation that maps it into the plane. By the above result, there exists another rigid transformation that maps it to a triangle specified by $\gamma \in \mathcal{T}_\gamma^{\pm}$. Now if $\gamma_3 > 0$, we are done. If $\gamma_3 < 0$, then a final rigid rotation around the x-axis of 180 degrees, will set γ_3 to $-\gamma_3 > 0$ □

We thus fix the world coordinate system by assuming that the scene points to be parameterized by some vector $\gamma \in \mathcal{T}_\gamma$ where

$$\mathcal{T}_\gamma = \{(\gamma \in \mathbb{R}^3 : \gamma_1 > 0, \gamma_3 > 0\} \subseteq \mathbb{R}^3, \tag{3.9}$$

unless otherwise stated.

3.1.2 Ambiguities

The planar nature of such a triangle model induces a special per-frame depth flip ambiguity in each viewpoint's cameras coordinates. Mathematically, this flip takes the form

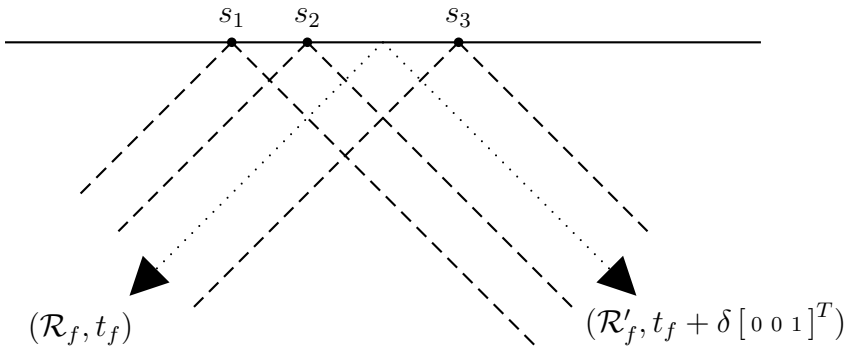

Figure 3.3: Illustration of depth flip ambiguity in a two dimensional scene. There are two different camera orientations which allow the three scene points s_1, s_2 and s_3 to project to the same points on the image plane. The relative depths of the points with respect to the camera gets flipped.

of another rotation $\mathcal{R}'_f = \mathcal{B}\mathcal{R}_f\mathcal{B}$ where

$$\mathcal{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{3.10}$$

is a flip through the x-y plane. To see that $\mathcal{R}'$ is a rotation, the reader is referred to equations (2.12) and (2.13). In frame f, the camera coordinates of point n becomes

$$\mathcal{R}'_f s_n + t_f = \mathcal{B}\mathcal{R}_f\mathcal{B}s_n + t_f = \mathcal{B}\mathcal{R}_f s_n + t_f \tag{3.11}$$

where the last equality is due to the s_n having zero z component. This means that the point has flipped its z-component around the z-component of t_f. As the z-component of t_f is also ambiguous, it is convenient to just refer to this ambiguity as a depth flip ambiguity.[1] Regardless, under this new interpretation, the projected points remain the same as

$$\Pi(\mathcal{R}'_f s_n + t_f) = \Pi\mathcal{B}\mathcal{R}_f s_n + h_f = \Pi\mathcal{R}_f s_n + h_f = R_f s_n + h_f \ . \tag{3.12}$$

This ambiguity is illustrated for a two dimensional model in Figure 3.3. These 2^F ambiguous depth flip assignments, that exist in F frames, are of particular interest to the reader of this document, as a good deal of the next chapter is devoted to disambiguating them.

[1]Since we are dealing with a planar model, the "depth flip ambiguity" is equivalent to the ambiguity in the 3D rotation, $\mathcal{R}_f$ or $\mathcal{R}'_f$

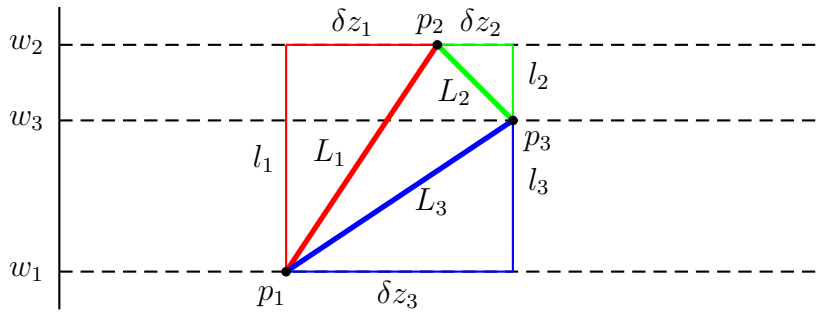

Figure 3.4: Top view of three points in camera coordinates p_1, p_2, p_3 being projected to their image points w_1, w_2, w_3 in the image plane. The thick lines form the triangle with vertices representing the points. Each side i has link length L_i that is the hypotenuse of a right triangle (coloured either red, green or blue) formed with the projected link length l_i and the relative depth δz_i.

3.2 Strictly Rigid Noise Free Model

Here we consider the problem of finding three point rigid models that *exactly* explain a set of image trajectories. The problem of counting such rigid interpretations in $F = 3$ or $F = 4$ frames has been considered previously [7, 24]. In contrast, we demonstrate how a simple algorithm allows for the recovery of solutions in $F \geq 3$ frames when they exist.

3.2.1 Linear Recovery of Structure

In [49], it was demonstrated that there is a succinct relationship between the edge lengths $L \in \mathbb{R}^3$ of a triangle and their projected lengths $l_f \in \mathbb{R}^3$ in each frame f. This takes the form of a quadratic constraint between the *squared* edge lengths $M \in \mathbb{R}^3$ and the *squared* image lengths $m_f \in \mathbb{R}^3$.

Result 2. *Let $w_i = \Pi p_i$ be the orthographic projection of a point p_i for $i \in \{1, 2, 3\}$. Then if $M \in \mathbb{R}^3$ is the vector of squared link lengths (i.e., $M_i = \|p_j - p_i\|^2$ where $j = i \bmod 3+1$) and $m \in \mathbb{R}^3$ is the vector of squared projected link lengths (i.e., $m_i = \|w_j - w_i\|^2$ where $j = i \bmod 3 + 1$), then*

$$M^T A M - 2M^T A m + m^T A m = 0 \tag{3.13}$$

where

$$A = \begin{bmatrix} 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{bmatrix} . \tag{3.14}$$

Proof. The geometry is illustrated in Figure 3.4. Let z_i be the depth of point i. The relative depth change across link i is $\delta z_i = z_{i \bmod 3+1} - z_i$. Using the right triangle that this relative depth forms with the link in the image plane, we see that

$$\delta z_i^2 = L_i^2 - l_i^2 \tag{3.15}$$
$$= M_i - m_i \ . \tag{3.16}$$

Further, all three links form a closed loop and thus their relative depths must add to zero

$$\delta z_1 + \delta z_2 + \delta z_3 = 0 \ . \tag{3.17}$$

Rearranging and squaring both sides we have

$$\delta z_1^2 = (\delta z_2 + \delta z_2)^2 \tag{3.18}$$
$$= \delta z_2^2 + 2\delta z_2 \delta z_3 + \delta z_3^2 \tag{3.19}$$

or equivalently that

$$\delta z_1^2 - \delta z_2^2 - \delta z_3^2 = 2\delta z_2 \delta z_3 \ . \tag{3.20}$$

Squaring both sides again, we have that

$$(\delta z_1^2 - \delta z_2^2 - \delta z_3^2)^2 = 4\delta z_2^2 \delta z_3^2 \ . \tag{3.21}$$

Rearranging, we have that

$$0 = (\delta z_1^2 - \delta z_2^2 - \delta z_3^2)^2 - 4\delta z_2^2 \delta z_3^2 \tag{3.22}$$
$$= \sum_{i=1}^{3} \delta z_i^2 - 2\delta z_1^2 \delta z_2^2 - 2\delta z_1^2 \delta z_3^2 + 2\delta z_2^2 \delta z_3^2 - 4\delta z_2^2 \delta z_3^2 \tag{3.23}$$
$$= \sum_{i=1}^{3} \delta z_i^2 - 2\sum_{i=1}^{3}\sum_{j=i+1}^{3} \delta z_i^2 \delta z_j^2 \ . \tag{3.24}$$

Since this now only contains squared relative depths, we can plug in (3.16) and continue as

$$0 = \sum_{i=1}^{3} (M_i - m_i) - 2\sum_{i=1}^{3}\sum_{j=i+1}^{3} (M_i - m_i)(M_j - m_j) \tag{3.25}$$
$$= (M - m)^T A (M - m) \tag{3.26}$$
$$= M^T A M - 2M^T A m + m^T A m \tag{3.27}$$

to obtain the quadratic equation in (3.13). □

Note that equation (3.13) can be said to be coordinate free as it does not involve the three original points explicitly. Equations from different frames can thus be combined as long as they share the common unknown link lengths. In particular, by subtracting the equation for frame 1 from the other $F-1$ equations, we can cancel out each quadratic term M^TAM leaving $F-1$ linear equation in M. This results in the following corollary.

Corollary 2. *Given three 3D points with squared link length vector $M \in \mathbb{R}^3$, orthographically projected to F views with corresponding squared projected link lengths $m_1, ..., m_F \in \mathbb{R}^3$,*

$$B_{(F-1)\times 3} M_{3\times 1} = b_{(F-1)\times 1} \tag{3.28}$$

where

$$B = 2 \begin{bmatrix} m_1^T - m_2^T \\ \vdots \\ m_1^T - m_F^T \end{bmatrix} \tag{3.29}$$

and

$$b = \begin{bmatrix} m_1^T A m_1 - m_2^T A m_2 \\ \vdots \\ m_1^T A m_1 - m_F^T A m_F \end{bmatrix} . \tag{3.30}$$

When $F = 4$, this matrix B will be generically of rank 3, as demonstrated numerically in Figure 3.5, and thus we can recover M using linear least squares. If the estimate of M has no negative component, which empirically always occurs in the noiseless case, then the link lengths can be recovered by taking an element-wise square-root. When $F = 3$, this matrix will be generically of rank 2 and thus we can find its null vector v_{null}, and a single solution M_0 such that

$$BM_0 = b . \tag{3.31}$$

This allows us to parameterize the solutions that satisfy this system as $M_0 + \lambda v_{null}$ because

$$B(M_0 + \lambda v_{null}) = BM_0 + \lambda B v_{null} = b . \tag{3.32}$$

By plugging this back into (3.13) for a single frame, we get a quadratic in λ. This can be solved to find up to two 2 length solutions.

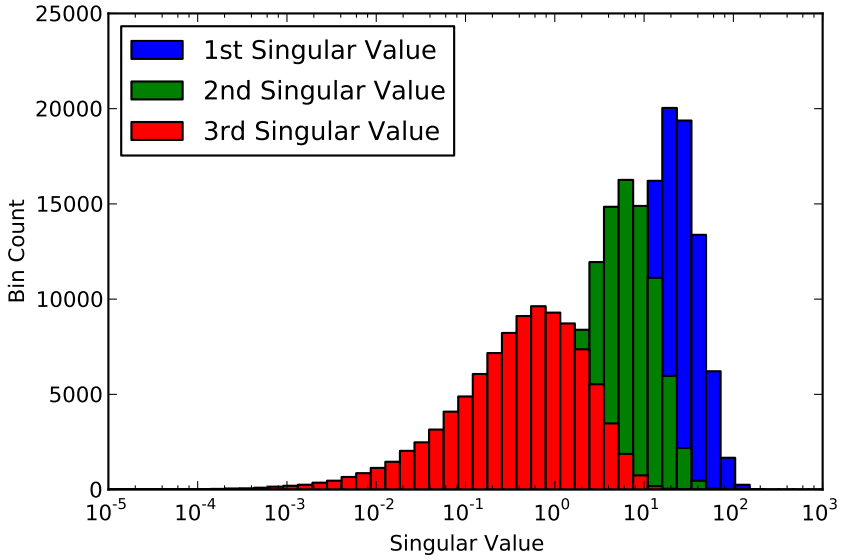

Figure 3.5: Estimated distributions of the three singular values of the linear squared length recovery system B where 3 random points are projected randomly projected to 4 views 10000 times.

3.2.2 Exterior Orientation

The above procedure only recovers the structure of the triangle, leaving its per-frame pose, or exterior orientation, to be estimated. Note that we can only hope to recover the pose up to a depth flip and translation in each frame as it was demonstrated that these transformations will not alter the orthographic image projections. In the noiseless case, this can be done by pinning the x-y components of points p_1, p_2, p_3 to the rays that pass through the image projections w_1', w_2', w_3' as illustrated in Figure 3.4 and solving for the depths z_1, z_2, z_3 of these points. An arbitrary depth translation is fixed by selecting $z_1 = 0$ and the depth flip by setting $z_2 = \sqrt{L_1^2 - l_1^2}$. The first two points constrain the last by $z_3 = \pm\sqrt{L_3^2 - l_3^2}$ and $z_3 = z_2 \pm \sqrt{L_2^2 - l_2^2}$ for which one of these values will exactly satisfy both constraints. This can be done in each frame to extract candidate camera points $p_{f1}, p_{f2}, p_{f3} \in \mathbb{R}^3$ and a rigid alignment $(\mathcal{R}_f, t_f)$ found such that $p_{fi} = \mathcal{R}_f s_i(L) + t_f$. Combined with the linear extraction of the link lengths $L \in \mathbb{R}^3$ using (3.13), this gives an exact solution to the three point rigid structure from motion problem (up to the aforementioned ambiguities) in the noiseless case.

3.2.3 On Counting Rigid Interpretations

We have shown here, that for each link length solution $L \in \mathcal{T}_L$ there is a unique triangle specified by $\gamma \in \mathcal{T}_\gamma$. This triangle admits 2^F possible depth flips that produce the same image projections. Thus, for $F = 4$ frames, we have shown how to recover a single link length solution yielding $2^4 = 16$ possible rigid interpretations. For $F = 3$ frames, we have shown how to recover up to two possible link length solutions yielding up to $2 \cdot 2^3 = 16$ possible rigid interpretations. In contrast, the work [7] shows that there are at most 32 different rigid interpretations for $F = 4$ frames, and the work [24] shows that there are up to two distinct link length solutions that yield up to $2 \cdot 2^3 = 32$ different rigid interpretations for $F = 3$ frames. We suggest that these works are likely counting, for each length solution, the 2^F depth flip ambiguities that result from each of the two triangles that can be specified using $\gamma^\pm \in \mathcal{T}_\gamma^\pm$ (see Section 3.1.1). That is, that they are allowing a triangle model in such a rigid interpretation to have a "visible" face based on the sign of $\gamma_3^\pm$, accounting for the factor of two discrepancy. Regardless of how one counts rigid interpretations though, for $F = 4$ frames, we have shown that all rigid interpretations correspond to a single length solution. For our purposes, however, we do not differentiate between the sides of the triangle and thus only admit the 2^F interpretations corresponding to F possible depth flips.

3.3 Plausibly Rigid with Gaussian Noise Model

In practice, we would like an algorithm that can cope with multiple noisy frames. In this section we consider two such algorithms.

3.3.1 3-SFM (Linear)

One approach, is to simply substitute the squared *observed* link lengths for the squared *projected* link lengths when forming the linear system in (3.28) and solve for M using linear least squares. If M has only positive components we can take the element-wise square-root to recover a vector $L \in \mathbb{R}^3$. If M has negative components or the estimated lengths do not form a valid triangle (i.e., $L \notin \mathcal{T}_L$), we fall back to a more naive estimate by using the observed image link lengths l_{f^*} in the frame

$$f^* = \underset{f}{\operatorname{argmax}} \|l_f\|_1 \tag{3.33}$$

where the observed triangle has the largest perimeter. Note that this will actually be a high quality estimate if the triangle is ever seen nearly "head on". Further, it can be motivated by the tendency for humans to perceive the maximal extension between two imaged points as a frontal-parallel orientation [31]. Regardless, the practical advantage is simply that it forces this procedure, 3-SFM (Linear), to always recover lengths L that form a valid triangle.

Naturally, the noise free algorithm for solving exterior orientation cannot be applied in the presence of noise. We therefore consider solving for an orientation that minimizes squared reprojection error. For this we assume that the rotations are parameterized by some standard vector ω_f of rotational parameters so that $R_f = R(\omega_f)$. We then formulate the minimization of the squared reprojection error under this model with a fixed rigid triangle by seeking this rotational parameter in addition to the necessary 2D translation h_f.

$$\min_{h_f,\omega_f} \sum_{n=1}^{3} \|R(\omega_f)s_n + h_f - w'_{fn}\|^2 \tag{3.34}$$

Note that if ω_f is fixed, we can use the well known result [32] that the optimal translation between the projected scene points and observed points is the difference in centroids. Therefore, given ω_f, the optimal translation is

$$h_f = \bar{w}'_f - \frac{1}{3}\sum_{n=1}^{3} R(\omega_f)s_n \tag{3.35}$$

$$= \bar{w}'_f - R(\omega_f)\frac{1}{3}\sum_{n=1}^{3} s_n \tag{3.36}$$

$$= \bar{w}'_f - R(\omega_f)\bar{s} \tag{3.37}$$

where $\bar{w}'_f$ is the centroid of the observations and $\bar{s}$ is the centroid of the scene points. This allows us to rewrite (3.34) using the centred scene coordinates $\hat{s}_n = s_n - \bar{s}$ and centred image coordinates $\hat{w}'_{fn} = w'_{fn} - \bar{w}'$.

$$= \min_{\omega_f} \min_{h_f} \sum_{n=1}^{3} \|R(\omega_f)s_n + h_f - w'_{fn}\|^2 \tag{3.38}$$

$$= \min_{\omega_f} \sum_{n=1}^{3} \|R(\omega_f)s_n + \left(\bar{w}'_f - R(\omega_f)\bar{s}\right) - w'_{fn}\|^2 \tag{3.39}$$

$$= \min_{\omega_f} \sum_{n=1}^{3} \|R(\omega_f)\hat{s}_n - \hat{w}'_{fn}\|^2 \tag{3.40}$$

and thus the problem is reduced to estimating the 3 rotational degrees of freedom inherent in ω_f. In practice, this can be done reliably by using a non-linear optimizer such as L-BFGS [12] using a few random restarts. When the views correspond to a temporally smooth sequence as is often the case, the solution in a neighbouring frame can provide an additional high quality initial guess. Finally one can extract the optimal translation using (3.37).

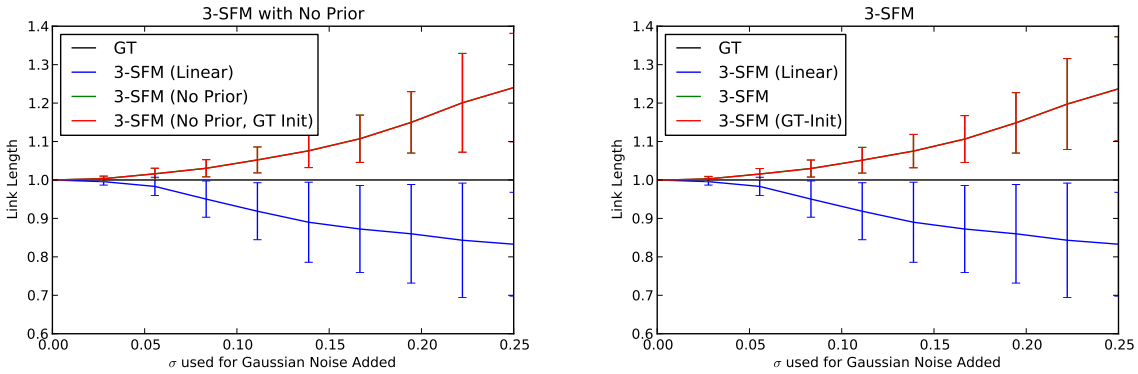

Figure 3.6: Accuracy of 3-SFM variants in recovering link lengths averaged over 50 runs. An equilateral triangle with link lengths 1 is projected randomly in 100 frames and $N(0, \sigma^2)$ noise added to each projection. Note that the lines for 3-SFM and 3-SFM (GT-Init) nearly coincide, and thus 3-SFM does a good job of finding the same local (perhaps global) optimum as found initializing from ground truth. Further, 3-SFM is much more robust to (Gaussian) noise than the linear solution at reasonable noise levels. Notice also, that there is virtually no difference between using the prior or not, as there is no viewpoint degeneracy.

3.3.2 Bundle Adjustment

In the presence of noise or violations of the fundamental rigidity assumption, solving for the lengths in (3.28) above only minimizes algebraic error and produces biased length estimates. This phenomenon can be seen in Figure 3.6, where various amounts of Gaussian noise are added to random views of an equilateral triangle in 100 random views. Naturally, with no noise, 3-SFM (Linear) recovers the correct solution but in the presence of noise, the lengths are underestimated. Therefore, that solution is used to initialize a non-linear refinement of both lengths and per frame pose jointly to minimize squared reprojection error across frames. Of course, we cannot always expect to have such ideal conditions: diverse viewpoints, Gaussian noise and equilateral geometry and thus we admit the possibility of adding a regularizer to discourage unlikely triangle structures.

This is formulated in the following bundle adjustment like energy.

$$\min_{\gamma} \left(\sum_{f=1}^{F} \min_{h_f, \omega_f} \sum_{n=1}^{3} \|R(\omega_f) s_n(\gamma) + h_f - w'_{fn}\|^2 + \lambda_{prior} \sum_{n=1}^{3} L_n(\gamma)^2 \right) \tag{3.41}$$

$$= \min_{\gamma} \left(\sum_{f=1}^{F} \min_{\omega_f} \sum_{n=1}^{3} \|R(\omega_f) \hat{s}_n(\gamma) - \hat{w}'_{fn}\|^2 + \lambda_{prior} \sum_{n=1}^{3} L_n(\gamma)^2 \right) \tag{3.42}$$

where the second line again follows from centring the data as was done in the previous section. Solving this corresponds to finding the MAP solution under a Gaussian noise model with a Gaussian prior placed on the link lengths. In the sequences that we consider in this thesis, the link lengths generally lie in a similar range from 1 to 100. We therefore find it unnecessary to tune this prior to each triangle, and simply use $\lambda_{prior} = 0.01$ unless otherwise stated.

In summary, our three point structure from motion algorithm, 3-SFM, simply uses the initial guess from Section 3.3.1 to initialize a local gradient based optimization of (3.41). [2] Looking back to Figure 3.6, we can see that 3-SFM provides a much more reasonable length recovery in the presence of significantly more noise. Further, it appears that initializing this bundle adjustment from ground truth does not cause a significant change in the recovered link lengths, and thus it appears that the local minima that it finds is very close to the global optimum.

3.3.3 The Effect of the Prior

When there are enough generic viewpoints, the image observations will quickly drown out the effect of the prior. In contrast, the prior allows the algorithm to fail gracefully by choosing a "reasonable solution" when the structure is left otherwise under-constrained by the provided viewpoints. Without the prior, it is unclear how factors, such as imaging noise or non-rigidity in the underlying point configuration, would combine to dictate the minimum of the energy. This will then result in the algorithm "hallucinating" non-existent triangle structure. One example of such a situation, is when a triangle spins around a fronto-parallel axis that remains in the plane of the triangle. As we have chosen the world coordinate system so that the triangle lies in the x-y plane, this corresponds to any axis also in that plane (as in Figure 3.1). In the noiseless case, this degeneracy is demonstrated by the following result.

[2] One has to be slightly careful to avoid saddle points that can occur if the triangle becomes fronto-parallel in a frame. In practice, this is not difficult to avoid, for example by slightly perturbing such rotations and restarting the optimization until no progress is made.

Result 3. *Let $s_1, s_2, s_3 \in \mathbb{R}^3$ define a rigid triangle lying in the x-y plane and let $d \in \mathbb{R}^3$ be any unit vector also in the x-y plane and $n = d \times e_3$ be a unit vector in the x-y plane normal to d. Let $\mathcal{R}$ be a rotation of $\theta \in \mathbb{R}$ around d. Then for any $\alpha \geq 0$, if $s'_n(\alpha) = (s_n^T d)d + (1+\alpha)(s_n^T n)n$ and $\mathcal{R}'(\alpha)$ is a rotation of $\theta'(\alpha) = \cos^{-1}(\cos(\theta)/(1+\alpha))$ around d, then we have that*

$$\Pi\mathcal{R}'(\alpha)s'_n(\alpha) = \Pi\mathcal{R}s_n \ . \tag{3.43}$$

That is, there is a continuous ambiguity specified by a relation between the triangle structure and its rotation.

Proof. To prove this, we will expand this equation, taking advantage of the fact that the orthographic projection Π will drop any z component. We thus do not need to keep track of any z component when $\mathcal{R}'(\alpha)$ rotates n out of the x-y plane.

$$\begin{aligned}
\Pi\mathcal{R}'(\alpha)s'_n(\alpha) &= \Pi(\mathcal{R}'(\alpha)(s_n^T d)d + \mathcal{R}'(\alpha)(1+\alpha)(s_n^T n)n) && (3.44)\\
&= \Pi((s_n^T d)d + \cos(\theta'(\alpha))(1+\alpha)(s_n^T n)n + \sin(\theta'(\alpha))(1+\alpha)(s_n^T n)e_3) && (3.45)\\
&= \Pi((s_n^T d)d + \cos(\theta'(\alpha))(1+\alpha)(s_n^T n)n) && (3.46)\\
&= \Pi((s_n^T d)d + \frac{\cos(\theta)}{1+\alpha}(1+\alpha)(s_n^T n)n) && (3.47)\\
&= \Pi((s_n^T d)d + \cos(\theta)(s_n^T n)n) && (3.48)\\
&= \Pi((s_n^T d)d + \cos(\theta)(s_n^T n)n + \sin(\theta)(s_n^T n)e_3) && (3.49)\\
&= \Pi\mathcal{R}((s_n^T d)d + (s_n^T n)n) && (3.50)\\
&= \Pi\mathcal{R}s_n && (3.51)
\end{aligned}$$

□

In the presence of imaging noise, or slight deviations from rigidity, this ray of zero energy solutions will become a long valley in the energy landscape. This valley has been observed to slowly descend in the direction of larger triangles as it becomes increasingly easy to model noise through small rotations. In contrast, as the rotational axis begins to deviate from being strictly fronto-parallel, more constraints will arise, pushing the minimum towards the true solution. As cameras and objects in our world tend to stay upright, we should not be surprised to see situations in practical sequences that closely resemble this ambiguity. The result will be that these two forces pushing towards (i.e., non-degenerate views) and away from (i.e., noise in degenerate views) the true solution will play off each other, often leaving the minimum of the energy far from the true

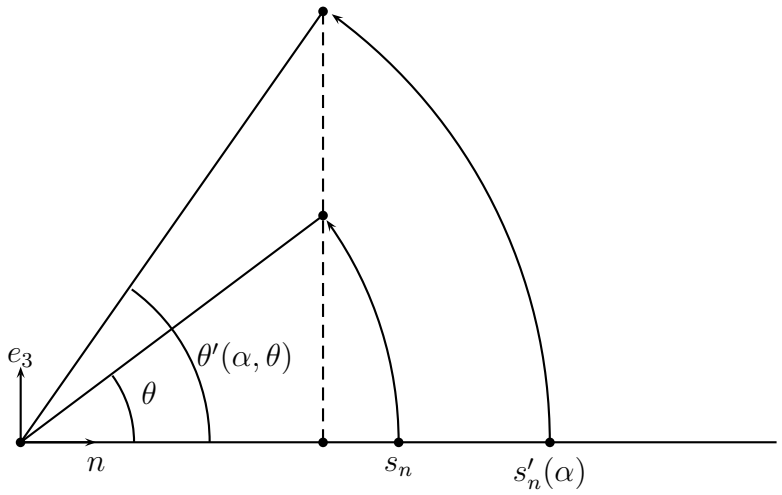

Figure 3.7: Illustration showing that if each scene point s_n is re-parameterized as $s'_n(\alpha)$, then for each rotation $\mathcal{R}$ of θ around d, there is a corresponding rotation $\mathcal{R}'(\alpha, \theta)$ that projects the scene points to the same image locations.

solution. The prior helps by providing a slight tilt to the entire energy landscape, resulting in the minima of these valleys being moved towards parameters that result in smaller and often more reasonable triangles than those that would arise from a minimum based purely on noise. In contrast, point configurations that are well constrained by the observed viewpoints will have sharp minima in the energy. The small tilt that the prior provides will generally not be enough to move these substantially.

To illustrate these effects we examine triangles from the Delaunay triangulation (see Figure 3.8) of the first frame of the `JACKY` sequence [52]. This sequence is particularly useful to illustrate these effects as we have ground truth data available and since the motion consists primarily of rotations around the y-axis. Furthermore, it contains, in addition to a largely rigid face, a variety of non-rigid local 3-point configurations such as those triangles covering the mouth or eyelid.

We first show the effect of the prior in regularizing against the poor variance of viewpoints presented in `JACKY`. This is illustrated in Figure 3.9 where it can be seen that a good portion of the link lengths recovered are massively overestimated. For example, without the prior, 3-SFM will fit an extremely long and obtuse rigid triangle model to the image observations of triangle 33 (highlighted in the Figure). This is done by choosing rotations so that the triangle extends deep into depth and using the remaining rotational freedom to minimize the squared image residuals. This behaviour can be seen in Figure 3.10, where the link lengths diverge while very little improvement in the energy function is obtained. This indicates that the energy is following the nearly flat bottom of a long valley, which corresponds to the lack of constraints. In contrast, 3-SFM is

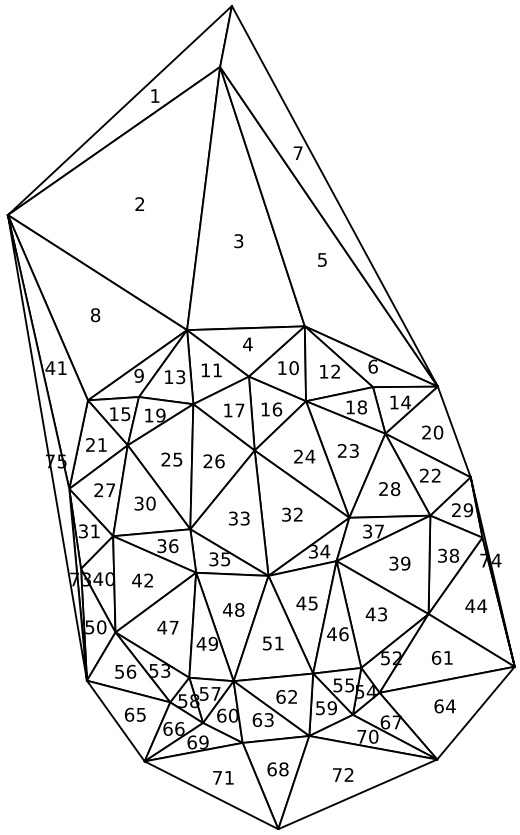

Figure 3.8: Delaunay triangulation of Jacky sequence in first frame.

able to recover "reasonable" link lengths using the prior term as the quadratic penalty adds a well defined minimum near the beginning of this valley. This also has the clear advantage of avoiding needless exploration, massively accelerating convergence in such under-constrained scenarios (see Figure 3.10).

If we artificially add viewpoint diversity to `JACKY`, naturally the accuracy of both

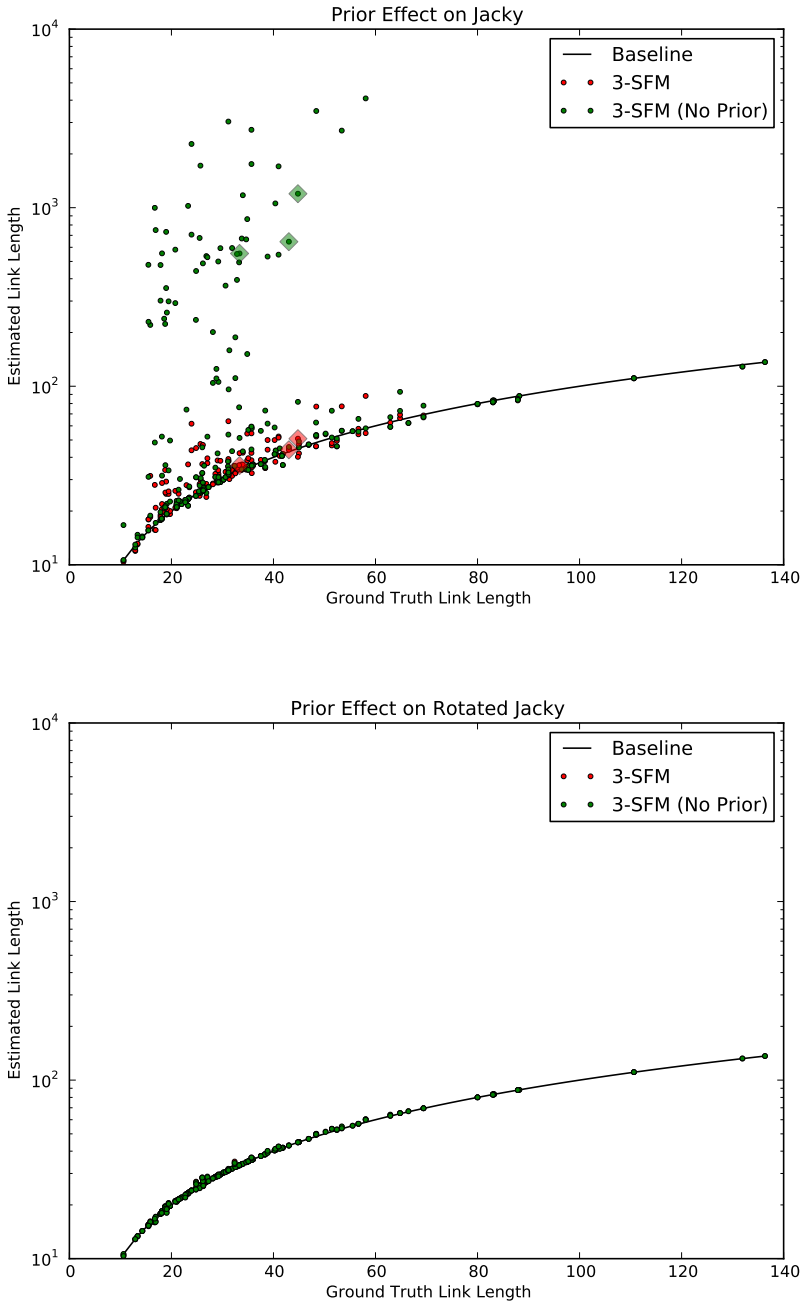

Figure 3.9: The effect of running 3-SFM with and without the prior on the triangles in the Delaunay triangulation in Figure 3.8. Above: The standard Jacky sequence which contains degenerate viewpoints which the prior helps to regularize. Triangle 33 is highlighted as an example. Below: The sequence has each frame randomly rotated to provide generic viewpoints. The prior has little effect and the lengths are properly recovered.

methods drastically improve as can be seen in the bottom of Figure 3.9. Furthermore, there is virtually no difference in the recovered solutions as the prior term is so weak in the presence of the diverse viewpoint constraints.

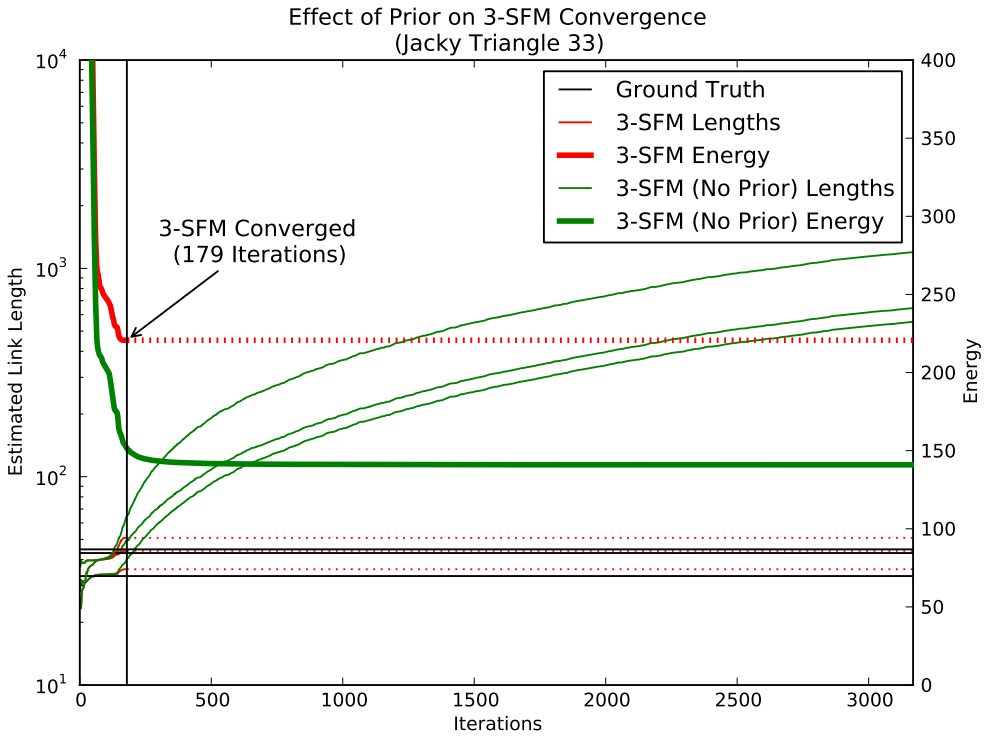

Figure 3.10: The convergence of 3-SFM with and without using the prior on triangle 33 of Jacky. Without the prior, the edge lengths are unconstrained and can be driven to large values to model reprojection error. The prior provides a well defined local minimum at a reasonable solution. This minimum is encountered very quickly.

3.3.4 Inferring Rigidity

One might also wonder whether one can decide whether three points belong to a rigid configuration by examining the residual errors provided by 3-SFM. It was demonstrated by Bennett and Hoffman [7] that in 4 generic viewpoints, the set of rigid models that can *exactly* explain a set of observations is measure zero. If we wish to allow some level of noise or deviance from rigidity, we can use 3-SFM to assign a non-rigidity score based

on RMS reprojection error. This is calculated as

$$\epsilon(\theta) = \sqrt{\frac{1}{3F}\sum_{f=1}^{F}\sum_{i=1}^{3}\|w_{fi}(\theta) - w'_{fi}\|} \tag{3.52}$$

where θ parameterizes the 3-point rigid models recovered from 3-SFM.

To evaluate the usefulness of this score, we again appeal to the `JACKY` sequence where we have ground truth available. We use the matrix of ground truth lengths $\mathbf{L}^{GT}_{F\times 3}$ to form a ground truth *estimate* of non-rigidity by taking the mean of the sample standard deviations of the three lengths.

$$\epsilon^{GT} = \frac{1}{3}\sum_{i=1}^{3}\sqrt{\frac{1}{F}\sum_{f=1}^{F}(\mathbf{L}^{GT}_{fi} - \frac{1}{F}\sum_{f=1}^{F}\mathbf{L}^{GT}_{fi})^2} \tag{3.53}$$

As can be seen in Figure 3.11 there is a correlation between the ground truth non-rigidity estimate and our estimate. Unfortunately, due to the lack of viewpoint variation, some non-rigid structures have a rigid interpretation that yield a relatively low reprojection error (see points in lower right of Figure 3.11). When the sequence is randomly rotated, these phantom interpretations tend to disappear and the correlation becomes much stronger and more useful.

3.4 Conclusion

This chapter began by developing the orthographic three point rigid model to explain the motion of three image trajectories. This model is minimal in the sense that dropping another point (i.e., a two point rigid model) could not be constrained by these trajectories alone and would require additional priors for reconstruction. In contrast, the three point rigid model is well constrained when imaged in a set of generic viewpoints. These constraints, however, admit an interesting per-frame depth flip ambiguity arising from the planar nature of the model. The admittance of an additional point (i.e., a four point rigid model) could potentially break this planarity and resolve the per-frame depth flips. However, in practical sequences, including the majority of the ones considered in this thesis, spatially local quadruplets of trajectories that arise from points in a near rigid configuration will often be near planar. There is therefore little to be gained by considering such an expanded model.

In section 3.2, we explored some options for fitting such a three point rigid model to a

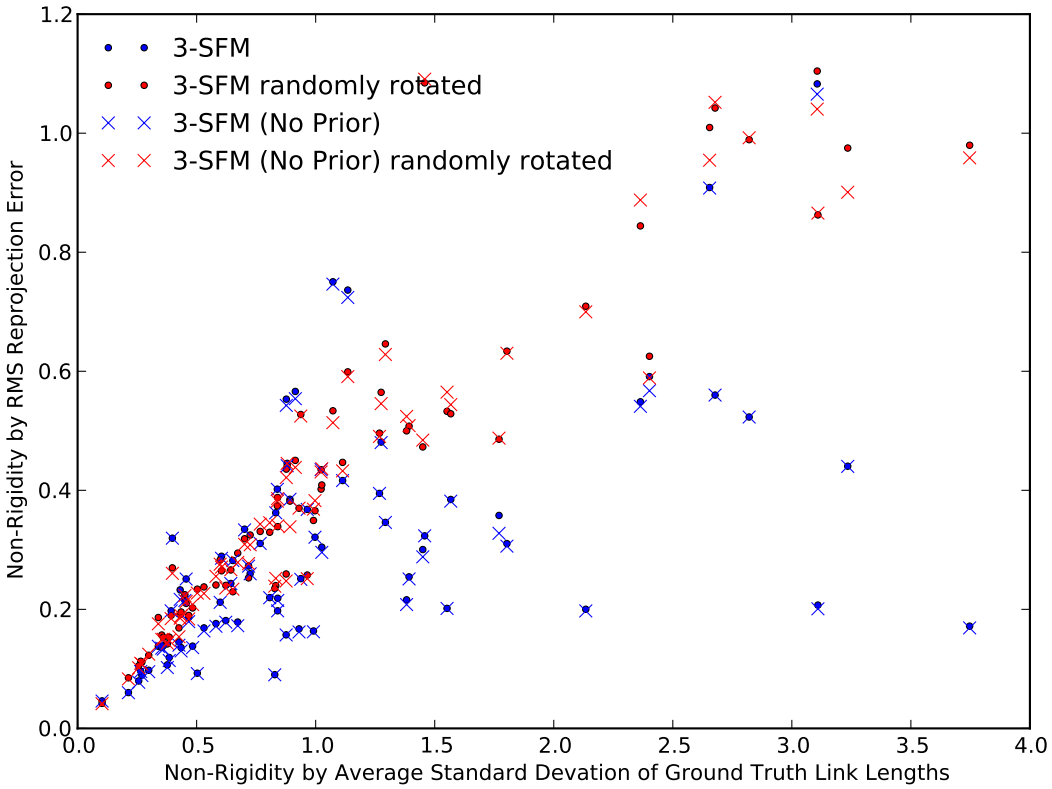

Figure 3.11: Inferred Non-Rigidity Measure vs. Ground Truth Non-Rigidity Measure

set of image trajectories. A key contribution here, is the formulation of a linear constraint on squared link lengths from a set of squared projected link lengths. This provides a direct method to recover the triangle's link lengths when we have access to noise free image projections from just four frames. In the presence of noise, we can incorporate observations from multiple frames using linear least squares. To recover a full solution, the triangle's orientation with respect to the camera (i.e., exterior orientation) in each frame must be recovered, a very manageable non-linear optimization problem. To extract the maximum likelihood estimate under a Gaussian noise model, this solution is then used as an initial guess to a bundle-adjustment like optimization. Although this problem is well-constrained by a generic set of viewpoints, we proceeded to explore some of the degenerate viewpoint configurations that are likely to arise in common sequences. A simple prior penalizing the sum of squared lengths was then formulated and demonstrated to bias the solution towards a smaller fronto-parallel triangle, when such viewpoint degeneracies leave the solution otherwise unconstrained. The resulting procedure, formed from the

initialization followed by the regularized bundle adjustment, defines our three point rigid model recovery algorithm called 3-SFM.

We hypothesized in the introduction, that a procedure like 3-SFM might be a powerful tool in allowing arbitrary scenes to be broken into a set of local rigid models. Indeed, 3-SFM can be used as a probe for local rigidity, as an extracted model with low reprojection error indicates that the motion of the image trajectories have a plausible rigid interpretation. It was further demonstrated that under generic viewpoints we can expect truly rigid configurations to be correctly recovered with low reprojection error, while non-rigid configurations to be identified by their higher reprojection error. The resulting models give cues about the local structure but are valid only up to the per-frame depth flip and translation ambiguity. Resolving these ambiguities and reconstructing a global solution is an interesting problem that is addressed in Chapter 4. In contrast to the more global models (detailed in Chapter 2) that integrate more observations per parameter, one can expect that fitting local models to such a small number of trajectories will be sensitive to image noise. This is addressed in Chapter 5 in which the local models are used only as an initialization for a global energy function. The global structure is then optimized in a large bundle adjustment framework that allows all observations to potentially constrain all points at once.

Chapter 4

Non-Rigid Structure from Locally Rigid Motion

This chapter outlines Locally Rigid Motion (LRM), a framework for solving the non-rigid structure from motion problem. The key idea is that many complex global deformations can be modelled locally by K-point rigid motion. This assumption will become increasingly likely to hold for smaller values of K, as more local configurations can be considered. Naturally then, we would like to use the smallest value for which such models can be fit to, and evaluated for true rigidity against, image observations. We thus explore this concept using the three point rigid models described in the previous chapter. This allows us to model a deforming scene as a "soup" of plausibly-rigid triangle models. This soup must then be integrated into a global solution by resolving each model's orthographic ambiguities. Our approach can be seen as a modern re-interpretation of a scheme advocated by [57] in which four point rigid models were suggested as local models.

Our proposed procedure begins by forming a soup of rigid triangles through a hypothesis and test framework. Triplets of points are proposed as candidates to be tested for rigidity. Such proposals should be as local as possible in order to maximize the chance that the local rigidity assumption holds. Some care should be taken, however, as extreme locality can cause the noise in the observed image trajectories to entirely wash away any identifiable patterns of rigid motion. Regardless, the trajectories from each proposal triplet are then tested for rigidity using 3-SFM and, as a side effect, a rigid triangle model recovered.

Alternatively, if the triplet can be matched to a known template shape, rigidity is evaluated by the proximity of points in the template, which is assumed to be locally rigid. In this case, a rigid triangle model can be extracted directly from the template and fit to a new frame by solving a small exterior orientation problem.

In either case, a soup of plausibly rigid triangle models is recovered from those triplets that passed the test. As discussed in Chapter 3, each local triangle model within this soup has an ambiguous depth flip and depth translation in each frame. The flip ambiguities can be partially resolved by establishing the constraint that two triangle models that share two image points should be flipped in such a manner that the vertices explaining these points can be aligned. If some degree of temporal smoothness can be assumed, then the flips of a given triangle over neighbouring frames might be selected to align the triangle across these frames. We formulate these constraints as a Markov Random Field over binary variables, each of which corresponds to the flip of a triangle in some frame, and encode the described constraints as pairwise potentials. We then consider a variety of techniques to find a low energy state that resolves the per-frame depth flip ambiguities. Each triangle is then translated in depth so as to align its vertices with those of its neighbours. The 3D locations of all vertices that explain an observation are averaged into a single location, thereby providing a final global reconstruction.

4.1 Rigid Triangle Models

In this section we propose to model deforming scenes as a soup of local plausibly rigid triangle models. Given N scene points, we can label such models as triplets of increasing indices. The set of all such triplets is:

$$\mathbb{T}_{\text{all}} = \{(i,j,k) : i,j,k \in 1,..,N, i < j < k\} \ . \tag{4.1}$$

The goal is then to find a subset $\mathbb{T}_{\text{soup}} \subseteq \mathbb{T}_{\text{all}}$, where each triplet represents the indices of observations that can be explained by a rigid triangle model in the soup. For each such triplet τ, we assume that this rigid triangle model has been fit and we label its components with the τ in superscript. We can thus write out this model as an equation that, for each frame $f \in \{1,...,F\}$, explains the observation $w'_{f\tau_i}$ with

$$w^{\tau}_{fi} = \Pi(\mathcal{R}^{\tau}_{f} s^{\tau}_{i} + t^{\tau}_{f}) \tag{4.2}$$

for $i \in \{1,2,3\}$. We first demonstrate how such a soup can be formed by exploiting structure from motion (through 3-SFM) and then briefly consider the alternative of using correspondences to a known template.

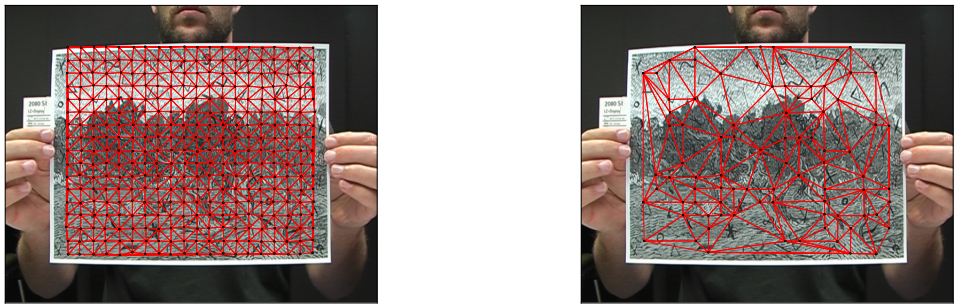

Figure 4.1: Left: Delaunay Triangulation of all N observations in a frame. Notice the long strings of collinear edges due to the grid like structure of the underlying image observations. Right: Delaunay Triangulation of observations with indices in $\mathcal{I}_\zeta$. Notice that the grid-like structure has been mostly broken.

4.1.1 Plausibly Rigid Models from 3-SFM

When we are given $F \geq 4$ frames, we can use the 3-SFM procedure from the previous chapter in a hypothesis and test framework. Triangle models that pass a rigidity test will form our soup.

Proposal Triplets

The first step is to propose a set $\mathbb{T}_{\text{prop}} \subseteq \mathbb{T}_{\text{all}}$ of triplets of indices whose corresponding points will be tested for a valid rigid interpretation. Ideally, we might examine all possible triplets but as $|\mathbb{T}_{\text{all}}| = \binom{N}{3} = O(N^3)$, doing so is prohibitive and therefore some reasonable subset must be selected. However, whichever strategy is employed, it is important that it has a reasonable chance of discovering triplets with nearly rigid point configurations. Our local rigidity assumption hints that we should probe for point configurations that have spatially localized observations in the hopes that they correspond to local nearly rigid structure. Furthermore, it is also desirable to have $\mathbb{T}_{\text{prop}}$ cover the set of image trajectories so that every point has a possibility of being reconstructed.

A natural way to satisfy both of the above criteria, is to triangulate the image observations $\{w'_{f1}, ..., w'_{fN}\}$ of a single frame f as was done in [49] and illustrated in Figure 4.1. We advocate here, however, to augment this set with additional proposals to increase the coverage of the resulting soup. To deal with tracks that get lost, and to some extent occlusion, it is also helpful to include triplets that occur in a triangulation of *any* frame. We further find that it is helpful to break up grid-like structures (such as the one illustrated in the left pane of Figure 4.1) by also triangulating, in each frame, the

image observations corresponding to a random set of point indices $\mathcal{I}_\zeta \subseteq \{1, ..., N\}$ where $\zeta \in [0, 1]$ indicates what fraction of the full N track indices are used. We set $\zeta = 0.25$ by default, which corresponds to roughly downsampling a grid by a single factor of two in each spatial direction as illustrated in the right pane of Figure 4.1.

To be explicit, the set of triplets considered is then

$$\mathbb{T}_{\text{prop}} = \cup_{f=1}^{F} \left(\text{DT}(\{w'_{fn}\}_{n=1}^{N}, \{1, ..., N\}) \cup \text{DT}(\{w'_{fn}\}_{n=1}^{N}, \mathcal{I}_\zeta) \right) \ , \tag{4.3}$$

where $\text{DT}(\{v_m\}_{m=1}^{M}, \mathcal{I})$ takes a set of indexed points $\{v_m\}_{m=1}^{M} \subseteq \mathbb{R}^2$ and an index set $\mathcal{I} \subseteq \{1, ..., M\}$ and returns the set of triplets of increasing indices $\{(i, j, k) \in \mathcal{I}^3 : i < j < k\}$ corresponding to the Delaunay Triangulation [17] of the points $\{v_m : m \in \mathcal{I}\}$.

Fitting and Testing Triangle Triplets using 3-SFM

The next step is to fit a three point rigid model to the corresponding observations of each triplet $\tau \in \mathbb{T}_{\text{prop}}$ using 3-SFM. That is, for each frame f and each vertex i in the rigid triangle model, the observation $w'_{f\tau_i}$ is modelled by

$$w^\tau_{fi} = \Pi(\mathcal{R}^\tau_f s^\tau_i + t^\tau_f) \ . \tag{4.4}$$

We evaluate this model using two tests. The first test filters out any models that are determined not to correspond to valid rigid motion on the basis of reprojection error. As models that undergo generic non-rigid motion are not expected [7] to have a valid rigid interpretation, we assign RMS reprojection error

$$\epsilon^\tau = \sqrt{\frac{1}{3F} \sum_{i=1}^{3} \|w'_{f\tau_i} - w^\tau_{fi}\|^2} \tag{4.5}$$

as a measure of non-rigidity to each $\tau \in \mathbb{T}_{\text{prop}}$. Models for which this measure is above a certain threshold ϵ^* are considered to be non-rigid and discarded. This set is

$$\mathbb{T}_{\text{non-rigid}} = \{\tau \in \mathbb{T}_{\text{prop}} : \epsilon^\tau > \epsilon^*\} \ . \tag{4.6}$$

Models below this threshold are assumed to be rigid but may actually be under-constrained due to degenerate triangle structure or from a lack of generic motion as discussed in section 3.3.4. A useful heuristic for setting this parameter automatically from the data is

to set it to be $\epsilon^* = \eta\tilde{\epsilon}$, where $\tilde{\epsilon}$ is the median of the set

$$\{\epsilon^\tau : \tau \in \mathbb{T}_{\text{prop}}\} \ . \tag{4.7}$$

We generally set η to 1.5 unless otherwise indicated, which helps to ensure a large coverage of the trajectory set, while still filtering large outliers.

It was noticed in [49] that some of the more extreme degenerate triangles, that slip through this filter have a characteristic geometry. Such triangle models typically are very thin and become turned deeply into depth to model the image observations and noise using foreshortening. This motivates our second test that employs a simple heuristic for identifying these false positives. This heuristic looks at each triangle $\tau \in \mathbb{T}_{\text{prop}}$ and finds its smallest angle

$$\kappa^\tau = \min_i \angle(s^\tau_{f(i \bmod 3+1)} - s^\tau_{fi}, s^\tau_{f((i+1) \bmod 3+1)} - s^\tau_{fi}) \ , \tag{4.8}$$

where

$$\angle(v_1, v_2) = cos^{-1}\left(\frac{v_1^T v_2}{\|v_1\|\|v_2\|}\right) \in [0, 180] \tag{4.9}$$

is the angle in degrees between v_1 and v_2. We then discard triangles whose smallest angle is less than some parameter κ^*, which we set to 20 degrees. This set of discarded triplets is

$$\mathbb{T}_{\text{degenerate}} = \{\tau \in \mathbb{T}_{\text{prop}} : \kappa^\tau < \kappa^*\} \ . \tag{4.10}$$

This leaves us with our final "soup" of plausibly rigid triangles

$$\mathbb{T}_{\text{soup}} = \mathbb{T}_{\text{prop}} - (\mathbb{T}_{\text{non-rigid}} \cup \mathbb{T}_{\text{degenerate}}) \ . \tag{4.11}$$

4.1.2 Plausibly Rigid Models from a Planar Template

In this section, we briefly consider an alternative scenario in which we only attempt to reconstruct a single frame (i.e., $F = 1$). Naturally, we cannot use a structure from motion algorithm like 3-SFM, as the problem would be massively under-constrained. We therefore also assume that we are provided a single template frame, in which each observation w'_{1n} has a match $\omega_n \in R^2$ in this frame. We further assume that we have a distance matrix $D_{N\times N}$ in which D_{nm} is the true 3D distance between the n'th and m'th scene points in that template frame. We generally assume that the underlying structure is nearly planar and that we have a head on view (as illustrated in Figure 4.1), and thus $D_{nm} \approx \|\omega_n - \omega_m\|$, but we do not explicitly require this. This template, allows us to

compute the link lengths of our triangle models directly from the template instead of relying on 3-SFM. We now show how to modify the hypothesis and test framework to form a soup using this template.

Proposal Triplets

If we assume that the template is locally rigid, then for points i and j on the model we can use the distance D_{ij} as a proxy measure for rigidity. We thus, again, use a Delaunay triangulation of the template points to find triplets of points that are likely to be nearly rigid. We foreshadow that the use of a template will allow us to fit triangle models of much higher quality and thus do not deem it necessary to integrate further proposal triplets. Furthermore, for reasons that will become clear in Section 4.2.1, this will actually be a significant advantage when integrating our local models into a global model. To be explicit, the set of triplets that we consider is just

$$\mathbb{T}_{\text{prop}} = \text{DT}(\{\omega_n\}_{n=1}^{N}, \{1, ..., N\}) \ . \tag{4.12}$$

Fitting and Testing Triangle Triplets using a Template

For each triangle $\tau \in \mathbb{T}_{\text{prop}}$, we can extract a link length solution directly from the template using $L_i^\tau = D_{\tau_i \tau_{i \bmod 3+1}}$ for $i \in \{1, 2, 3\}$. We hold $L^\tau \in \mathbb{R}^3$ fixed and follow the procedure in Section 3.3.1 to find the rigid transformation $(\mathcal{R}_f^\tau, t_f^\tau)$ that best models the corresponding observations under a Gaussian noise model. This gives us a local model

$$w_{fi}^\tau = \Pi(\mathcal{R}_f^\tau s_i^\tau + t_f^\tau) \tag{4.13}$$

for $f = 1$ and $i \in \{1, 2, 3\}$. When testing these proposals, we accept their spatial locality in the template as certification of rigidity, and thus do not examine reprojection error. We do, however, allow thin near co-linear triangles from the triangulation to be filtered out as before. This leaves us with our final "soup" of plausibly rigid triangles as

$$\mathbb{T}_{\text{soup}} = \mathbb{T}_{\text{prop}} - \mathbb{T}_{\text{degenerate}} \ . \tag{4.14}$$

4.2 Non-Rigid Structure from Locally Rigid Motion

At this point, we assume that we have formed a soup $\mathbb{T}_{\text{soup}}$ of plausibly rigid triangle models to model the scene. These $T = |\mathbb{T}_{\text{soup}}|$ rigid triangle models each have a depth flip and depth translation ambiguity in each frame which we will resolve in this section.

The latter can be resolved easily if the depth flips are known in each frame, as we will see in Section 4.2.2, and thus we now concentrate first on resolving the depth flips.

4.2.1 Resolving Depth Flips

We attack this problem of selecting one of the 2^{FT} depth flip configurations by formulating a pairwise Markov Random field. This field assigns an energy $F(y)$ to the vector $y \in \{0,1\}^{FT}$ of binary depth flip variables. Orthographic projection relays no information about the flip of a single triangle model in a single frame and thus there are no unary potentials. Instead, we rely solely on pairwise potentials that encode the interactions between the components of y. For simplicity, we define $\psi_{f\tau}$ to assign triangle τ in frame f a unique index in the FT length flip vector y. Thus we can define $y_{\psi_{f\tau}} = 0$ to indicate that triangle τ has taken the arbitrary flip in frame f output by 3-SFM and $y_{\psi_{f\tau}} = 1$ to indicate instead a flip through the $x-y$ plane. Further, we can define the pairwise potential $F_{\psi_{f\tau},\psi_{f'\tau'}}(y_{\psi_{f\tau}}, y_{\psi_{f'\tau'}})$ as the energy associated with its interaction with triangle model τ' in frame f'. We use these potentials to define the energy as

$$F(y) = \sum_{(i,j)\in\mathcal{G}} F_{ij}(y_i, y_j) \ . \tag{4.15}$$

where $\mathcal{G} \subseteq \{(i,j) : i,j \in \{1,...,FT\}; j > i\}$ is a set of pairwise interactions between indices in the flip vector.

Pairwise Spatial Potentials

Let τ denote a triangle with vertices i and j modelling points n and m (i.e., $\tau_i = n$ and $\tau_j = m$). Recall that p^{τ}_{fj} and p^{τ}_{fi} are the 3D positions of these vertices in camera coordinates. We can then define a directional vector along this edge, that depends on a flip variable $b \in \{0,1\}$ indicating whether this triangle has flipped or not as

$$d^{\tau}_{f,i,j}(b) = B(b)(p^{\tau}_{fj} - p^{\tau}_{fi}) \tag{4.16}$$

where

$$B(b) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & (-1)^b \end{bmatrix} \ . \tag{4.17}$$

That is, $B(b)$ is either the identity or a reflection through the x-y plane depending on its argument b. Let τ' denote another triangle with vertices i' and j' that also model

points n and m (i.e., $\tau'_{i'} = n$ and $\tau'_{j'} = m$). In any frame f, let $k = \psi_{f\tau}$ and $k' = \psi_{f\tau'}$ denote their indices in the flip vector y. Their flips y_k and $y_{k'}$ should be constrained to maximize some measure of the possible alignment of their corresponding vertices. We therefore ensure $(k, k') \in \mathcal{G}$ and encode a corresponding potential that measures some function of the angle between their corresponding edges as:

$$F_{kk'}(y_k, y_{k'}) = \Upsilon_s(\angle(d^{\tau}_{f,i,j}(y_k), d^{\tau'}_{f,i',j'}(y'_k); \sigma_s) \tag{4.18}$$

where

$$\Upsilon_s(\theta; \sigma_s) = \frac{\theta^2}{\theta^2 + \sigma_s^2} \tag{4.19}$$

is the Geman-McClure [8] error function with parameter σ_s that is set to 10 degrees unless otherwise indicated. This function is close to quadratic near zero but inflects as it asymptotes to 1 as can be seen in Figure 4.3. This mapping is admittedly heuristic, but reflects the intuition that correct local models will have one alignment error near 0 and thus a harsh penalty should be imposed if the other is chosen in a flip configuration. In contrast, if one of the triangles is a false positive, this pair will not likely see a low error (e.g. less than 20 degrees) under either flip arrangement and thus each will be assigned similar potential values downweighting the effect of this interaction in the resulting energy. Further note that when the edges aligning these triangle models are roughly fronto-parallel, both alignment errors will be close to zero, leaving a potential that provides little information to inference methods.

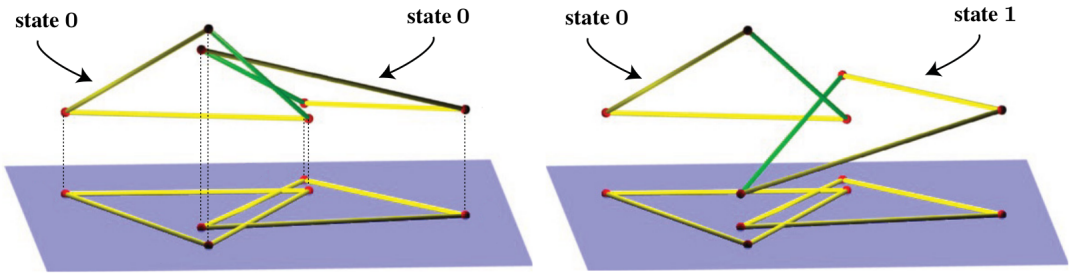

Figure 4.2: Two neighbouring three point rigid models that share a common edge are shown orthographically projected onto an image plane (blue) under the two possible depth flip assignments. Of the four possible depth flip assignments only two are shown corresponding to the even flip interaction (left) and the odd flip interaction (right). The other two assignments correspond to depth flipping both triangles in each of these two cases.

Pairwise Temporal Potentials

In the typical sequences that we consider, some degree of temporal continuity can be assumed. We thus consider how the pose of a triangle $\tau \in \mathbb{T}_{\text{soup}}$ changes between two consecutive frames. Given such a triangle, we use the ordering of the triangle vertices to define a vector normal to the surface of the triangle in frame f under flip $b \in \{0, 1\}$ as

$$n_f^\tau(b) = \frac{d_{f,1,2}^\tau(b) \times d_{f,1,3}^\tau(b)}{\|d_{f,1,2}^\tau(b) \times d_{f,1,3}^\tau(b)\|} \tag{4.20}$$

$$= \frac{(-1)^b B(b)(d_{f,1,2}^\tau(0) \times d_{f,1,3}^\tau(0))}{\|(-1)^b B(b)(d_{f,1,2}^\tau(0) \times d_{f,1,3}^\tau(0))\|} \tag{4.21}$$

$$= \frac{(-1)^b B(b)(d_{f,1,2}^\tau(0) \times d_{f,1,3}^\tau(0))}{\|d_{f,1,2}^\tau(0) \times d_{f,1,3}^\tau(0)\|} \tag{4.22}$$

In the next frame, $f' = f + 1$, we should not expect the normal $n_{f'}^\tau(b)$ to be radically different, and thus we constrain them to be similar. This is done by ensuring that $(k, k') \in \mathcal{G}$ where $k = \psi_{f\tau}$ and $k' = \psi_{f'\tau}$ and defining the corresponding potential on this edge as

$$F_{kk'}(y_k, y_{k'}) = \Upsilon_t(\angle(n_f^\tau(y_k), n_{f'}^\tau(y_{k'}); c_t) \tag{4.23}$$

where

$$\Upsilon_t(\theta; c_t) = c_t\theta \tag{4.24}$$

scales the alignment error using c_t to account for the degree of trust in temporal continuity. In practice, for typical sequences we will trust temporal continuity much more than spatial coherence. Even when we have a very poorly fit triangle model, we do not expect its pose to drastically change from frame to frame. The default value of $c_t = \frac{1}{50} deg^{-1}$ shown in Figure 4.3 allows spatial coherence to only dominate temporal continuity in the energy when very good spatial alignments can be found.

Now that the MRF is well defined, we explore options for inferring a flip vector y so that its energy $F(y)$ is as low as possible. That said, one also needs to be realistic about the value one expects to extract from a low energy labelling. The heuristic nature in which the MRF is constructed means that the correlation between lower energy values and higher quality reconstructions will almost necessarily be weak. One can, of course, try to tweak the MRF so as to produce better labellings, but without a large quantity of realistic ground truth data to use as a validation set we will simply overfit the sequences that we do have. Such tweaking was, therefore, not considered and instead we have deferred to intuition to provide some plausible pairwise potentials, while maintaining a

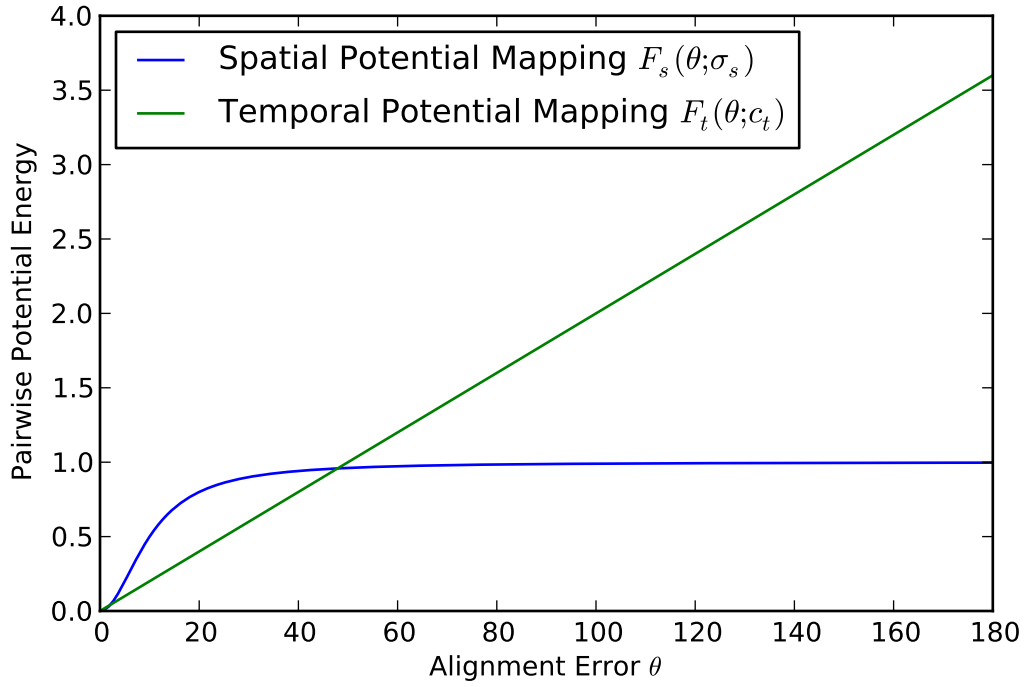

Figure 4.3: Comparison of mapping from spatial and temporal alignment errors using the default parameters mentioned in the text.

few parameters that can be set when a new dataset is encountered.

Inference Options

One thing to notice is that the MRF has some special characteristics. In addition to not having any unary potentials, the pairwise potentials are symmetric around an even and an odd depth flip configuration. That is, any assignment of flips to a pair of triangle models with $(i,j) \in \mathcal{G}$ will have their pairwise potential contribute the same energy if both models are again flipped (i.e., $F(y_i, y_j) = F(1-y_i, 1-y_j)$). To see why this is, let $A(b) = I[b=0]I_{3\times 3} + I[b=1]\mathcal{A}_{3\times 3}$ where $\mathcal{A}$ is any symmetric orthonormal matrix, then

$$\angle(A(a)u, A(b)v) = \cos^{-1}\left(\frac{(A(a)u)^T A(b)v}{\|A(a)u\|\|A(b)v\|}\right) \tag{4.25}$$

$$= \cos^{-1}\left(\frac{u^T A^T(a)A(b)v}{\|u\|\|v\|}\right) \tag{4.26}$$

$$= I[a=b]\cos^{-1}\left(\frac{u^T v}{\|u\|\|v\|}\right) + I[a\neq b]\cos^{-1}\left(\frac{u^T \mathcal{A} v}{\|u\|\|v\|}\right) \tag{4.27}$$

Note that $B(b)$ in (4.16) and $(-1)^b B(b)$ in (4.22) are such matrices and thus, applying this result to the spatial potential in (4.18) and (4.23) allows us to define for any $(i,j) \in \mathcal{G}$,

$$F_{ij}^{EQ} = F_{ij}(0,0) = F_{ij}(1,1) \tag{4.28}$$

and

$$F_{ij}^{OP} = F_{ij}(0,1) = F_{ij}(1,0) \tag{4.29}$$

and thus rewrite the energy as

$$F(y) = \sum_{(i,j)\in\mathcal{G}} I[y_i = y_j] F_{ij}^{EQ} + I[y_i \neq y_j] F_{ij}^{OP}. \tag{4.30}$$

Note that due to this symmetry, it is easy to see that F will not generally be submodular because $(F_{ij}(1,1)+F_{ij}(0,0))-(F_{ij}(0,1)+F_{ij}(1,0)) = 2(F_{ij}^{EQ}-F_{ij}^{OP})$ can have an arbitrary sign. Further, the graph can contain a frustrated cycle (a cycle of odd length where the sign is negative) and thus an equivalent submodular MRF cannot be constructed [42]. This means that efficient methods such as graph cuts [34] cannot be employed to find the minimum energy labelling y^*. We now, therefore, explore some strategies to perform approximate inference. Note that, without loss of generality, we assume the graph $\mathcal{G}$ is connected, as otherwise, we can simply perform inference independently in each connected component.

Optimal Labelling on any Spanning Tree. One approach is to perform inference in a simpler MRF and then upgrade the solution to the original MRF. If we let $\mathcal{G}' \subseteq \mathcal{G}$ denote some sub-set of the edges in the original MRF, then we can construct a new MRF

$$F_{\mathcal{G}'}(y) = \sum_{(i,j)\in\mathcal{G}'} F_{ij}(y_i, y_j) \tag{4.31}$$

that is restricted to this subgraph. If we can find a good solution to this MRF, then we might hope that it is a good solution to the original energy. The question is then, what kind of subgraphs can we obtain good solutions for?

Due to their widespread tractability in many problems, it was proposed in [49] to look at trees. If $\mathcal{G}'$ is a tree within $\mathcal{G}$ finding a MAP labelling becomes very simple. First, we let the root node take an arbitrary flip, and consider a recursive function that must decide the flip of a node i given the flip of its parent node $\pi(i)$. Due to symmetry, the

optimal choice

$$y_i = I[F^{EQ}_{\pi(i)i} \leq F^{OP}_{\pi(i)i}]y_{\pi(i)} + I[F^{EQ}_{\pi(i)i} > F^{OP}_{\pi(i)i}](1 - y_{\pi(i)}) \tag{4.32}$$

that minimizes the energy contribution from the pairwise potential $F_{\pi(i)i}$ can always be chosen. This means that the contributions of all pairwise potentials from edges in $\mathcal{G}'$ have been minimized and thus y minimizes $F_{\mathcal{G}'}(y)$ optimally[1].

This procedure admits the use of any arbitrary tree $\mathcal{G}' \subseteq \mathcal{G}$, but a principled approach is to assign a weight $w(e)$ to each edge $e \in \mathcal{G}$ and then use Kruskal's algorithm to find a minimum/maximum spanning tree (MST) $\mathcal{G}'$. One could assign w heuristically [49] but a more natural way is to measure the energy that can be guaranteed to be discarded by choosing an edge. That value is

$$w(i,j) = |F^{EQ}_{ij} - F^{OP}_{ij}| \ . \tag{4.33}$$

Note that this approach only uses $FT-1$ constraints provided by the edges of the minimum spanning tree to determine the $FT-1$ relative depth flip variables in y. Therefore, it is likely to provide very brittle solutions as it ignores the rest of the constraints in the MRF. In a uniform triangle mesh, for example, there are roughly $4FT$ inter-triangle interactions providing constraints, three in space and one in time for each triangle. Interestingly, due to symmetry, an incorrect flip will actually cause an entire subtree to change their flips, but another incorrect flip at a node further down will correct the problem for its subtree. Nonetheless, the solutions are likely to be more robust if we can leverage the excluded constraints provided by potentials defined on $\mathcal{G} - \mathcal{G}'$.

Optimal Labelling on any Planar Sub-Graph. To this end, we take another look at rewriting the full energy:

$$\begin{aligned} F(y) &= \sum_{(i,j)\in\mathcal{G}} I[y_i = y_j]F^{EQ}_{ij} + I[y_i \neq y_j]F^{OP}_{ij} - F^{OP}_{ij} + F^{OP}_{ij} && (4.34)\\ &= \sum_{(i,j)\in\mathcal{G}} I[y_i = y_j](F^{EQ}_{ij} - F^{OP}_{ij}) + I[y_i \neq y_j](F^{OP}_{ij} - F^{OP}_{ij}) + F^{OP}_{ij} && (4.35)\\ &= \sum_{(i,j)\in\mathcal{G}} F^{OP}_{ij} + \sum_{(i,j)\in\mathcal{G}} I[y_i = y_j](F^{EQ}_{ij} - F^{OP}_{ij}) && (4.36) \end{aligned}$$

[1]Note that because of orthography, the last remaining variable corresponds to the global depth flip, which can be set arbitrarily

The first term is constant for any configuration y and thus performing inference in this MRF is actually equivalent to performing it in a related Ising MRF in which the constant term is dropped.

$$F_{\text{ising}}(y) = \sum_{(i,j)\in\mathcal{G}} I[y_i = y_j](F_{ij}^{EQ} - F_{ij}^{OP}) \tag{4.37}$$

Recent work [48], provides an optimal labelling for F_{ising} if the underlying graph $\mathcal{G}$ is planar. This is the case, for example, when we are reconstructing a single frame using triangles modelled from a template as described in Section 4.1.2. If $\mathcal{G}$ is not planar, then we could use a planar subgraph $\mathcal{G}' \subseteq \mathcal{G}$ to construct a new MRF for which the optimal solution can be found. In contrast to a spanning tree, a planar subgraph can contain many more edges. Unfortunately, it is not always obvious how to find a good or optimal one as a cheap analogue to a MST algorithm does not exist.

Optimal Labellings using QPBO. The QPBO method [42] allows one to obtain a portion of the full optimal labelling z^* of a binary MRF $E(z)$ with $z \in \{0,1\}^D$. Specifically, their approach defines a function $z = QPBO(E)$ where $z \in \{0,1,\chi\}^D$ and $z_i \in \{0,1\}$ indicates that the optimal label for variable i has been found (i.e., $z_i = z_i^*$) and $z_i = \chi$ indicates that it could not be found. Naturally, one would hope that QPBO would produce an optimal labelling for the flip MRF $F(y)$ but unfortunately it generally leaves the majority of the MRF unlabelled. Experiments [42] show that the number of nodes labelled is tied to the unary strength of the MRF of which $F(y)$ has none[2]. An enhancement called QPBOP that tries more aggressively to label nodes has a very long run time and still fails to label any more than a small fraction of nodes in a reasonable amount of time.

Improving Labellings with QPBOI. All is not lost, however, if we consider methods that use QPBO as a subroutine to improve an existing labelling. These methods generally rely on an operation $z' = FUSE(z, z_{prop}; E)$ called a fusion move that considers updating some labels from z to those of a proposal labelling z_{prop} [42, 35]. The resulting labelling z' is guaranteed to not increase the energy.

This allows the definition of another procedure $z' = QPBOI(z)$ that works by fixing a (random) set of nodes S to take their labels from z, running QPBO on the remaining nodes to obtain a partial solution z_S and using $z' = FUSE(z, z_S)$ operation to improve the solution [42]. This operation's runtime is not prohibitive and allows us to improve the quality of the labellings that we get from other methods.

[2]Note that fixing a strong unary for one flip variable has not been observed to remedy the situation.

Merging Solutions with Fusion Moves. Another option is to use the fusion move to efficiently merge a diverse set of solutions into a strictly better solution [35]. For example, we could generate a proposal solution y_{prop} by performing inference on a randomly selected spanning tree. We can then fuse this to our current estimate to obtain a better solution. We will consider the following methods for acquiring proposal solutions.

- RANDOM: Select a flip vector $y \in \{0,1\}^{TF}$ randomly from a binomial(TF, 0.5) distribution.
- RANDOM MST: Weight each edge (i,j) with a random weight $w_{ij} \sim U(0,1)$ sampled from a uniform distribution. Find a maximum spanning tree $\mathcal{G}'$ using this weighting and find the optimal labelling y restricted to this tree.
- REWEIGHTED MST: Select $\alpha \sim U(0,1)$ and use it to scale down the weights in (4.33) for temporal edges. That is, for each edge $(i,j) = (\psi_{f\tau}, \psi_{f'\tau'}) \in \mathcal{G}$ assign a new weight
$$w'(i,j) = I[f = f']w(i,j) + I[f \neq f']\alpha w(i,j), \tag{4.38}$$
and find a maximum spanning tree. Then find the optimal labelling y restricted to this tree.

Inference Summary. The above discussion indicates that although we are generally unable to directly solve for the minimum flip configuration y^*, there is quite an arsenal of approaches to obtain an approximation. The exact approach is deferred to section 4.3.7 where various strategies are explored and one is selected that strikes a balance between speed and accuracy.

Housekeeping

Now that we have inferred some low energy flip vector $y \in \{0,1\}^{FT}$, we update the global soup of triangle models to incorporate these flips. This corresponds to, for each $\tau \in \mathbb{T}_{\text{soup}}$ and every frame f, replacing the rotation $\mathcal{R}_f^\tau$ with

$$I[y^*_{\psi_{f\tau}} = 0]\mathcal{R}_f^\tau + I[y^*_{\psi_{f\tau}} = 1]\mathcal{B}\mathcal{R}_f^\tau\mathcal{B} \; . \tag{4.39}$$

Also, as the graph $\mathcal{G}$ may not be fully connected we can compute a set of disjoint resolvable segments $\mathbb{T}_1, ..., \mathbb{T}_C \subseteq \mathbb{T}_{\text{soup}}$ such that

$$\mathbb{T}_{\text{soup}} = \cup_{c=1}^{C} \mathbb{T}_c \; . \tag{4.40}$$

Each such segment $\mathbb{T}_c$ can have a global depth flip applied without changing the energy. That is, we can define y' such that

$$y'_{\psi_{f\tau}} = I[\tau \notin \mathbb{T}_c] y_{\psi_{f\tau}} + I[\tau \in \mathbb{T}_c](1 - y_{\psi_{f\tau}}), \tag{4.41}$$

resulting in $F(y') = F(y)$.

4.2.2 Resolving Depths

After flip resolution, within each resolvable component $\mathbb{T}_c$, each triangle model $\tau \in \mathbb{T}_c$ still has an ambiguous depth translation due to orthography in each frame. In a single frame, let z_i^τ denote the depth of vertex i. Let $z^\tau = \frac{1}{3}\sum_{i=1}^{3} z_i^\tau$ be the mean depth of triangle τ.

If vertex i shares a common feature point n with vertex j of another triangle $\tau' \in \mathbb{T}_c$ (i.e., $\tau_i = \tau'_j = n$), then ideally their depths would be equal to allow these vertices to align. That is

$$z_i^\tau - z_j^{\tau'} = 0 \ . \tag{4.42}$$

We know the depth change $\Delta z_i^\tau = z_i^\tau - z^\tau$, so we can rewrite this as.

$$(z^\tau + \Delta z_i^\tau) - (z^{\tau'} + \Delta z_i^{\tau'}) = 0 \tag{4.43}$$

$$z^\tau - z^{\tau'} = \Delta z_i^{\tau'} - \Delta z_i^\tau \tag{4.44}$$

which is a linear constraint on the mean depths of the two triangles. By combining these equations for all such vertex interactions, we form a sparse overdetermined linear system which we solve using least squares to extract the optimal mean depth $\hat{z}^\tau$ for each triangle model $\tau \in \mathbb{T}_c$.[3] Finally, we update each triangle model by replacing the translation vector t_f^τ in the triangle model for that frame by $t_f^\tau + (\hat{z}^\tau - z^\tau)\left[\begin{smallmatrix}0 & 0 & 1\end{smallmatrix}\right]^T$.

4.2.3 Integrating Locally Rigid Triangle Models

The final step is to infer from the set of per-frame rigid 3D triangles, a per-frame position for each point contained in a resolvable segment $\mathbb{T}_c$. This set of such points is

$$P_c = \cup_{\tau \in \mathbb{T}_c} \tau \ , \tag{4.45}$$

[3] We also add an additional constraint that the mean depth be zero (i.e., $\sum_{\tau \in \mathbb{T}_c} z^\tau = 0$) to deal with the global depth ambiguity.

where we are abusing the notational convenience of treating each ordered triplet τ as a set. For each point in this set, we simply take the mean of all vertices associated with that point in each frame. That is, for point $n \in P_c$ in frame f, we write

$$p_{fn}^c = \frac{1}{|V_c(n)|} \sum_{(\tau,i)\in V_c(n)} p_{f\tau_i}^\tau \tag{4.46}$$

where

$$V_c(n) = \{(\tau, i) : \tau \in \mathbb{T}_c, i \in \{1,2,3\}, \tau_i = n\} \ . \tag{4.47}$$

4.3 Experiments with Ground Truth

In order to evaluate our reconstruction algorithm, Locally Rigid Motion (LRM), we utilize sequences in which we have 3D ground truth data available. This data, takes the form of a set of 3D trajectories

$$\{p_{fn}^{gt} : f \in \{1,...,F\}, n = \{1,...,N\}\} \subseteq \mathbb{R}^3 \ . \tag{4.48}$$

As is standard practice, we generate a set of observed image trajectories by simply projecting to the x-y plane via

$$w'_{fn} = \Pi p_{fn}^{gt} \tag{4.49}$$

for each $n \in \{1,...,N\}$ and $f \in \{1,...,F\}$. Given a reconstructed component c we can use the ground truth data to resolve the remaining global depth flip and depth translation ambiguities. That is, in each frame f, we align the points with indices in P_c to the data using

$$\min_{b\in\{0,1\},\delta} \sum_{n\in P_c} \|B(b)p_{fn}^c + \delta e_3 - p_{fn}^{gt}\|^2 \ . \tag{4.50}$$

and updating each point p_{fn}^c with $B(b)p_{fn}^c + \delta e_3$. In particular, we do this for each resolvable component output by LRM. Note that technically speaking, LRM has only a single global depth flip ambiguity due to the use of temporal constraints in the MRF. This per-frame alignment is common, however, as reconstructions that are structurally similar to depth flipped version of ground truth, should still be considered high quality.

When comparing to ground truth, we use RMS 3D error

$$\mathcal{E}(c) = \sqrt{\frac{1}{F|P_c|} \sum_{f=1}^{F} \sum_{n\in P} \|p_{fn}^{gt} - p_{fn}^c\|^2} \tag{4.51}$$

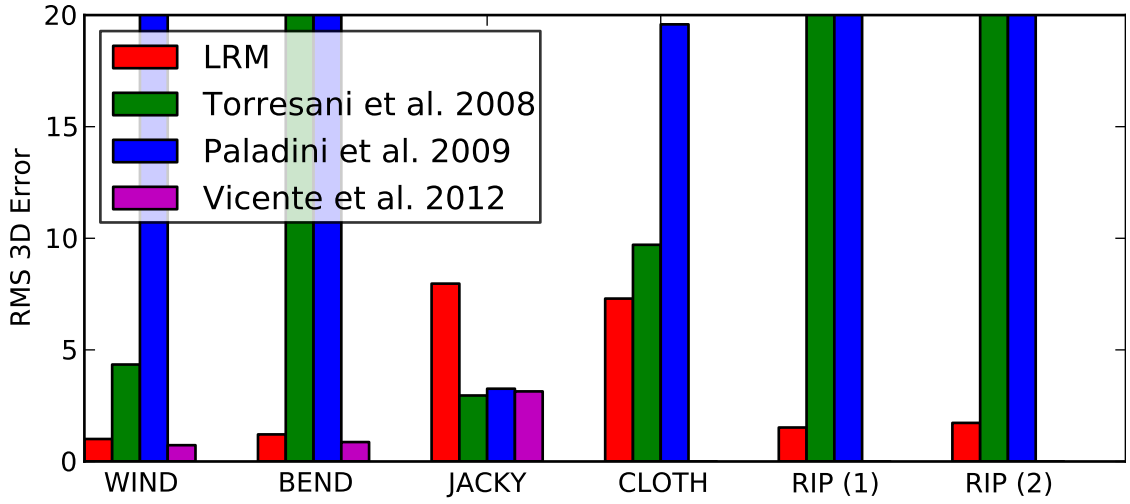

Figure 4.4: Comparison of various methods on ground truth datasets. Results for Vicente *et al.* 2012 are taken from [60], which does not provide values for `CLOTH` or `RIP`

for an aligned component c. For our method, LRM, we calculate this error for the largest resolvable component (i.e., $c = 1$) of each dataset, with the exception of the `RIP` sequence where we separately calculate this error for the two largest components (i.e., $c = 1$ and $c = 2$). This is due to the fact that this sequence naturally contains two distinct deforming bodies, that our algorithm detects. The other algorithms do not perform such a segmentation, so we calculate the error using the single component that they reconstruct (i.e., $P_1 = \{1, ..., N\}$).

We compare to two factorization methods, Torresani *et al.* 2008 [52] and Paladini *et al.* 2009 [38], for which code is available. For these methods, we used the code nearly as provided[4] but attempted to select an optimal number of basis shapes. We also compare to Vicente *et al.* 2012 [60], for which the error defined in (4.51) is available for the `WIND`, `BEND` and `JACKY` sequences. As noted in Section 2.4, this approach also implicitly assumes some degree of local rigidity, and thus we expect similar performance. The approach described in the next chapter, however, shares a similar energy based formulation and thus we defer a more detailed exploration of the contrasts of [60] and our models until Section 5.1.5. The results of this comparison are summarized in Figure 4.4 that compares this error for LRM and these other methods. For now, we introduce the datasets that we consider, and discuss qualitatively the reconstructions.

[4]Note that very minor modifications were made to allow the code to run on all of these datasets.

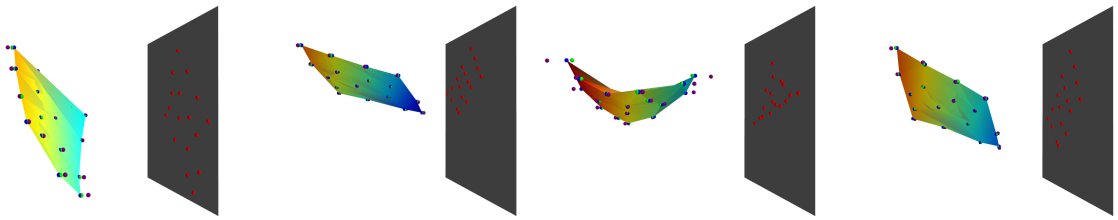

Figure 4.5: 4 frames of WIND with LRM reconstruction (depth colored triangles and blue points), ground truth points (in green), Torresani et al. 2008 (in purple) and image points (in red).

4.3.1 WIND

The result of our algorithm on the WIND dataset is illustrated in Figure 4.5. This dataset consists of 17 trajectories across 1000 frames. The points are located on the a piece of paper that is deforming as it is being blown in the wind. The dataset is an excellent fit for our method as it satisfies the local rigidity assumption and the scene is subjected to wide viewpoint variation. The method qualitatively does an excellent job of recovering the deformation as illustrated in Figure 4.5. It is difficult for the factorization based methods to represent this deformation with a limited number of basis shapes, and the optimization becomes more difficult when this set is expanded. As expected, Vicente *et al.* produce a similar reconstruction error.

4.3.2 JACKY

The result of our algorithm on the JACKY dataset is illustrated in Figure 4.6. This dataset consists of 43 trajectories across 1000 frames. As discussed in Section 3.3.3, this is a very challenging sequence as most of the motion is around a single fronto-parallel axis. This not only makes it difficult to fit accurate rigid triangle models, but it also makes it difficult to evaluate rigidity. Further, there is a noticeable amount of *local* non-rigid deformation as the face deforms even though *globally* the structure is approximately rigid. This is not generally a problem for factorization methods that can model such local deformations and further benefit from the fact that they are initialized from a rigid factorization. In contrast, the local rigidity assumption of our model is repeatedly violated and the test for these violations is rendered impotent due to the viewpoint degeneracy. One can see, for example, that the algorithm has hallucinated rigid triangle models that fit in between the lips, and this causes the chin to move back and forth in depth while the actor's mouth is opening and closing. Nonetheless, we do manage to recover a plausible reconstruction, as the prior allows us to fit reasonable approximate triangle models. In contrast to

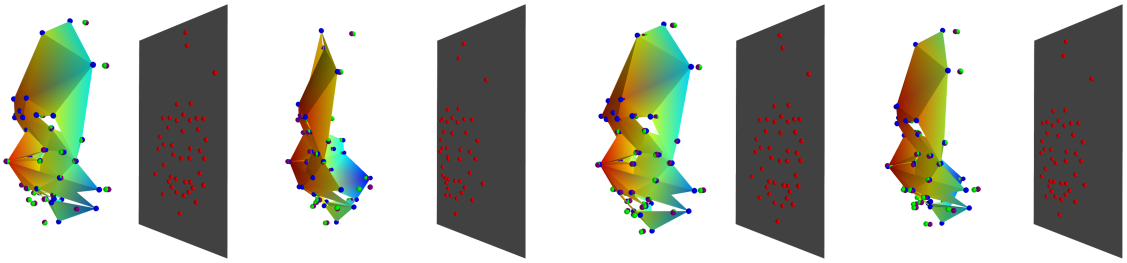

Figure 4.6: 4 frames of JACKY with LRM reconstruction (depth colored triangles and blue points), ground truth points (in green), Torresani et al. 2008 (in purple) and image points (in red).

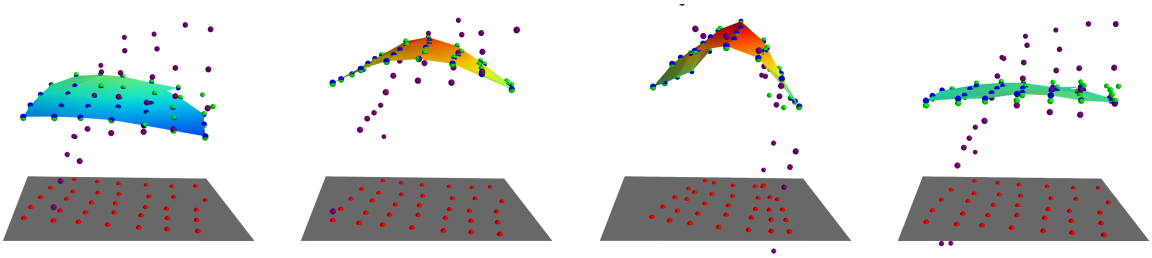

Figure 4.7: 4 frames of BEND with LRM reconstruction (depth colored triangles and blue points), ground truth points (in green), Torresani et al. 2008 (in purple) and image points (in red).

Vicente *et al.* , our reconstruction error is higher due to an incorrect flip assignment in the forehead that lasts for a few frames. This is illustrated in the second image of Figure 4.6.

4.3.3 BEND

The result of our algorithm on the BEND dataset is illustrated in Figure 4.7. This dataset consists of 34 trajectories across 471 frames. The sequence contains a piece of paper that is initially nearly fronto-parallel and then is slowly bent. Although our local rigidity assumptions hold well here and there is a fair degree of viewpoint variation, the main difficulty is with flip resolution due to fronto-parallel edges. In contrast, factorization models can model this type of motion but find their optimization becomes trapped in bad local minima due to a rigid initialization. Indeed, the best rigid model that can fit this data is actually an S shape, where one end of the paper bends towards the camera and the other bends away, rotating to model the image observations. This is best illustrated in the third image of Figure 4.7. Again, Vicente *et al.* produce a similar error to us as expected.

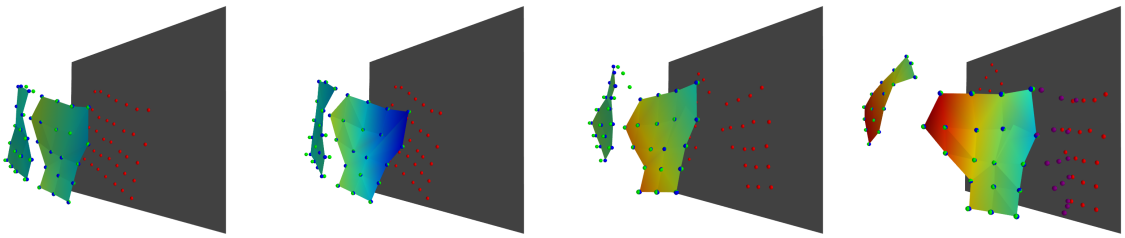

Figure 4.8: 4 frames of RIP with LRM reconstruction (depth colored triangles and blue points), ground truth points (in green), Torresani et al. 2008 (in purple) and image points (in red).

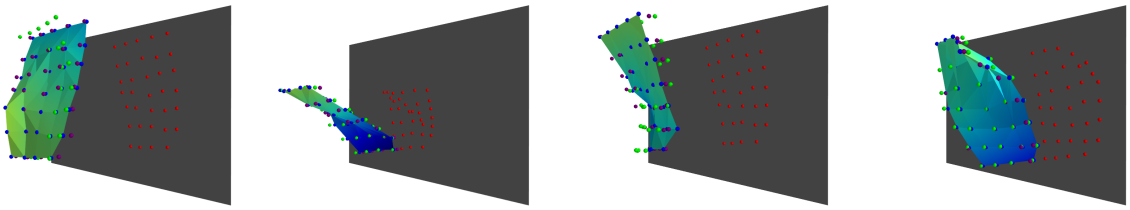

Figure 4.9: 4 frames of CLOTH with LRM reconstruction (depth colored triangles and blue points), ground truth points (in green), Torresani et al. 2008 (in purple) and image points (in red).

4.3.4 RIP

The result of our algorithm on the RIP dataset is illustrated in Figure 4.8. This dataset consists of 35 trajectories across 1227 frames. This sequence contains the motion of a piece of paper, that is slowly ripped in half in front of the camera. This is a challenging sequence for other methods, that cannot model this topology change. In contrast, our algorithm finds two resolvable components and does a good job reconstructing them.

4.3.5 CLOTH

The result of our algorithm on the CLOTH dataset is illustrated in Figure 4.9. This dataset consists of 34 trajectories across 796 frames. The sequence contains the motion of a light cloth being waved so that it deforms in the wind. Due to the lighter material, there is more local non-rigid deformation than with the sequences using paper. This compounds with nearly fronto-parallel edges to present a challenge to our algorithm. The complex deformation, however, is still a challenge for factorization based methods which we beat by a considerable margin.

4.3.6 Sensitivity to Noise

To evaluate our algorithm's robustness to noise, we add Gaussian noise to the projected ground truth locations proportional to some factor α of the 2D dataset size. More precisely, in contrast to (4.49), we now define the image observation for each point index $n \in \{1, ..., N\}$ and frame $f \in \{1, ..., F\}$ as

$$w'_{fn} = \Pi p^{gt}_{fn} + \epsilon_{fn}(\alpha), \tag{4.52}$$

where $\epsilon_{fn}(\alpha) \sim N(0, \alpha\sigma_{2D})$ and σ_{2D} measures the scale of the 2D ground truth data defined by

$$\sigma_{2D} = \frac{1}{2F}\sum_{f=1}^{F}(\sigma_{fx} + \sigma_{fy}) \ . \tag{4.53}$$

where σ_{fx} and σ_{fy} are the standard deviations of the x and y components of the N noise free image observations in frame f.

We perform 10 runs for each value of $\alpha \in \{\beta/100\}_{\beta=0}^{5}$ and plot the mean and standard deviation of the resulting normalized RMS 3D error $\mathcal{E}(c)/\sigma_{2D}$ for component c. Note that when noise is added to the image projections, the reprojection error ϵ^τ for each triangle $\tau \in \mathbb{T}_{\text{prop}}$ will change, and thus the rigidity cutoff ϵ^* would have to be heuristically reset for each noise level. To avoid this obstacle, and to ensure that we always reconstruct the same points P_c for component c, we instead arrange for the soup of triangles $\mathbb{T}^{i,\alpha}_{\text{soup}}$ to be fixed for each run i and noise level α. We do this by using the set of triplets $\mathbb{T}^{*}_{\text{soup}}$ from a single noiseless run and force every run at all noise levels to utilize this same set of triplets. That is, for every run i and noise level α, we replace the filtering step of LRM by simply setting $\mathbb{T}^{i,\alpha}_{\text{soup}} = \mathbb{T}^{*}_{\text{soup}}$.

The results of this experiment are plotted in Figure 4.10. In general, the results indicate that the algorithm, given $\mathbb{T}_{\text{soup}}$, can tolerate these noise levels as normalized RMS 3D errors of up to .20 have reasonable reconstructions (e.g. the noiseless reconstruction of `JACKY` in Figure 4.6). It is also interesting that some sequences, such as `BEND`, have very large error bars due to a bimodal distribution over the reconstruction error. This points to a drastic change in the reconstruction, caused by different flip vectors being inferred, which causes the algorithm to, on occasion, be quite sensitive to its input. Although this could be the flip optimization getting stuck in local minima, the fact that things improve occasionally with more noise (e.g. on the `BEND` sequence), suggests that the *same* local minimum is just moving slightly. That is, the constraints that the triangle models provide, dictate a low energy local minimum consistent with a poor reconstruction. This points to a fundamental limit in how well the flip energy can serve as a proxy metric for

reconstruction error. One might consider more complicated energy functions, as is done in the next chapter, but this will also open up new optimization challenges.

4.3.7 Evaluation of Flip Assignment Strategies

Here we compare the performance and runtime of a few different strategies for finding a flip assignment y with low energy $F(y)$. We will use the flip assignment inferred using the greedy algorithm as a baseline [49] and consider improving these solutions either using the principled QPBOI method or using a variety of fusion moves. Specifically, we consider the following strategies:

- GREEDY: This is simply the greedy algorithm from section 4.2.1.
- QPBOI: Improve the labelling y_0 provided by the greedy algorithm using $y_{k+1} = QPBOI(y_k)$ until the energy $F(y)$ does not change for 1 iteration.
- RANDOM: Improve the labelling y_0 provided by the greedy algorithm using $y_{k+1} = FUSE(y_k, y_{prop})$ until the energy does not change for 20 iterations. Here y_{prop} is selected using the RANDOM strategy as defined in section 4.2.1.
- RANDOM MST: Improve the labelling y_0 provided by the greedy algorithm using $y_{k+1} = FUSE(y_k, y_{prop})$ until the energy does not change for 20 iterations. Here y_{prop} is selected using the RANDOM MST strategy as defined in section 4.2.1.
- REWEIGHTED: Improve the labelling y_0 provided by the greedy algorithm using $y_{k+1} = FUSE(y_k, y_{prop})$ until the energy does not change for 20 iterations. Here y_{prop} is selected using the REWEIGHTED strategy as defined in section 4.2.1.
- CHOICE: Improve the labelling y_0 provided by the greedy algorithm using $y_{k+1} = FUSE(y_k, y_{prop})$ until the energy does not change for 20 iterations. Here y_{prop} is chosen to be constructed with either the RANDOM MST or REWEIGHTED strategy with equal probability.

In Figure 4.11, we have plotted the energy yielded by a strategy against its runtime. We ran each strategy (except GREEDY since it initializes the others) once on each dataset up to a maximum of five minutes. The QPBOI strategy almost always converges to a slightly lower energy than the fusion strategies. Unfortunately, its runtime performance is difficult to justify in the larger sequences, such as `SCARF` and `TWO CLOTHS`, and would generally affect the scalability of our algorithm.

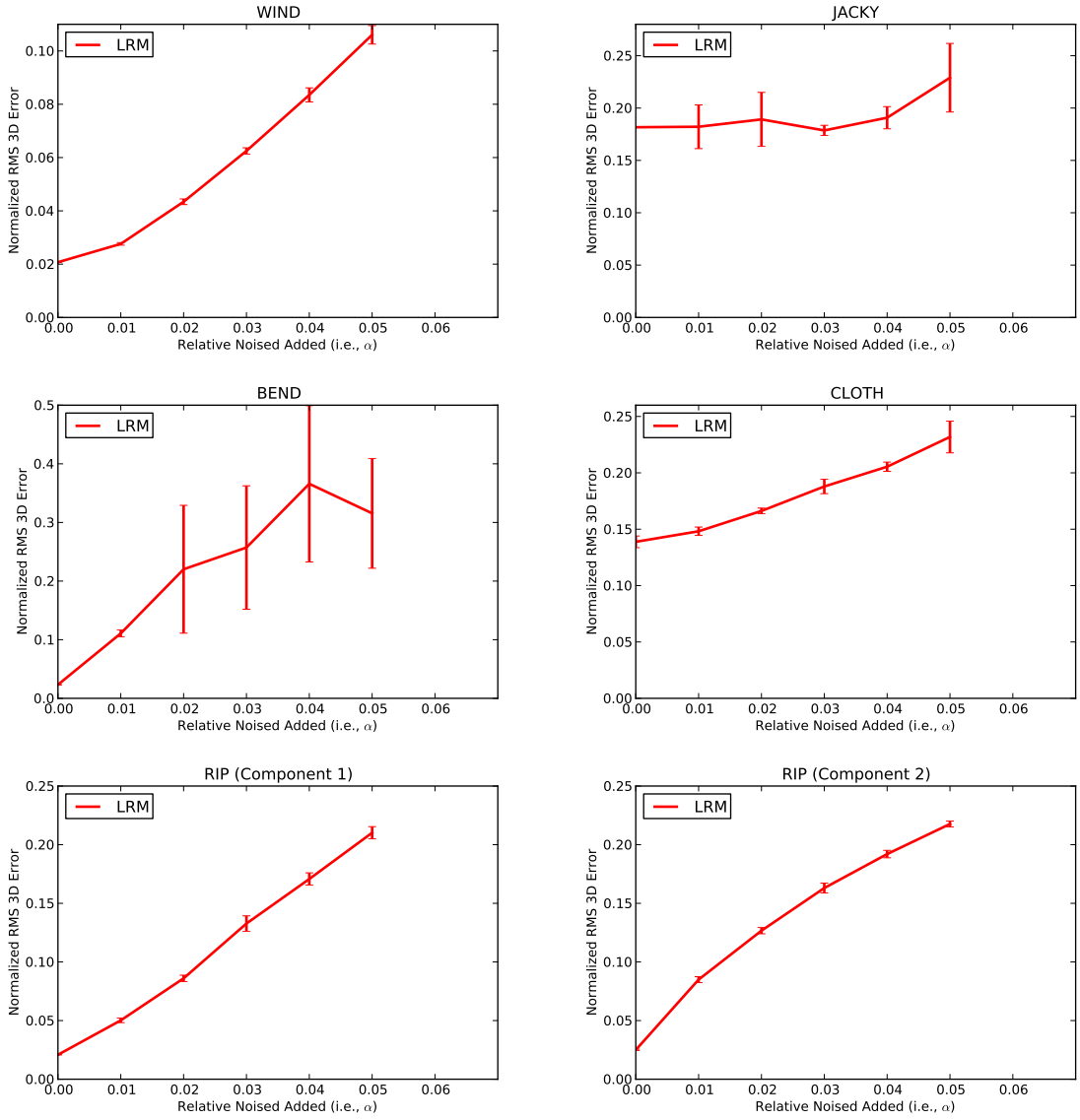

Figure 4.10: Effect of Gaussian noise in the observed image trajectories, on normalized 3D error. The setting α roughly corresponds to the fraction of noise added, relative to the scale of the dataset.

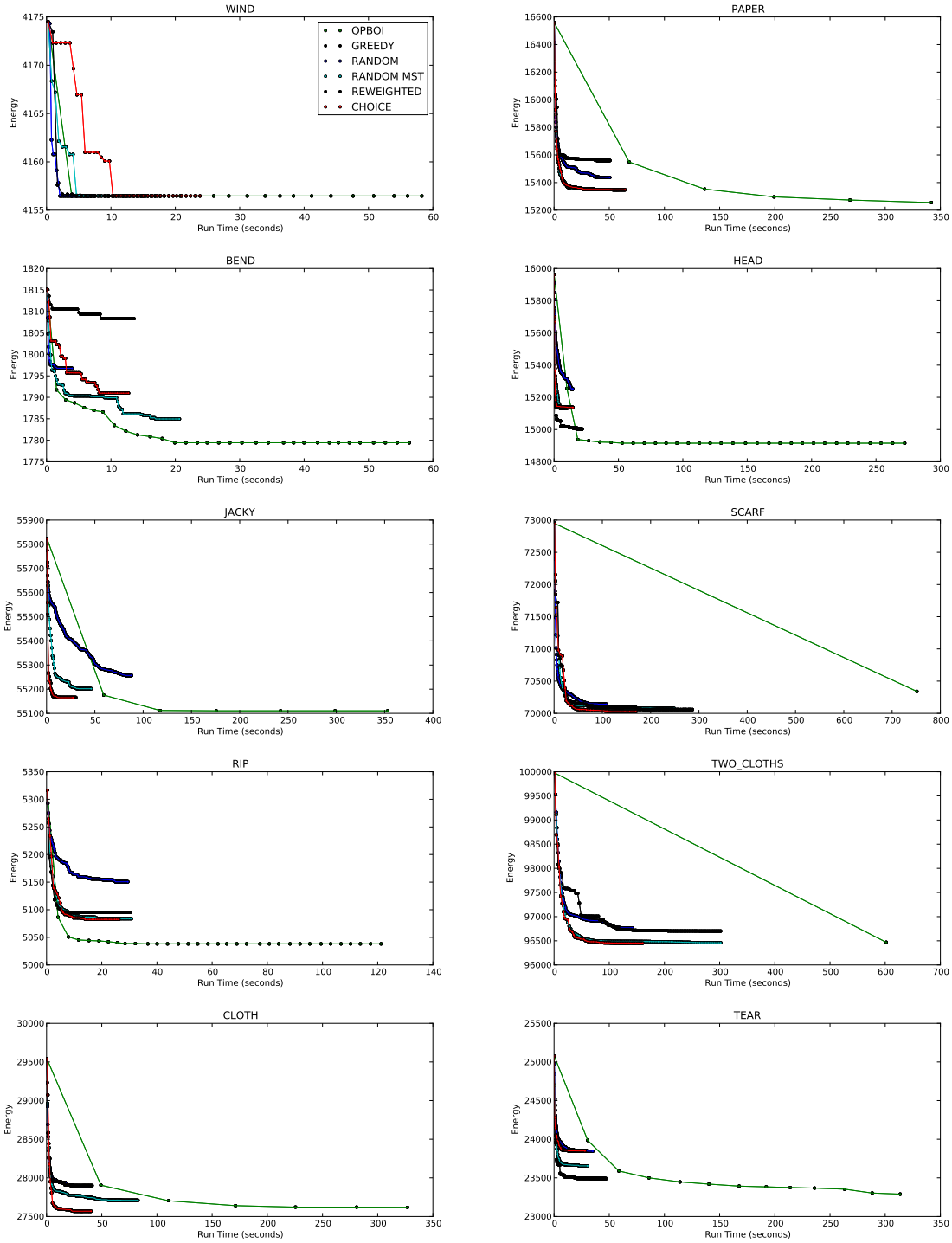

Figure 4.11: Evaluation of different flip improvement strategy. Although QPBOI is somewhat more principled, using a variety of fusion moves tends to be more efficient.

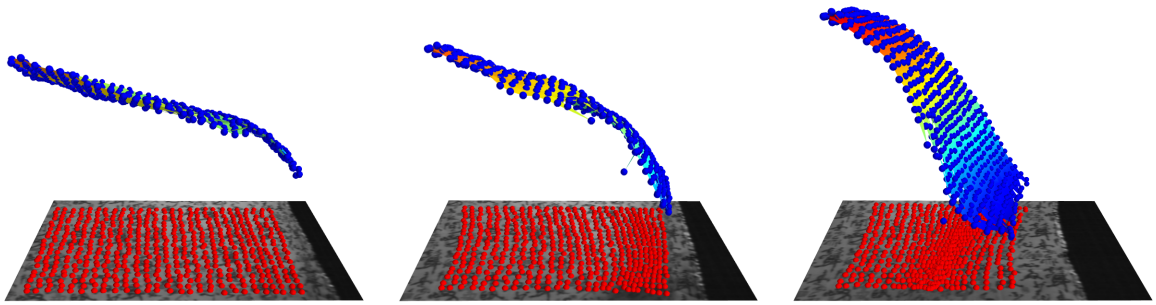

Figure 4.12: 3 frames of SCARF with LRM reconstruction (depth colored triangles and blue points) and image points (in red).

Knowing that our MRF formulation is likely not optimal for most problems anyways, we don't concern ourselves with obtaining the absolute lowest possible energy. We therefore use the CHOICE strategy, but cut off after only 5 iterations that have not made progress, which tends to converge very quickly, even for extremely large problems.

4.4 Qualitative Results on Real Sequences

It is also important to evaluate our algorithm using trajectories that we can obtain for real image sequences. We describe five such sequences and their reconstructions below. For the PAPER, SCARF and TWO CLOTHS sequences we use image trajectories sampled from an estimate of dense 2D motion [50]. For the sequences HEAD and TEAR, we use trajectories obtained from tracking feature points [29].

4.4.1 SCARF

The result of our algorithm on the SCARF dataset is illustrated in Figure 4.12. This dataset consists of 567 trajectories across 101 frames. In this sequence, a scarf is being waved rapidly in front of a high speed camera. We appear to do quite well, as the trajectories are quite accurate in representing plausible locally rigid motion of the cloth. Even though there are long lines of fronto-parallel edges, our strategy of randomly selecting trajectories to triangulate (see Section 4.1.1 and Figure 4.1) allow triangles to span these edges and provide strong constraints to the flip MRF.

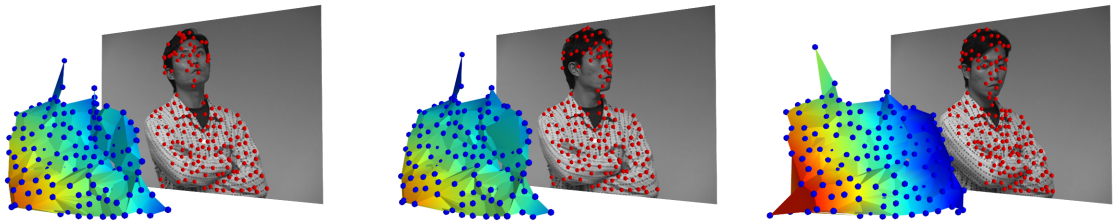

Figure 4.13: 3 frames of HEAD with LRM reconstruction (depth colored triangles and blue points) and image points (in red).

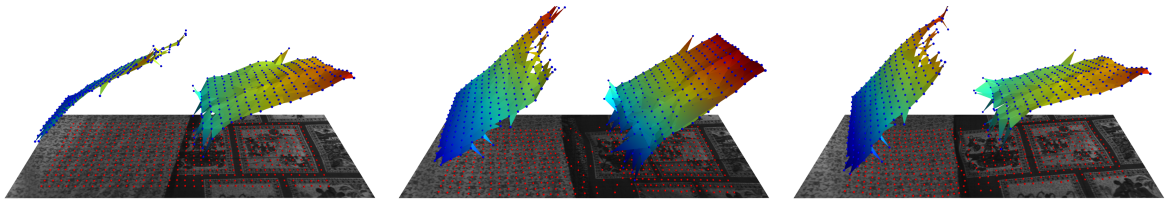

Figure 4.14: 3 frames of TWOCLOTHS with LRM reconstruction (depth colored triangles and blue points) and image points (in red).

4.4.2 HEAD

The result of our algorithm on the HEAD dataset is illustrated in Figure 4.13. This dataset consists of 160 trajectories across 61 frames. This is a sequence, from [64], of a person rotating their torso and their head back and forth. The head and torso are both nearly rigid, so ideally our method would segment the two. Unfortunately, many of the trajectories slide along image edges, including nearly all of the trajectories on the head. We thus only display the torso, which LRM provides a very plausible reconstruction for. There are, however, some creases in his chest due to incorrect flips which will become more apparent when we compare to the reconstruction provided by our approach in the next chapter.

4.4.3 TWO CLOTHS

The result of our algorithm on the TWO CLOTHS dataset is illustrated in Figure 4.14. This dataset consists of 525 trajectories across 163 frames. This sequence is very similar to SCARF, except now there are two cloths deforming side by side. Our algorithm is able to successfully separate and reconstruct each piece but there are some long triangles hanging off the reconstruction.

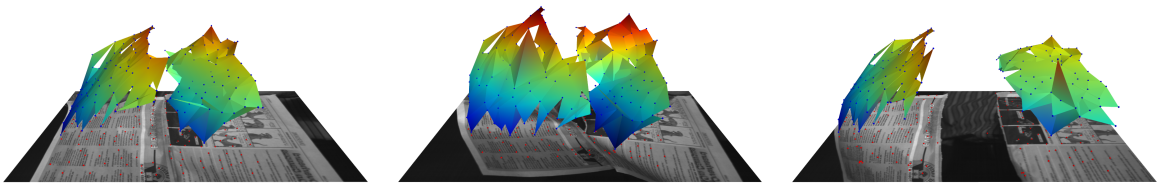

Figure 4.15: 3 frames of TEAR with LRM reconstruction (depth colored triangles and blue points) and image points (in red).

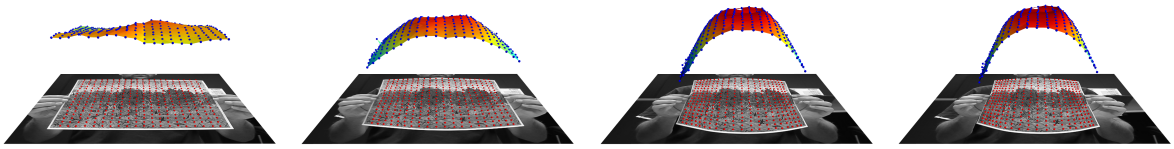

Figure 4.16: 4 frames of PAPER with LRM reconstruction (depth colored triangles and blue points) and image points (in red).

4.4.4 TEAR

The result of our algorithm on the TEAR dataset is illustrated in Figure 4.15. This dataset consists of 159 trajectories across 167 frames. This is an extremely challenging sequence of a piece of paper being ripped in half. Although, topologically similar to the RIP ground truth sequence, it was extremely difficult to track image features due to motion blur, sparse texture and different depths being out of focus. Nonetheless, we do appear to resolve two separate components and qualitatively, there is evidence of a ripping motion occurring in our reconstruction.

4.4.5 PAPER

The result of our algorithm on the PAPER dataset is illustrated in Figure 4.16. This dataset consists of 340 trajectories across 70 frames.

The PAPER sequence shows a piece of paper being bent from its original planar configuration into a U shape. Orthographically, such a motion has an equally valid S-shaped interpretation that is often recovered. Also, the rotation is almost entirely around the y-axis, leaving the motion quite degenerate and the rigid triangle models underconstrained. A further difficulty is the large perspective effects present, causing the middle of the paper to bulge in the image when it is bent (see Figure 1.1). This, further, challenges our test for rigidity as a triangle modelling a triplet of points on the middle of the paper cannot lie fronto-parallel in the initial frames, and expand to model the bulging in the later frames. Such triangles then generate a higher than expected reprojection error and

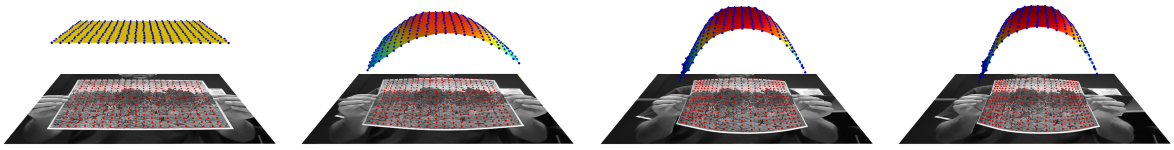

Figure 4.17: 4 frames of PAPER with LRM (from a planar template) reconstruction (depth colored triangles and blue points) and image points (in red).

we therefore choose to weaken our rigidity test by setting the cutoff as $\eta = 2.5$. Using this, we appear to do well, although these perspective effects prevent the initial frames of the reconstruction from being planar as they should.

This sequence is also unique, in that we can clearly identify the planar structure in the first frame. We can therefore also reconstruct it using a soup of triangles extracted from a template (see Section 4.1.2). The noticeably higher quality reconstruction, illustrated in Figure 4.17, is due to the fact we are given the local structure, and we only need to estimate its motion.

4.5 Conclusion

In this chapter, we have provided a solution to the Non-Rigid Structure from Motion problem composed of both a piecewise local model of global scene deformation and an algorithm to fit this model. This piecewise local model is composed of a set of loosely connected rigid triangle models whose averaged vertices explain the non-rigid motion in a scene. Our algorithm LRM for fitting this model to a set of image trajectories, involves breaking up the problem into a series of tractable sub-problems, each of which can be solved individually in a bottom-up fashion. This method begins by identifying triplets of trajectories that can be modeled as the projected vertices of a rigid triangle model. By resolving each triangle's depth flip and translation ambiguities, we obtain our piecewise model.

Using ground truth sequences, we have shown that LRM is generally competitive with other state of the art algorithms and considerably outperforms standard factorization based approaches when our assumptions strongly hold. Further, our algorithm appears to be robust to noise, although this property does not always hold due to the brittleness of the discrete depth flip optimization step. On image trajectories obtained from real sequences, we achieve plausible reconstructions despite non-Gaussian noise, non-uniform trajectory coverage and degenerate motions all presenting challenges to our algorithm.

Chapter 5

Globally Optimized Locally Rigid Motion

This chapter describes our global point cloud based model for non-rigid scene structure, regularized by a set of pairwise isometric constraints. An isometric constraint $(n, m) \in \{0, ..., N\}^2$ requires scene points with indices n and m to remain at a fixed distance across all views. A set of such constraints $\mathcal{L} \subseteq \{1, ..., N\}^2$ is the key to regularizing the otherwise ill-posed problem of inferring the 3D scene point's locations, given their observed 2D trajectories, but naturally their strict imposition will quickly lock the entire scene into rigidity. Indeed, rigid models such as those fit by 3-SFM or rigid factorization could be described equivalently by a set of such strict isometric constraints between all pairs of points. In order to allow the flexibility to model non-rigid scenes, it is then necessary that these constraints are not strictly enforced. The local rigidity framework from the previous chapter achieves this by not requiring two triangle models τ and τ' that model some feature point n through vertices i and j to have their positions coincide (i.e., the constraint $p^{\tau}_{fi} = p^{\tau'}_{fj}$ for every frame f is not enforced). In order for the final "reconstruction" stage to impose this restriction and produce a single prediction for each point, these corresponding vertex locations are averaged, necessarily violating the isometric constraints within each triangle model.

In contrast, the model that will be presented here only ever allows for a single reconstructed scene point to correspond to each feature point, and thus cannot directly enforce these isometric constraints. Instead, the isometric constraint violations are simply discouraged through the use of an additional term in an energy based formulation.

Such an energy based formulation presents three main questions that this chapter attempts to address.

1. How can one select the set of isometric constraints $\mathcal{L} \subseteq \{1, ..., N\}^2$ that the energy function requires?

2. The resulting energy will generally be non-convex and littered with local-minima, so how can one effectively optimize this energy?

3. What sort of penalties should be imposed in the energy when the isometric constraints are violated?

To address the first two questions, we appeal again to the notion of local rigidity presented in the previous chapter. As a valid rigid triangle model implicitly constrains three pairs of points to be isometric, we can use the same strategy of probing the scene locally using 3-SFM to identify isometric constraints. In practice, we can simply run the LRM algorithm provided in the previous chapter. This provides us with the set of constraints, but also provides a complete approximate reconstruction.

This reconstruction, can then used as an initial guess to fast local search methods that refine the reconstruction through a bundle adjustment like minimization of the energy function. Under harsh penalties (e.g. squared error) one might expect this to refine and polish the initial reconstruction, but with more robust penalties, it also has the potential to allow subsets of these constraints to be overridden to create a better global consensus.

As this energy based formulation is a very natural (albeit challenging) way to approach things, it is not surprising that a similar but independent formulation was proposed in [60].[1] That work is briefly described in Section 2.4, but a more detailed comparison is provided in Section 5.1.5 in a notation consistent with this thesis, in order to elucidate contrasts with the work presented here. Interestingly, they provide a very different and complimentary set of answers to the above three questions.

5.1 Formulation

The energy function that will be formulated here will assign an energy value $E(\theta; \mathcal{L})$ to a vector $\theta \in \mathbb{R}^d$ that we will use to parameterize our model. We include in this vector, the concatenation of the $3FN$ components contained in the N points deforming in front of each camera, and thus write $p_{fn}(\theta) \in \mathbb{R}^3$ to indicate this dependence. Residuals errors from the observed image point w'_{fn} can then be penalized directly in the energy through

[1] Note that [60] was published after the author of this thesis presented a version of this energy based formulation in his depth examination in April 2012.

a data term. This term is

$$E_{data}(\theta) = \sum_{f=1}^{F} \sum_{n=1}^{N} \|\Pi p_{fn}(\theta) - w'_{fn}\|^2 \ . \tag{5.1}$$

Note that, for notational simplicity, the dependence of energy terms like $E_{data}(\theta)$ on the observed data points themselves is left implicit.

Naturally, the energy $E_{data}(\theta)$ leaves the parameter vector highly ambiguous, as each point can move arbitrarily in depth without changing the resultant energy value. This untethered movement, however, can be discouraged by an isometric constraint $(i, j) \in \mathcal{L}$, that indicates that points i and j should remain at a constant distance to each other. We do not necessarily know this distance beforehand, and thus we also include it in the parameter vector, denoting its value as $L_{ij}(\theta) \in \mathbb{R}^+$ to indicate this dependency. We can then formulate an additional energy term that encourages $\|p_{fi}(\theta) - p_{fj}(\theta)\|$ to be equal to $L_{ij}(\theta)$ for all frames as

$$E_{iso}(\theta; \mathcal{L}) = \sum_{f=1}^{F} \sum_{(i,j)\in\mathcal{L}} \rho_{iso}(\|p_{fi}(\theta) - p_{fj}(\theta)\| - L_{ij}(\theta)) \tag{5.2}$$

where ρ_{iso} is some appropriate error function. We will discuss the choice of this function in Section 5.1.1.

Of course we may have other prior knowledge about these constraints or the scene, so it is natural to consider encoding additional priors as terms in the energy. For example, we know that the image trajectories that we consider, arising from image or motion capture sequences, are discrete samples of temporally continuous scene deformation. It is natural, therefore, to encourage our model of scene deformation to plausibly represent such a sampling by penalizing large structural deviations from frame to frame. This preference is encoded as another energy term

$$E_{temp}(\theta) = \sum_{f=1}^{F-1} \sum_{n=1}^{N} \|p_{(f+1)n}(\theta) - p_{fn}(\theta)\|^2 \ . \tag{5.3}$$

Further, it was demonstrated in previous chapters, that the strict isometric constraints in triangle models are often not enough to constrain the model when the viewpoints are degenerate. In this model, we have actually softened these isometric constraints in (5.2). On the other hand, these degeneracies are likely to be somewhat ameliorated as each point can be constrained by a larger set of image observations, coming from its

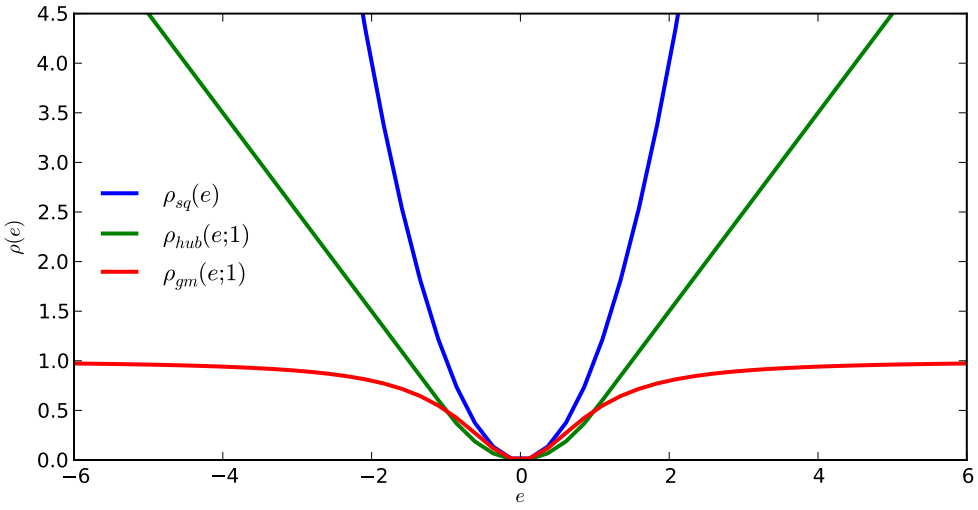

Figure 5.1: The three different error functions considered for penalizing isometric constraints.

isometrically constrained neighbours. For cases when this is not enough, however, we simply add a prior term, similar to that in 3-SFM, that penalizes squared edge lengths, as

$$E_{prior}(\theta;\mathcal{L}) = \sum_{(i,j)\in\mathcal{L}} L_{ij}(\theta)^2 \ . \tag{5.4}$$

Combining the energies (5.1), (5.2), (5.3) and (5.4) with scalars λ_{data}, λ_{iso}, λ_{temp} and λ_{prior} yields our full energy

$$E(\theta;\mathcal{L}) = E_{data}(\theta) + \lambda_{iso}E_{iso}(\theta;\mathcal{L}) + \lambda_{temp}E_{temp}(\theta) + \lambda_{prior}E_{prior}(\theta;\mathcal{L}) \ . \tag{5.5}$$

5.1.1 Choice of Isometric Error Function

The choice of the error function ρ_{iso} used to penalize deviations from our set of isometric constraints is particularly interesting. This is because there is often not a reliable way to obtain the set of isometric constraints $\mathcal{L}$, and thus we might actually expect a good deal of them to be wrong. We therefore consider three different penalties of varying degrees of robustness as illustrated by Figure 5.1. The first is a harsh standard squared error measure

$$\rho_{sq}(e) = e^2. \tag{5.6}$$

The second is a smoothed L1-like measure called the Huber Loss Function [25]

$$\rho_{hub}(e;\delta) = \begin{cases} \frac{e^2}{2} & |e| < \delta \\ \delta(|e| - \frac{\delta}{2}) & |e| \geq \delta \end{cases} . \tag{5.7}$$

This error is as robust and attentive to large errors as a standard L1 error, but in contrast, its continuous first derivative causes less trouble for standard gradient based methods. Lastly, we consider a very robust measure called the Geman and McClure Function[8]

$$\rho_{gm}(e;\sigma) = \frac{e^2}{e^2 + \sigma^2} . \tag{5.8}$$

This function is quadratic near zero, but quickly approaches unity for absolute values over (roughly) 2σ, allowing it to largely ignore errors of such magnitude (i.e., its derivative will be nearly zero).

When we are confident about the constraints, we might indeed want to use ρ_{sq} to harshly penalize isometric deviations. If, on the other hand, we do not believe all of these constraints to be correct, we could use a more robust function. For example, using ρ_{gm} might allow $E(\theta;\mathcal{L})$ to completely ignore faulty isometric constraints when it is minimized. Even though robust error functions generally add more local minima to a function, such a function may also change the energy landscape in ways that allow other, stronger terms to provide paths that descend to better local minima.

5.1.2 Optimization

The approach to optimization taken here is to use local gradient based methods, and thus some care was taken to ensure that the function $E(\theta;\mathcal{L})$ is differentiable. This is somewhat similar to a standard bundle adjustment [55] except that without a rigid scene, there are many more parameters, of which many are tied through the isometric constraints. We thus utilize L-BFGS that maintains approximate second order information instead of requiring the entire Hessian (or its inverse).

5.1.3 Initialization of θ and $\mathcal{L}$

The final ingredient in order for $E(\theta;\mathcal{L})$ to be well defined, is to determine which isometric constraints should be contained in $\mathcal{L}$. Also, in order for our bundle adjustment-like optimization to find a good local minimum, we will require a reasonable initialization of θ.

For this we appeal to the notion of local-rigidity, once again, to find and test isometric constraints. In practice, we simply run the local rigidity procedure detailed in the previous chapter. We will then optimize each resolvable component $\mathbb{T}_c \subseteq \mathbb{T}_{\text{soup}}$ individually. To simplify notation, we thus assume that we are working with a single resolvable component, which spans the entire point set (i.e., $P_c = \{1, ..., N\}$).

We then use each side of each rigid triangle in this component to define the set of isometric constraints as

$$\mathcal{L} = \{(\tau_i, \tau_{i \bmod 3+1}) : i \in \{1, 2, 3\}, \tau \in \mathbb{T}_c\} \ . \tag{5.9}$$

For any such link $(i, j) \in \mathcal{L}$, we also initialize the appropriate component of the parameter vector θ to ensure that $L_{ij}(\theta)$ is the median of the set of framewise distances

$$\{\|p^c_{fi} - p^c_{fj}\| : f \in \{1, ..., F\}\} \ . \tag{5.10}$$

Lastly, we set the remaining components of θ so that our model aligns with the reconstructed component. That is, for every point index n and frame f we ensure that $p_{fn}(\theta) = p^c_{fn}$.

5.1.4 Parameter Choices

The most interesting aspect of the parameter space, is the choice of the isometric error function as this fundamentally reshapes the energy landscape. Without access to training data, we therefore choose to fix the other parameters using intuitive values as detailed in Table 5.1. As with our triangle models, we only want our prior terms to take affect when the data or isometric terms fail to provide constraints. We therefore set the data terms and isometric terms to be equal to unity, and the temporal and prior term to be correspondingly very weak. We refer to optimizing the energy detailed above using the initialization and local optimization procedure explained as Locally Rigid Motion with Bundle Adjustment (LRMBA), and specifically LRMBA-SQ, LRMBA-HUB and LRMBA-GM to indicate whether ρ_{iso} is set to ρ_{sq}, ρ_{hub} or ρ_{gm}.

5.1.5 Relation to Vicente *et al.* 2012

An alternative approach to this bundle adjustment like method is proposed in [60]. In their method, a similar energy $E(\theta; \mathcal{L})$ is formulated and the set $\mathcal{L}$ is set using pairs of points with consistently near image observations. The optimization procedure, tries to improve the current solution θ_k at iteration k, using a new proposal solution θ'_k.

Parameter	Value
λ_{data}	1
λ_{iso}	1
λ_{temp}	0.01
λ_{prior}	0.01
σ	1.0
δ	1.0

Table 5.1: The fixed parameters of our energy. We set both the data term and isometric term to be equal to unity and allow the choice of isometric penalty (see Section 5.1.1) dictate how these interplay. We considerably downweight the two priors so that they only take affect when the other terms leave the local parameter space underconstrained.

This proposal is then fused together into a new solution θ_{k+1} in such a way [35] that $E(\theta_{k+1};\mathcal{L}) \leq E(\theta_k;\mathcal{L})$ and $E(\theta_{k+1};\mathcal{L}) \leq E(\theta'_k;\mathcal{L})$. This has the distinct advantage, over bundle adjustment, of being able to jump out of local minima if a proposal allows it, but each iteration is expensive and requires a thoughtful proposal to be chosen. Also, it is not guaranteed that these methods will jump out of local minima and thus the faster convergence generally exhibited by continuous second order methods may be preferred. We will now briefly provide a simplified description of their general method, using the notation of this thesis, in order to draw comparisons to our approach.

Isometric Constraint Set. In this work, they also require a set of pairwise isometric constraints between points. To decide which pairs of the N points to constrain, they define a distance metric between two point indices n and m to be the maximum over all their projected distances in all frames

$$d(n,m) = \max_f \|w'_{fn} - w'_{fm}\| \ . \tag{5.11}$$

For each point index n, they define $K_n \subseteq \{1,...,N\}$ to include the k-nearest neighbours (excluding n) under this metric. They then define a set of isometric constraints between these neighbours, up to a maximum distance t as

$$\mathcal{L} \subseteq \{(n,m) : n \in \{1,...,N\}, n \in K_n, n < m, d(n,m) \leq t\} \ . \tag{5.12}$$

where the parameters are generally set to $k = 10$ and $t = 150$. As with our method the assumption that nearby points are more likely to be nearly rigid is being leveraged. In contrast, however, our method is often able to test, using 3-SFM, to see if the constraints are valid.

Model Paramaterization. They also parameterize their model using a vector θ which is composed in the following way. For the n'th point, they dedicate a component of θ to its depth in each frame f, and thus one can write $z_{fn}(\theta)$ to indicate this dependence. They do not, however, parameterize the x and y components of the point, and instead constrain that point to lie on the back projected ray through $w'_{fn} = \begin{bmatrix} x'_{fn} \\ y'_{fn} \end{bmatrix}$. They can thus write

$$p_{fn}(\theta) = \begin{bmatrix} x'_{fn} \\ y'_{fn} \\ z_{fn}(\theta) \end{bmatrix} . \tag{5.13}$$

For each constraint $(n, m) \in \mathcal{L}$ they also use a component of θ to parameterize the current estimate of the true distance between n and m by $L_{nm}(\theta)$. In contrast to our model that implicitly assumes a Gaussian noise model allowing the image projections to deviate from the observations, this model requires isometric constraints to be satisfied purely by movements in and out of depth. Our model has the disadvantage of having to keep track of these extra $2FN$ parameters.

Energy Formulation. Given this parameterization, they formulate the following energy function over the $FN + |\mathcal{L}|$ parameters that they would like to minimize:

$$E(\theta; \mathcal{L}) = \sum_{f=1}^{F} \sum_{(i,j)\in\mathcal{L}} |\|p_{fi}(\theta) - p_{fj}(\theta)\| - L_{ij}(\theta)| \tag{5.14}$$

They, also, admit the possibility of adding additional terms to the energy in order to encode temporal smoothness, similar to (5.3), or spatial smoothness.

Optimization. They choose to view the energy (5.14) as that of a Markov Random Field in which each term represents a potential over a clique of three variables. In order to optimize this energy, they choose to use a discrete optimization strategy in which, at each step, a proposal solution θ' is "fused" [35] to the current solution θ_k in such a way such that the resulting solution $\theta_{k+1} = FUSE(\theta_k, \theta')$ does not increase the energy (i.e., that $E(\theta_{k+1}; \mathcal{L}) \leq E(\theta_k; \mathcal{L})$ and $E(\theta_{k+1}; \mathcal{L}) \leq E(\theta'; \mathcal{L})$). The advantages of this regime are that proposal solutions can be made heuristically and that the fusion move has the ability to incorporate only part of the proposal in order to lower the value of the true energy.

This fusion move, however, does not apply directly to energies with potentials over cliques of size greater than two. It is therefore, necessary, to apply a transformation to

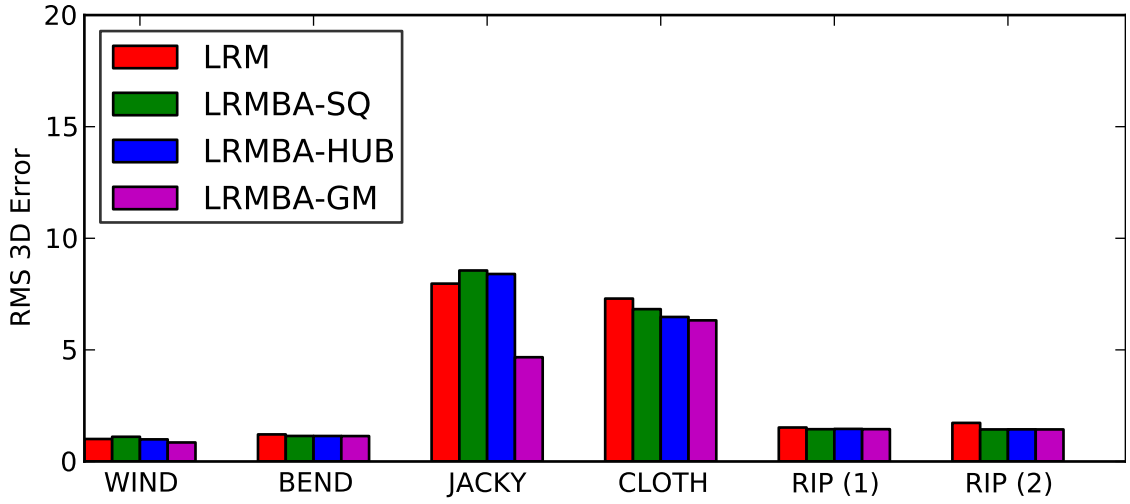

Figure 5.2: Comparison of our bundle adjustment methods (LRMBA) to our piecewise (LRM) method

the MRF by introducing extra variables [26]. The number of variables introduced by this reduction is, however, exponential in the clique size. Furthermore, it was hinted in [28] that proposal generation is further complicated with larger cliques by having "a richer class of null potentials". This makes fully parameterizing $p_{fn}(\theta)$ to contain all three components more difficult, as a naive approach could potentially introduce cliques of size $3 + 3 + 1 = 7$.

They initialize their optimization from a θ_0 in which $L_{mn}(\theta_0) = \max_f \|w'_{fn} - w'_{fm}\|$ and $z_{fn}(\theta_0) = 0$. They perform 50 fusion moves using proposals from a greedy region growing method in which the depths for a growing subset S of point indices are initially fixed. For each new point index n (in the order of distance to this set) optimal depths from a discrete set of possibilities are selected to minimize the summands in (5.14) involving n and any $m \in S$. After this, n is added to S. Finally, the distances $L_{nm}(\theta')$ for all points are then selected to be the median over all observed distances.

They perform the above procedure 8 times, and take the solution with minimum energy. From that solution, they then perform 100 fusion moves using proposals generated [27] using the gradient via

$$\theta' = \theta + \lambda \frac{\partial E(\theta; \mathcal{L})}{\partial \theta} \tag{5.15}$$

where λ is a fixed step size.

5.2 Experiments with Ground Truth

In this section we use the ground truth sequences to contrast the performance of LRM, from the previous chapter, to the three algorithms that we consider here LRMBA-SQ, LRMBA-HUB and LRMBA-GM. The main results are summarized in Figure 5.2, where it can be seen that generally, these global bundle adjustment methods provide a modest improvement improvement over LRM.

The glaring exception to this trend is the `JACKY` sequence, where the LRMBA-GM sequence does substantially better. Looking at the reconstruction (see Figure 5.5), we can see that the robust penalty is allowing the false isometric constraints on the mouth to be violated. This allows the mouth to open and close much more realistically as can be seen in the the final row of Figure 5.5. In contrast, the LRMBA-SQ and LRMBA-HUB methods do worse than LRM, as LRM can allow truly rigid triangles to "out-vote" these false constraints, whereas in LRMBA the rest of the structure is actually "optimized" so as to allow these false constraints to hold. The ability to choose different penalty functions (such as ρ_{gm}) is thus a key feature of LRMBA, in allowing the underlying model to be easily adapted to scenarios where our fundamental local rigidity assumptions do not strictly hold. Further, it is compelling that, even in such cases, LRM can be used to bootstrap the optimization so as to successfully fit the LRMBA model.

We now refocus our discussion on the dominant trend of LRMBA variants yielding modest improvements in accuracy on the remaining sequences. This result is consistent with our expectation that, when our local rigidity assumptions hold, the bundle adjustment will not lead to drastic changes in the structure and corresponding reconstruction error. Instead, the procedure may be able to average over observational noise leading to these modest improvements in performance.

Given that LRMBA generally only finds a local minimum of our energy, it is natural to wonder how good this minimum actually is. To investigate this, we also try initializing our model using ground truth, instead of from the result of LRM. For this, we set the parameters of our model θ as before (see Section 5.1.3) but using the ground truth points instead of our LRM reconstructed points. We then plot (see Figures 5.3 and 5.4), for both initializations, the RMS 3D error against the number of iterations to get an idea of the trajectory being taken by the optimizer. The LRMBA variants initialized from ground truth quickly diverge from zero error, as any local non-rigidity in the initialization will initially be violating the isometric constraints. The final result, however, is often very close to the LRM initialized result, leading us to believe that the local minimum that LRMBA finds is often close to the global minima.

On the other hand, it is clearly possible for the LRM initialization to present problems that are impossible for the local optimizer to overcome. This is most evident for the JACKY and CLOTH sequences, where there is a clear difference between the resulting RMS 3D error, when the optimization was initialized from ground truth. These divergences are often the result of a small number of incorrect flips causing gross errors in the LRM initialization, as opposed to a large number of minor errors, which the global optimization might be able to correct. One can see that this occurs in the forehead of the reconstructions of JACKY in the second column of Figure 5.5, the top of the reconstructions of CLOTH in the first column of Figure 5.9 and in the second component of the reconstructions of RIP in the third column of Figure 5.8.

5.2.1 Sensitivity to Noise

In order to evaluate the different LRMBA variants' robustness to noise, we repeat the experiment of Section 4.3.6 exactly but including these variants. The results of this experiment are plotted in Figure 5.11. Here we can see that when the local rigidity assumption holds strongly, in all sequences except JACKY, the bundle adjustment algorithms generally do a bit better as they are better able to average over the noisy image observations. In particular, LRMBA-SQ generally outperforms the other error methods as it more strongly regularizes the structure against the effects of noise.

The exception, again is the JACKY sequence where relatively low amounts of imaging noise leads to catastrophic failures in the LRMBA-SQ and LRMBA-HUB reconstructions. The added image noise puts the optimizer under intense pressure to model these errors in order to reduce the energy. Unfortunately, there is not enough viewpoint variation (as described in Section 3.3.3), to allow the soft isometric constraints to regularize against this. The optimizer finds that by extending the structure in depth (see Figure 5.10), it can approximately satisfy these constraints while lowering reprojection error. In the case of LRMBA-GM, it appears as if the robust error function has declared enough isometric constraints as outliers, that the other energy terms, mainly reprojection, can be easily satisfied, ending the optimization.

5.3 Qualitative Results on Real Sequences

We again run our algorithms on the image trajectories in the real datasets from Section 4.4 to see if there is a large difference from the LRM reconstructions. For the most part, there is not much difference which coincides with our intuition that LRM provides a good

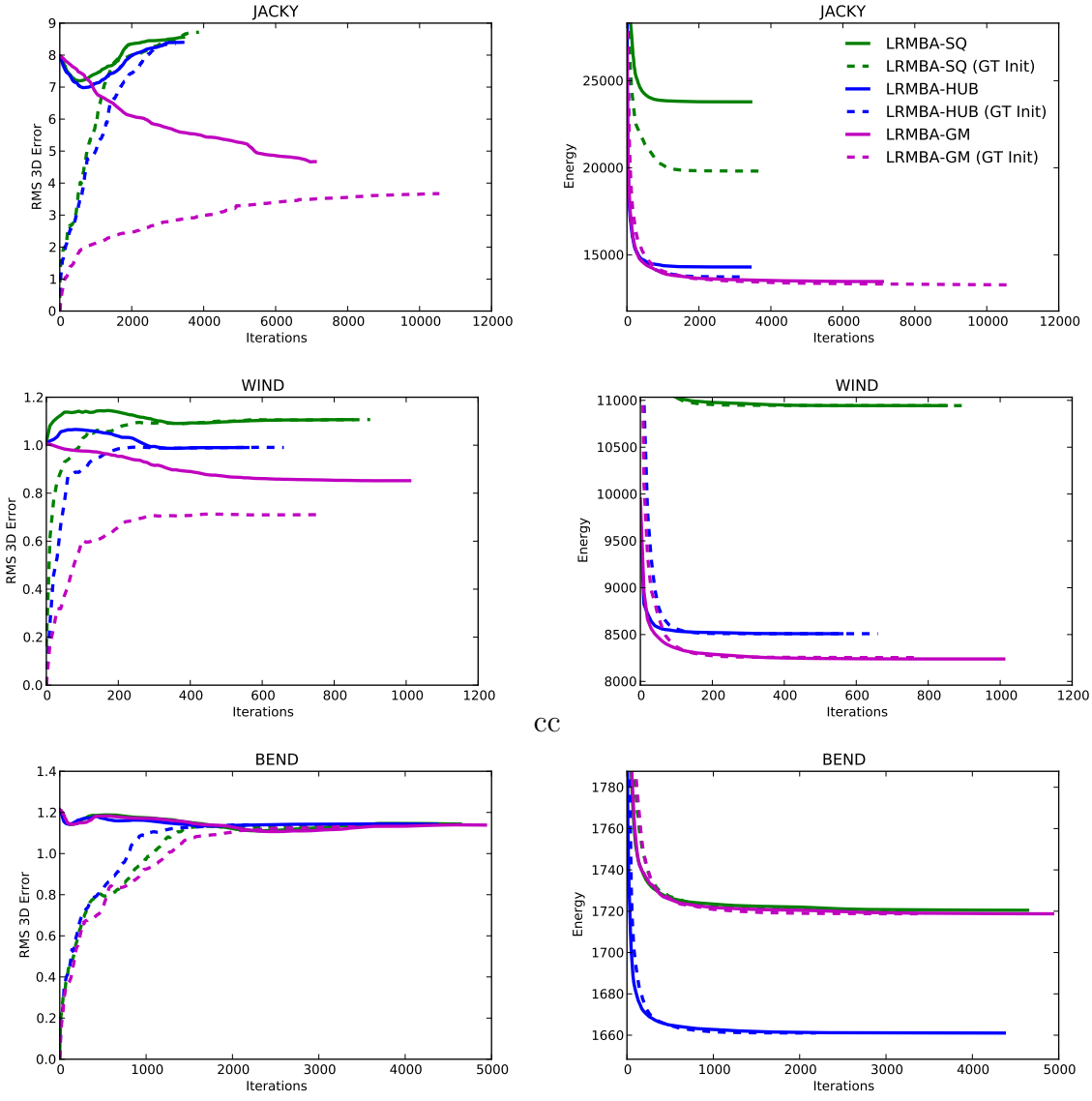

Figure 5.3: For a variety of ground truth datasets, the normalized RMS 3D error (left) and the corresponding energy (right) plotted during optimization of $E(\theta; \mathcal{L})$. The dashed lines indicate that the optimization started from ground truth. Note that the error measure is comparable across different datasets even though the limits on each axis is different. Energies are only comparable when the same energy function is used but with different initializations (i.e., The solid and dashed lines within a single plot)

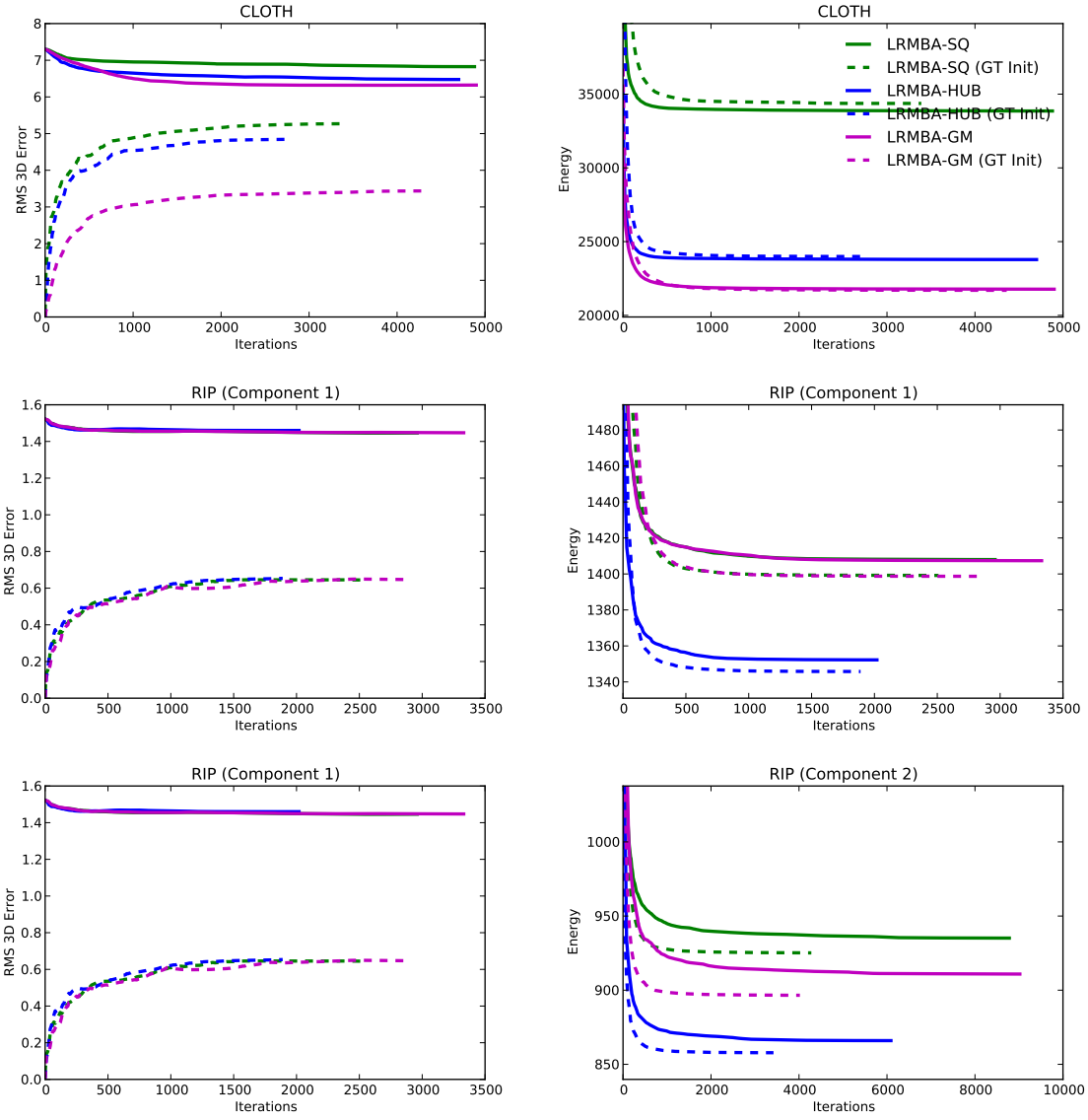

Figure 5.4: For a variety of ground truth datasets, the normalized RMS 3D error (left) and the corresponding energy (right) plotted during optimization of $E(\theta; \mathcal{L})$. The dashed lines indicate that the optimization started from ground truth. Note that the error measure is comparable across different datasets even though the limits on each axis is different. Energies are only comparable when the same energy function is used but with different initializations (i.e., The solid and dashed lines within a single plot)

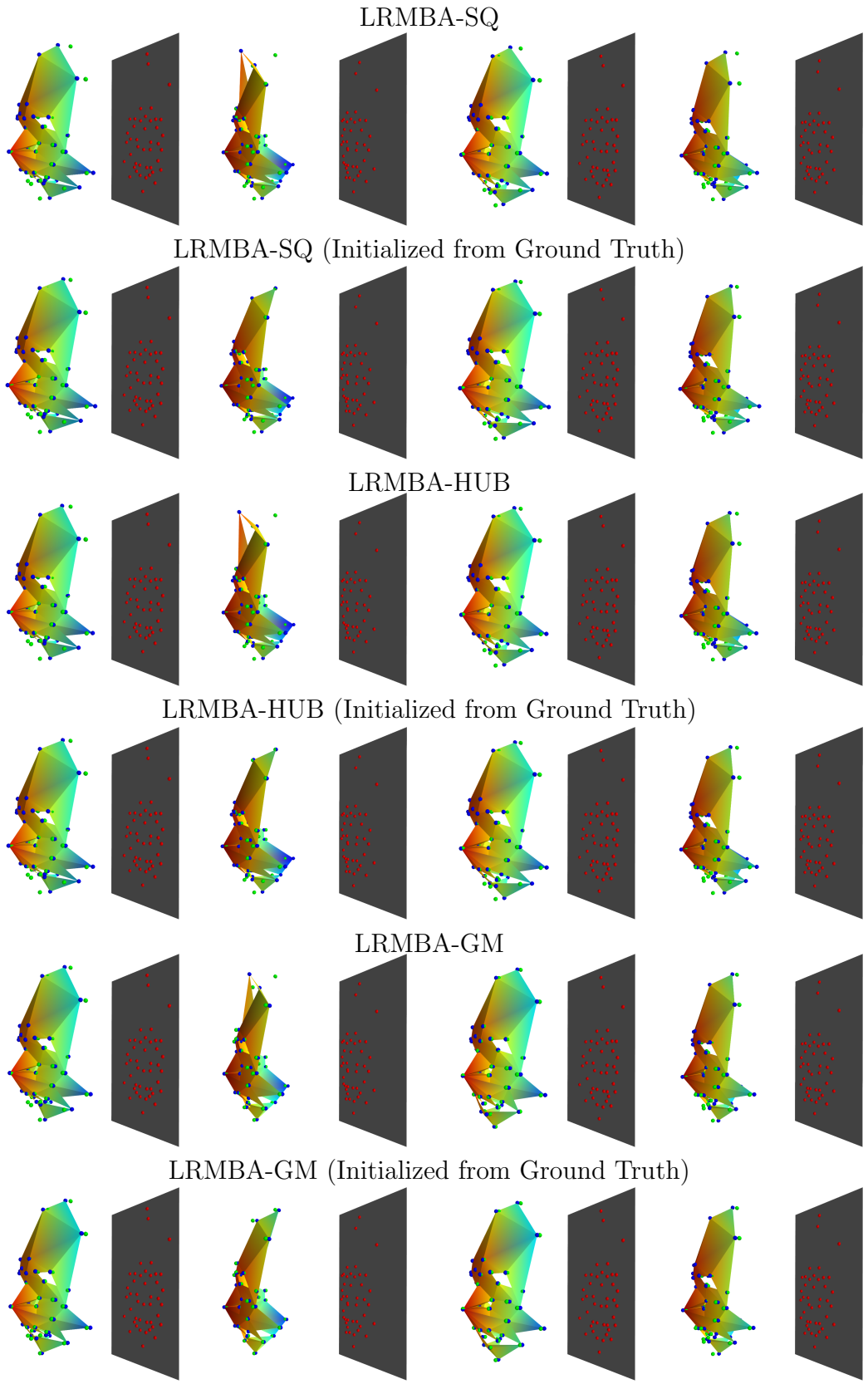

Figure 5.5: 4 frames of JACKY with LRMBA reconstructions (depth colored triangles and blue points), ground truth points (in green) and image points (in red).

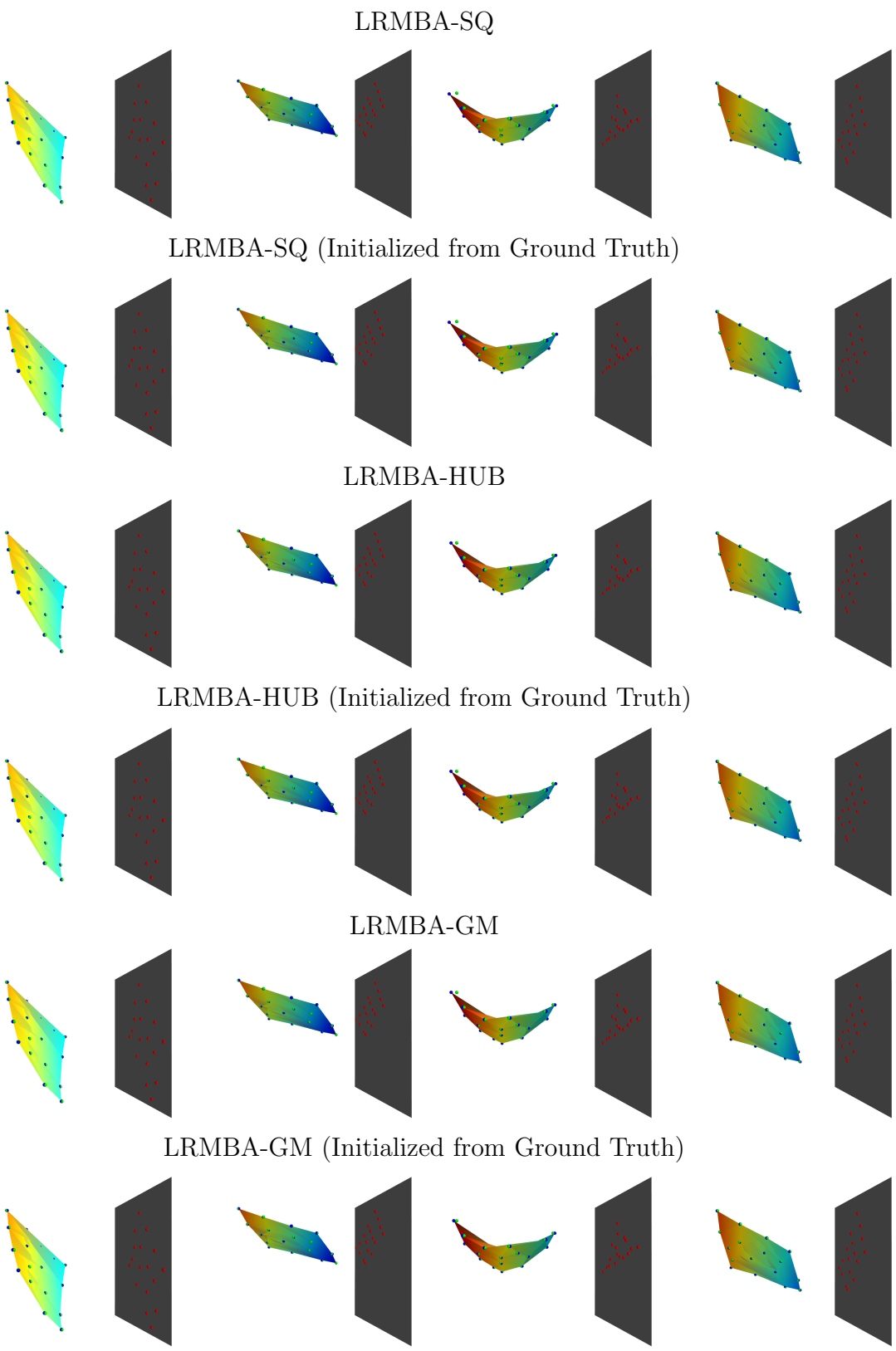

Figure 5.6: 4 frames of `WIND` with LRMBA reconstructions (depth colored triangles and blue points), ground truth points (in green) and image points (in red).

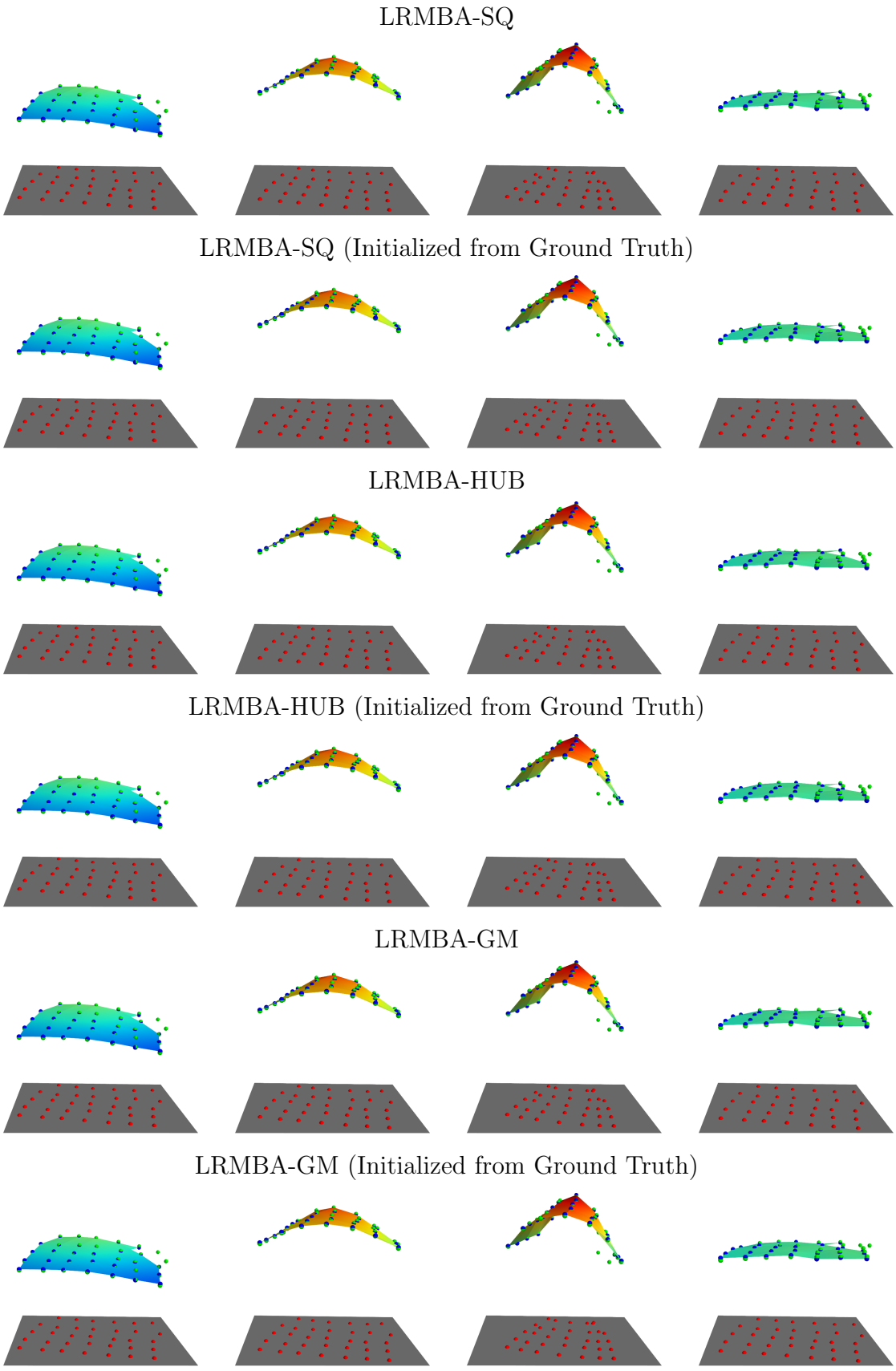

Figure 5.7: 4 frames of `BEND` with LRMBA reconstructions (depth colored triangles and blue points), ground truth points (in green) and image points (in red).

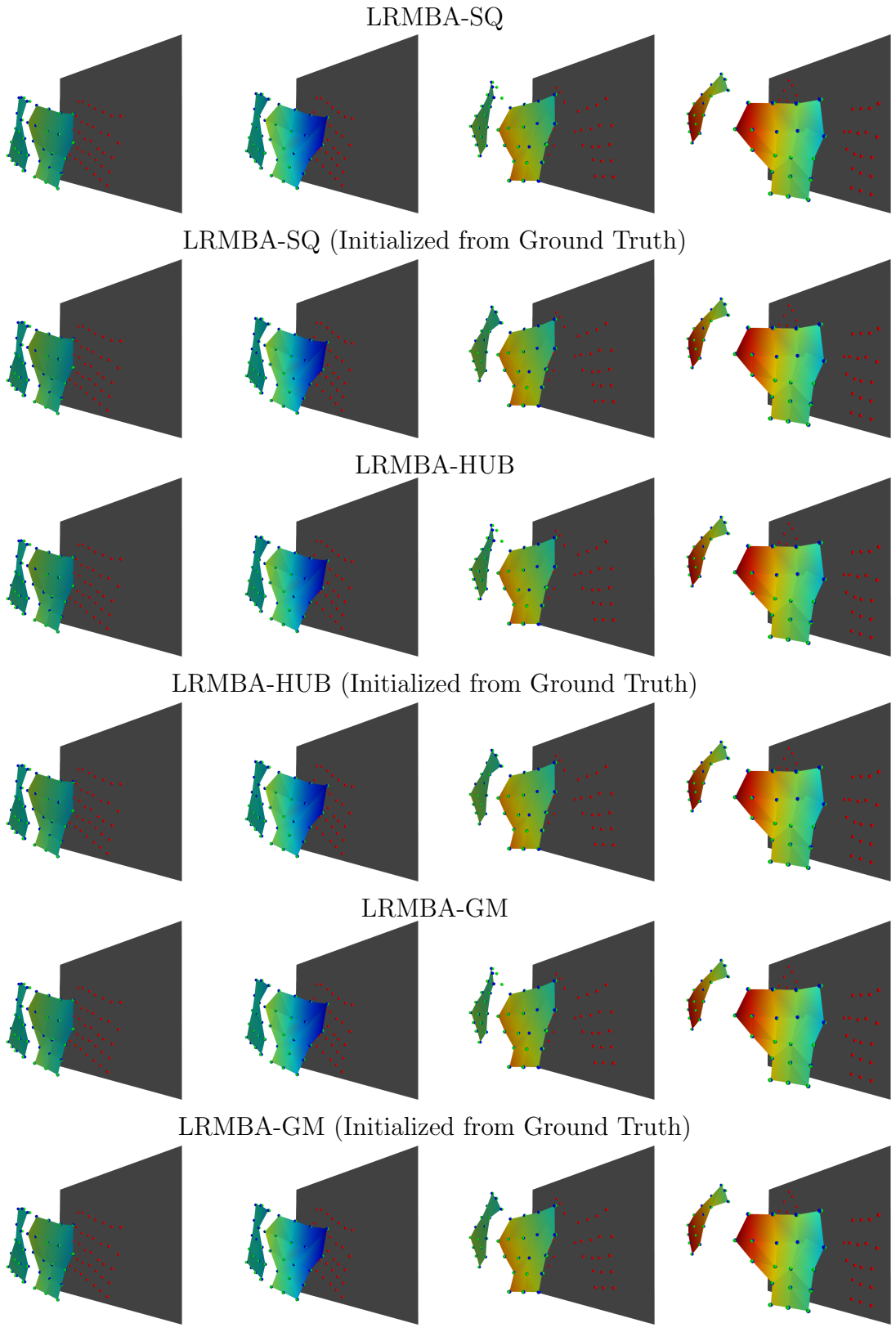

Figure 5.8: 4 frames of `RIP` with LRMBA reconstructions (depth colored triangles and blue points), ground truth points (in green) and image points (in red).

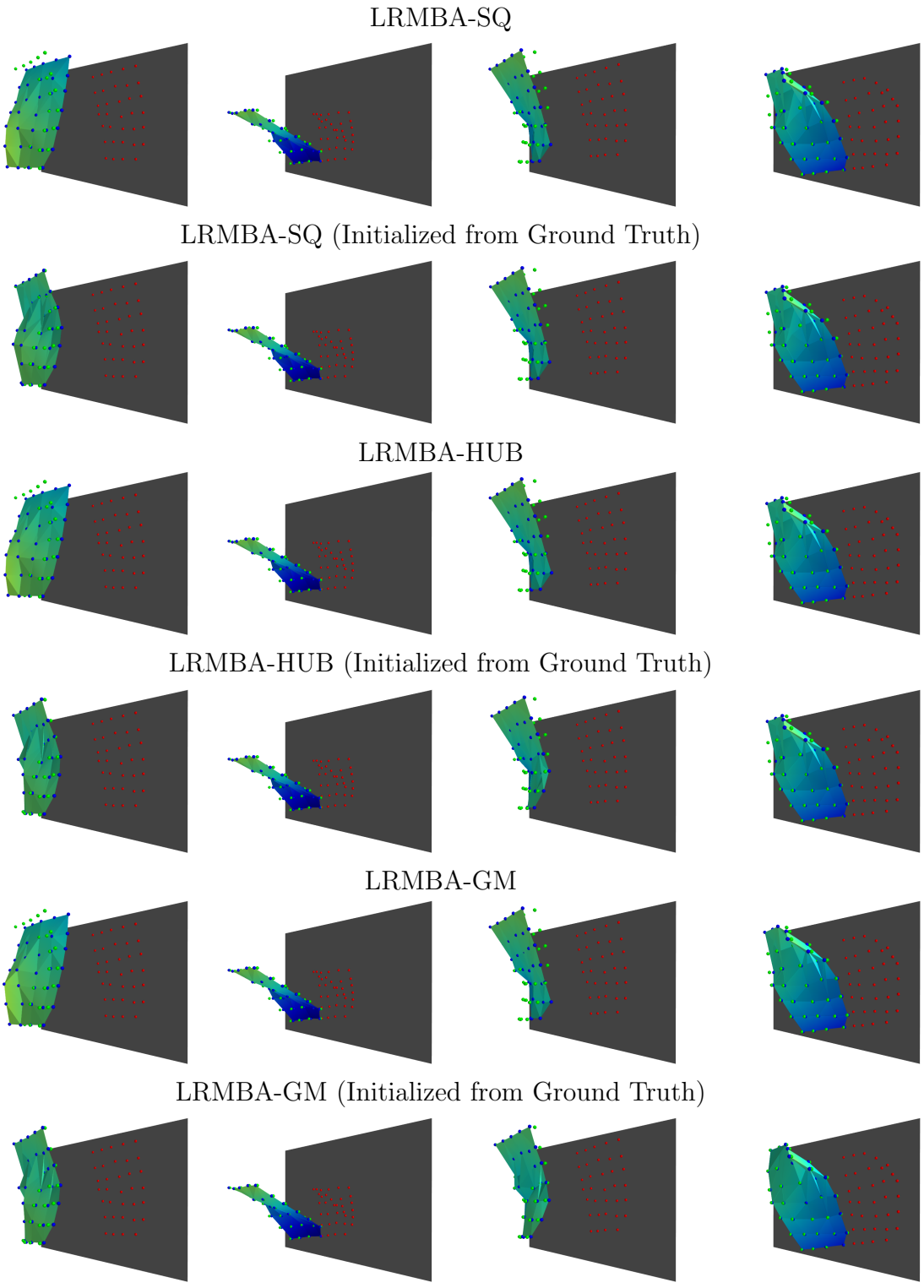

Figure 5.9: 4 frames of `CLOTH` with LRMBA reconstructions (depth colored triangles and blue points), ground truth points (in green) and image points (in red).

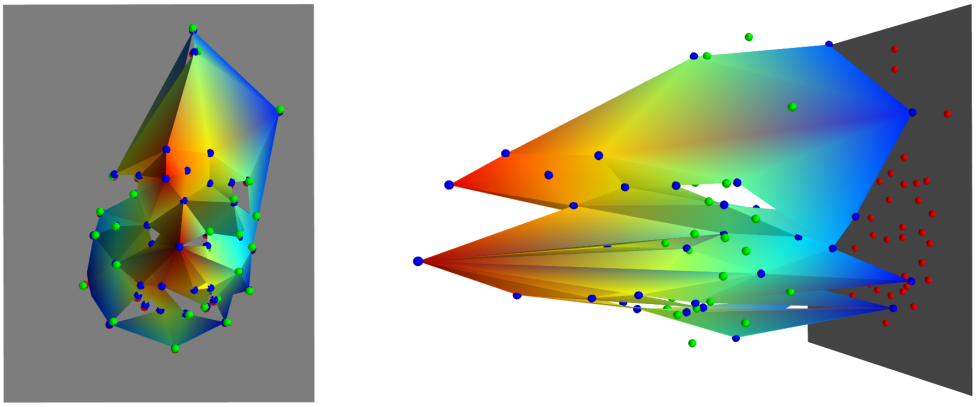

Figure 5.10: Head on and Side View of the LRMBA-SQ reconstruction (depth colored triangles and blue points) of JACKY with substantial Gaussian noise added to the image observations (red points). The ground truth points are in green. The lack of viewpoint variation, allows the optimizer to chase reprojection error (caused by the added noise) without substantially violating the isometric constraints by extending the structure into depth.

solution to begin with.

The most significant change is in the HEAD sequence (see Figure 5.13), where the bundle adjustment methods employing a robust isometric penalty have managed to correct the crease in the actor's upper left arm and chest. In the SCARF sequence (see Figure 5.12), although the overall shape that the LRM reconstruction is correct, there were also many creases. In the corresponding bundle adjusted reconstructions, these creases have been corrected resulting in a much smoother surface and resulting motion. In the PAPER sequence (see Figure 5.16), the LRM reconstruction struggled to resolve the fronto-parallel edges in the center of the paper, as evident by the dent there in the later frames. The bundle adjustment has managed to pop this dent outwards to give smoother and more realistic reconstructions. The perspective effects also become more pronounced in the earlier frames of the LRMBA reconstructions, as LRM's averaging of its (noisy) vertices actually gives the illusion that it realized that these frames were nearly planar. In contrast, the bulging LRMBA reconstructions of PAPER actually displays more optimal results (i.e., lower energies) under the invalid orthographic assumptions. In the TWO CLOTHS (Figure 5.14) and TEAR (Figure 5.15) sequences, there are a few slight corrections made by bundle adjustment. Again, this is in some ways confirmation that LRM is doing quite well. Naturally, the optimization could be just stuck in a local minima but the ground truth results seem to indicate that this local minima is often quite good. It would, on the other hand, be much more concerning if the reconstructions got worse when further optimized with bundle adjustment.

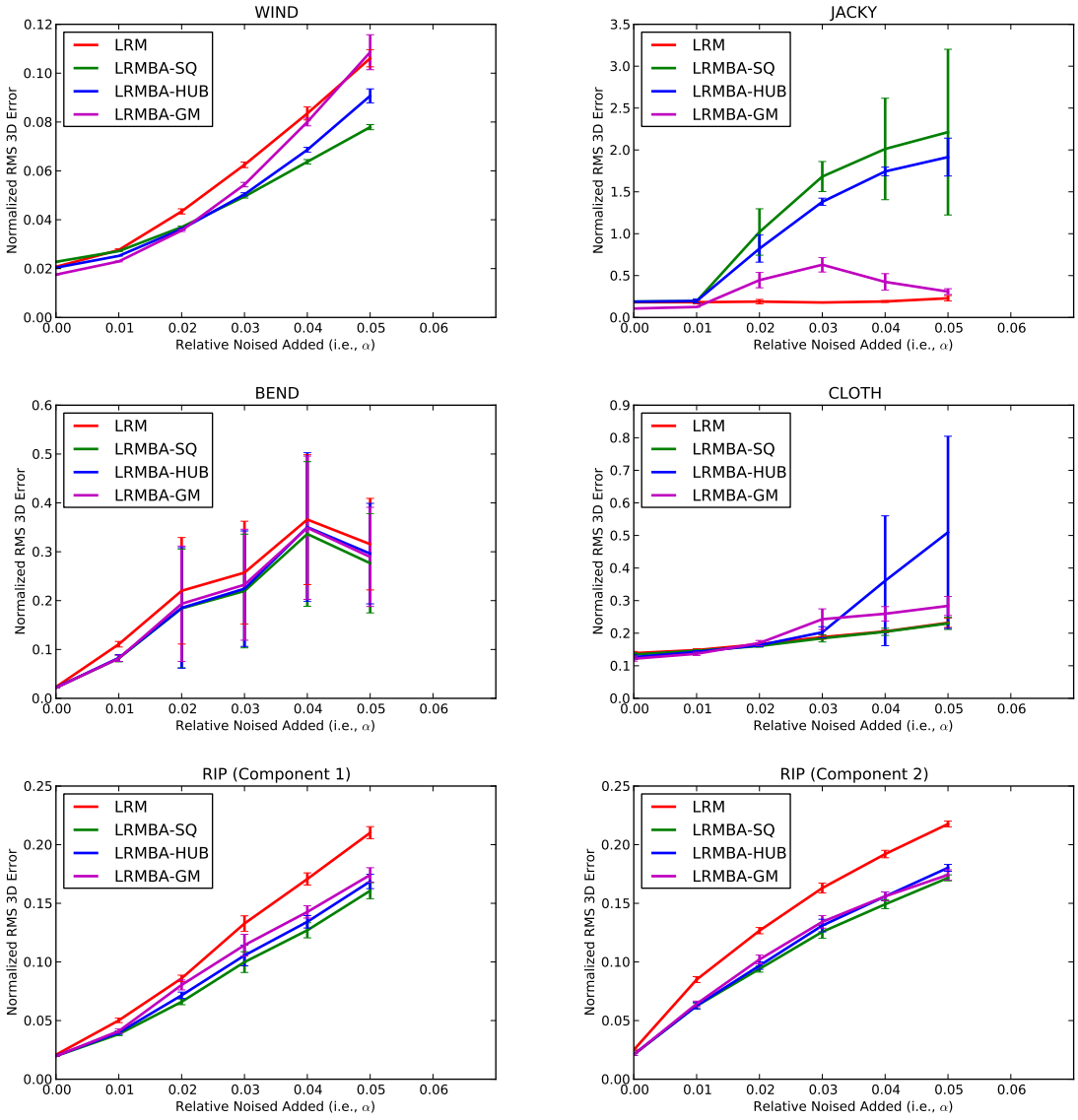

Figure 5.11: Effect of Gaussian noise in the observed image trajectories, on normalized 3D error. The setting roughly corresponds to the fraction of noise added, relative to the scale of the dataset.

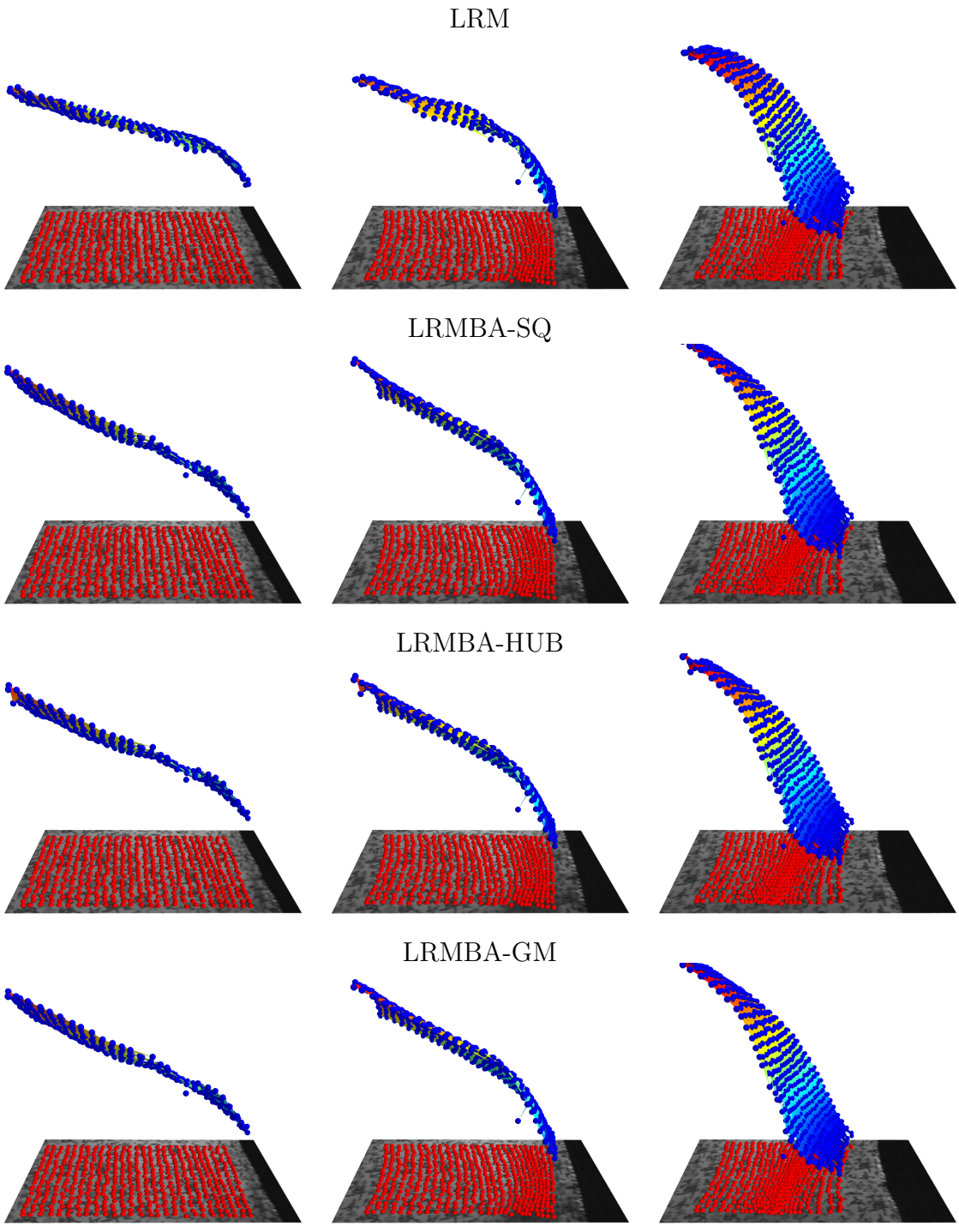

Figure 5.12: 3 frames of SCARF with LRMBA reconstructions (depth colored triangles and blue points) and image points (in red). The LRMBA reconstructions extend much further in depth, as a result of the many creases in the LRM reconstruction being smoothed.

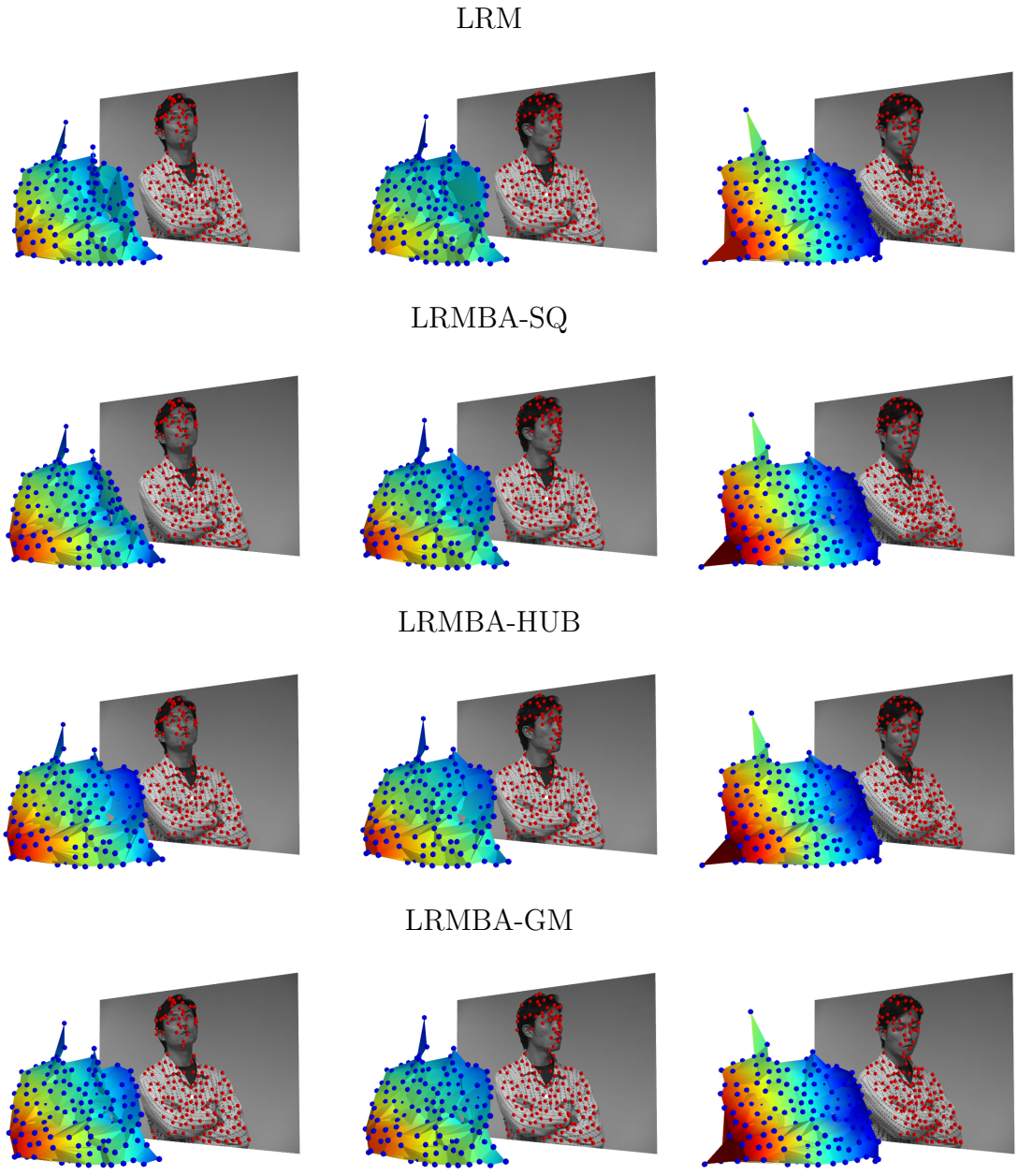

Figure 5.13: 3 frames of `HEAD` with LRMBA reconstructions (depth colored triangles and blue points) and image points (in red). The bundle adjustments have managed to correct the creases in the reconstruction of his upper left arm.

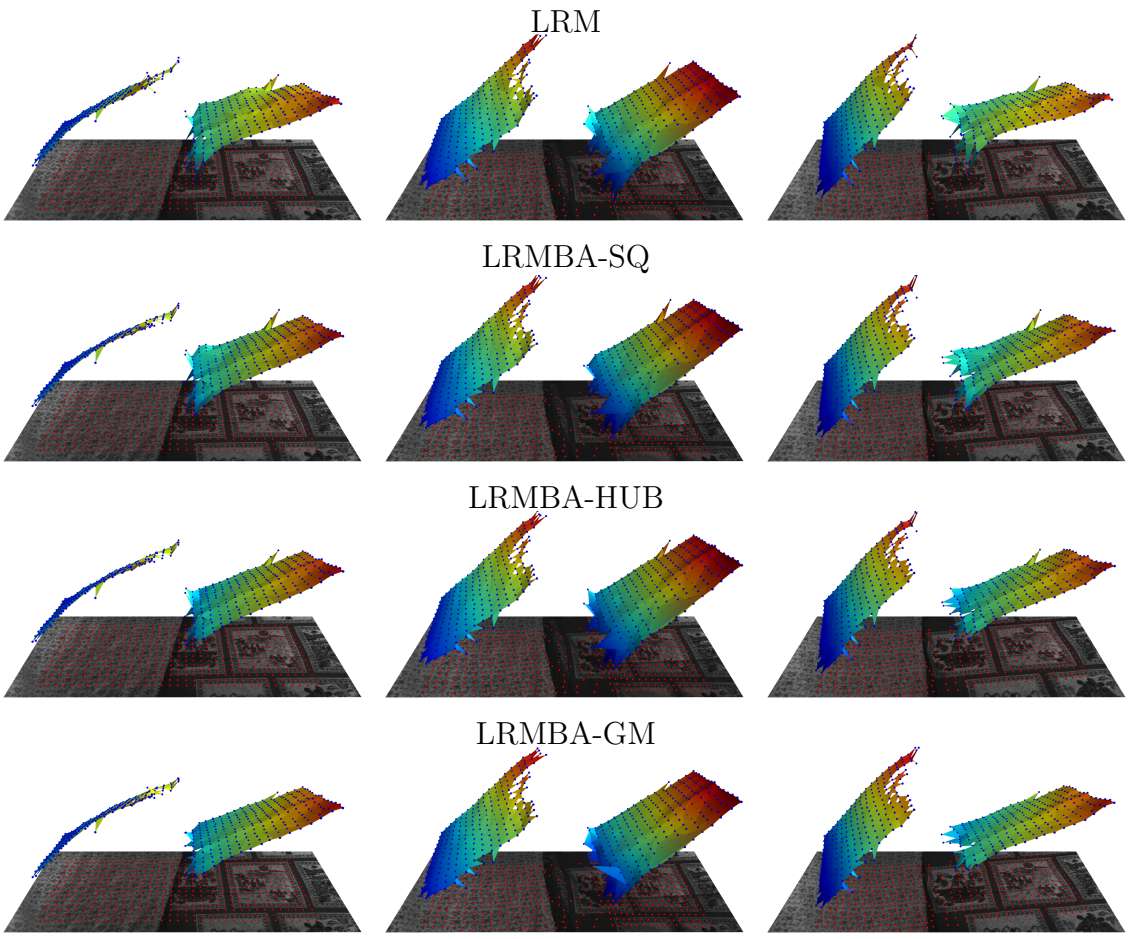

Figure 5.14: 3 frames of `TWOCLOTHS` with LRMBA reconstructions (depth colored triangles and blue points) and image points (in red).

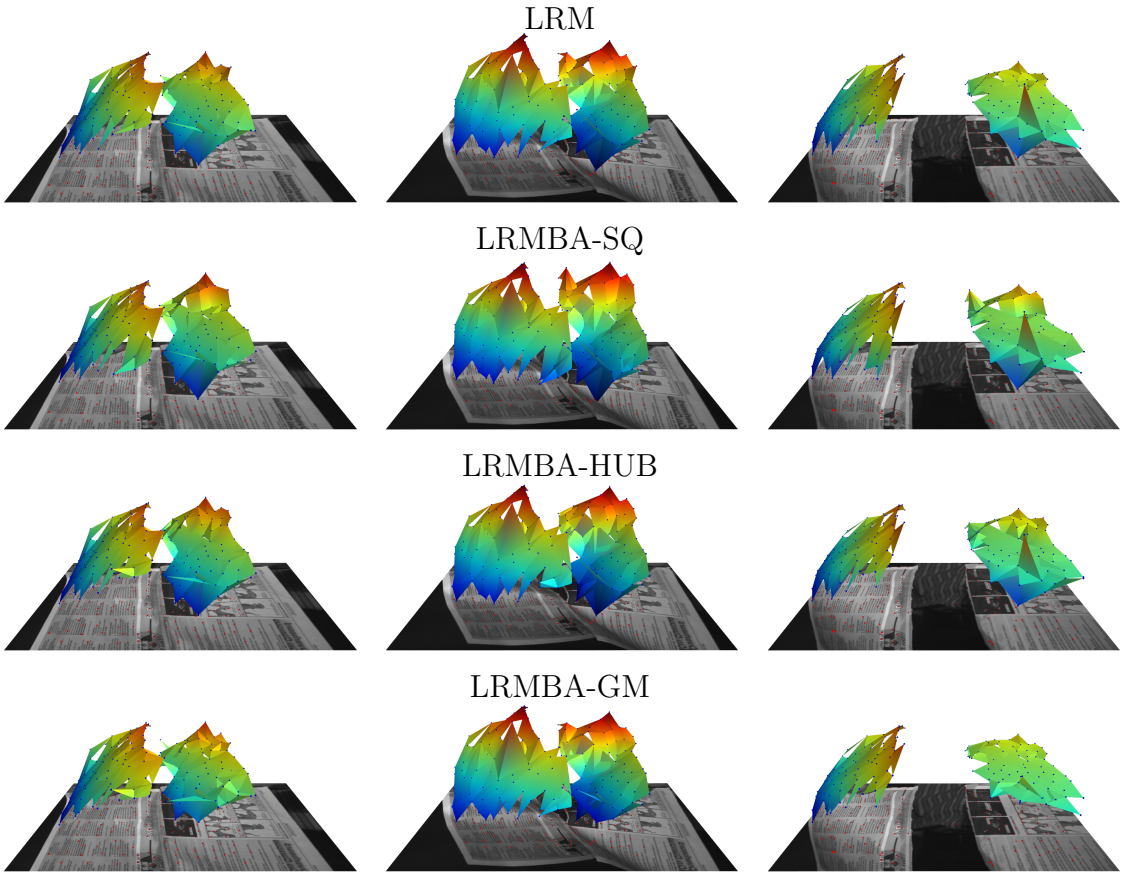

Figure 5.15: 3 frames of `TEAR` with LRMBA reconstructions (depth colored triangles and blue points) and image points (in red).

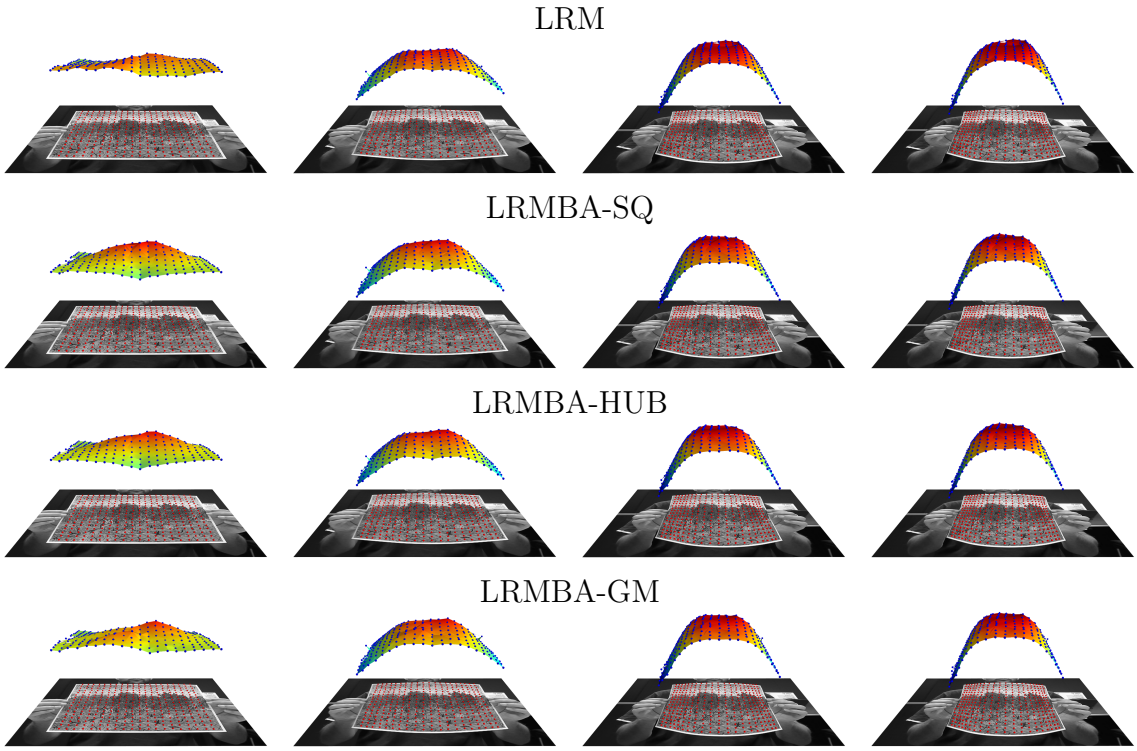

Figure 5.16: 4 frames of PAPER with LRMBA reconstructions (depth colored triangles and blue points) and image points (in red).

5.4 Conclusion

In this chapter we have formulated a global point cloud model regularized by a set of pairwise isometric constraints. We have also demonstrated four important things concerning this model and its relationship with LRM from Chapter 4:

1. This is a good model for the diverse set of sequences that we have considered here. This is most strongly evident by the reconstructions we obtain when we initialize LRMBA from ground truth (see left of Figures 5.3 and 5.4). Further qualitative evidence includes the reconstructions obtained on both ground truth and real sequences when the optimization is initialized using LRM.

2. LRM itself is actually extremely effective at initializing this model. This is evident by how close the local minima that LRMBA finds are to the local minima found when LRMBA is initialized from ground truth (see left of Figures 5.3 and 5.4).

3. In cases when LRM has accurately identified enough isometric constraints, this bundle adjustment can provide a modest increase in performance and robustness by allowing errors in the image observations to be averaged (see Figure 5.11).

4. Even when a scene requires the additional modelling capacity provided by robust error functions to represent a degree of local non-rigidity, LRM can provide an initialization of high enough quality to allow the LRMBA model to be fit. This is evident by the ability of the GM error function, to introduce non-rigidity into the `JACKY` sequence (see Figure 5.5).

A final point to make about this model, is that it is perhaps a much more natural place to start than the LRM model of Chapter 4. It is quite plausible to think of encoding isometric constraints into an energy function as has been done here, without even considering local rigidity leveraged through a piecewise structure made up of rigid triangle models. Indeed, this is what was done in [60], who provide another alternative to attacking this hard optimization problem. It would of course be interesting for future work to consider hybrid approaches or other more global optimizers.

Chapter 6

Conclusion

In this thesis, we have shown how a particular assumption, *local rigidity*, can be leveraged to attack the NRSFM problem. This assumption, that complex global deformations can be approximated locally by rigid motion, appears consistent with human intuition in many situations. For example, in our opening example (see Figure 1.1), we argued that the global bending of the paper could be modelled locally, around any point, by the rigid motion of its tangent plane. As this thesis has demonstrated through the last two chapters, the local rigidity assumption is not only applicable in such situations where human intuition suggests, but also over a diverse set of sequences. Further, we have demonstrated how this assumption can be formalized into two global models. One is a piecewise model of loosely connected rigid triangles and the other is a global deforming point cloud model regularized by a set of pairwise isometric constraints. Furthermore, we have demonstrated that it is actually possible to fit both of these models to noisy image trajectories.

In Chapter 3, we formulated our three point rigid triangle model and characterized its per-frame depth flip ambiguities. We presented three point rigid structure from motion procedures to fit this model in both the noiseless and noisy case. In the noiseless case, our simple solution recovers the single link length solution with just 4 frames and up to two such solutions with just 3 frames. In the presence of noise, we presented a bundle-adjustment like method for fitting the rigid triangle model and demonstrated empirically its robustness under a generic set of viewpoints. We further demonstrated how a simple prior can deal with viewpoint degeneracies, present in common sequences, that leave the solution otherwise underconstrained.

This rigid triangle model is the key ingredient in our piecewise model of global scene deformation presented in Chapter 4. To fit this model, we followed a bottom up reconstruction approach. This began first with a hypothesis and test framework to find triplets

of image trajectories that could be explained by the motion of the vertices of a rigid triangle. For the NRSFM problem we used 3-SFM, but when a planar template model is available this is reduced to solving exterior orientation. The per-frame depth flips of the triangles were resolved by performing inference in a non submodular binary MRF. When this MRF is planar, as was the case in our template based approach, tractable exact methods are available. A global piecewise model was then formed through a final step that resolved each triangle model's depth using linear least squares.

Finally, in Chapter 5, we formulated our deforming point cloud model and regularized it using a set of pairwise isometric constraints. We provided a tractable algorithm for fitting this model through local gradient based optimization, initialized from our piecewise model. We demonstrated that using different error functions to penalize violations of the isometric constraints can lead to interesting properties. Sometimes such functions can allow upstream mistakes, such as incorrect flip inference, to be corrected or allow a degree of local non-rigidity to be admitted by the model.

Perhaps the most interesting result of this thesis, stems from the premise that optimizing either of these global models is extremely difficult. If a model instance is assigned some function of fitness through an energy function, as was done explicitly in Chapter 5, one will encounter a non-convex energy landscape with a plethora of local minimum. The key to our success here is the decomposition of this hard optimization problem into many easy ones. To see this strategy in action, one should consider the sequence of steps in the entire procedure for fitting our final global point cloud model. Each step was formulated and solved as a relatively simple sub-problem:

1. Fit Local Rigid Triangle Models
    - Linear Least Squares
    - Local Optimization
2. Test for Rigidity
    - Trivial Calculations
3. Resolve Depth Flips
    - Approximate (Optimal) Inference in a Non-Planar (Planar) Binary MRF
4. Resolve Depth Translations
    - Linear Least Squares

5. Fit Point Cloud Model

    - Local Optimization

An obvious avenue for future work would be to consider trying to extract more information from these local three point models. The ability of these models to probe for rigidity in a sequence is very compelling but is tempered by the pervasive viewpoint degeneracy in typical sequences which sometimes results in false positives. Unfortunately, the emergence of such false positives appears to result from a very complex relationship between the degree of non-rigidity of a three point configuration, that configuration's geometry, and the variation in viewpoints in which that configuration is observed. One way to approach this would be to try to examine the local energy landscape near a solution to determine whether the minimum is well defined. A different approach might include building an appearance model for the projected interior of a triangle, that could be checked for consistency across frames.

Another interesting avenue of work would be to increase the flexibility of the model to include an isotropic scaling of the triangle. Interestingly, this is mathematically equivalent to considering a scaled orthographic projection model. This means, however, that we would have $6F + 3$ parameters to estimate with only $6F$ observations. If the trajectories come from a video sequence, a natural way to attack this problem would be to incorporate a strong prior to penalize scale parameters that do not vary smoothly in time.

Another brittle component of this overall approach is the flip resolution stage used in fitting our piecewise model. If enough flips can be inferred correctly then there is a chance that the downstream bundle adjustment can correct the others. Naturally, we would then like to perform more accurate inference in large non-planar MRFs. It appears that our greedy method does quite well in minimizing the energy, however, there is still considerable value in methods that allow more constraints to be considered. For example, if two sets of flip variables are connected only by a weak (i.e., both potentials are nearly the same) set of noisy constraints, the greedy algorithm is likely to choose the relative flip of these two sets based on only one of these noisy constraints. The availability of tractable exact methods for planar MRFs is thus quite compelling, as a MRF defined on a planar subgraph can include all of these constraints. When inference is performed, there will then be an increased chance of choosing the correct flip, as it will be determined by integrating over all of the noisy constraints instead of just one. Therefore, one could consider approximation algorithms for finding maximum weight planar subgraphs [15].

Unfortunately even if we can do optimal inference, the correlation between low energies and good reconstructions is not perfect. We might try to improve this situation

by reformulating the energy to make it more representative of how good a flip configuration truly is. To do this, we might consider higher order potentials to encode the interaction of small cliques of flip variables, instead of just pairs. Another option, might be to include a third, outlier label in order to allow the energy to pay a small penalty to remove non-integrable triangle models that would otherwise drastically increase the energy. Naturally, both of these options would considerably complicate inference and thus new methods would have to be considered.

Lastly, we might consider modifications to our global model and its fitting procedure in order to obtain better results. It was quite noticeable in the reconstructions of the `PAPER` sequence, that the orthographic projection assumption is limiting and causes problems when applied to motion undergoing pronounced perspective effects. It would be natural to parametrize a perspective projection model, which could be directly optimized in our bundle adjustment framework. In sequences such as `PAPER`, this might allow us to both obtain a cleaner reconstruction and disambiguate the global flip by comparing the two possible energies. Even with such realistic models, however, our local optimization procedure is limited and thus it would be natural to consider a hybrid approach integrating the fusion moves advocated in [60] or, alternatively, stochastic local search methods [33].

Finally, to conclude this thesis, we briefly summarize the main contributions of this thesis:

- A linear algorithm for recovering, from the orthographic projections of three rigid points in four (or three) views, the single set (or up to two sets for $F = 3$) of possible interpoint distances such a model admits.
- A regularized bundle-adjustment like procedure for fitting a three point rigid model to a triplet of image trajectories spanning an arbitrary number of views.
- The use of this procedure to *probe* a scene for local plausibly rigid three point configurations.
- A bottom up procedure that uses pairwise constraints between such models to resolve their local depth flip and depth translation ambiguities, resulting in a piecewise local model.
- An analogous procedure for fitting such a model to a single frame, when a planar template shape is available. Despite being seemingly tangent to the structure from motion work in this document, this contribution falls out naturally and elegantly as an aside.

- The formulation of a global point based model incorporating a set of weak pairwise isometric constraints and a strategy for fitting this model.
- Experimental results and comparisons with other methods on several ground truth (or ground truth with added noise) datasets.
- Demonstrations of the results given image trajectories from image motion estimation and tracking data.

Bibliography

[1] Henrik Aanaes and Fredrik Kahl. Estimation of deformable structure and motion. In *Workshop on Vision and Modelling of Dynamic Scenes, ECCV2002*, 2002.

[2] Ijaz Akhter, Yaser Sheikh, and Sohaib Khan. In defense of orthonormality constraints for nonrigid structure from motion. In *CVPR*, pages 1534–1541, 2009.

[3] Ijaz Akhter, Yaser Sheikh, Sohaib Khan, and Takeo Kanade. Trajectory space: A dual representation for nonrigid structure from motion. *IEEE PAMI*, 33:1442–1456, July 2011.

[4] Ijaz Akhter, Yaser Ajmal Sheikh, Sohaib Khan, and Takeo Kanade. Nonrigid structure from motion in trajectory space. In *NIPS*, December 2008.

[5] Adrien Bartoli, Vincent Gay-Bellile, Umberto Castellani, Julien Peyras, Sren I. Olsen, and Patrick Sayd. Coarse-to-fine low-rank structure-from-motion. In *CVPR*, 2008.

[6] Adrien Bartoli, Yan Grard, Franois Chadebecq, and Toby Collinsd. On template-based reconstruction from a single view: Analytical solutions and proofs of well-posedness for developable, isometric and conformal surfaces. In *CVPR*, 2012.

[7] B. M. Bennett and D. D. Hoffman. Inferring 3d structure from three points in rigid motion. *Journal of Mathematical Imaging and Vision*, 4:4–401, 1994.

[8] M. Black and A Rangarajan. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *IJCV*, 1996.

[9] Matthew Brand. Morphable 3D models from video. In *CVPR*, pages 456–463, 2001.

[10] Matthew Brand. A direct method for 3D factorization of nonrigid motion observed in 2d. In *CVPR*, pages 122–128, 2005.

[11] C Bregler, A Hertzmann, and H Biermann. Recovering non-rigid 3D shape from image streams. In *CVPR*, 2000.

[12] Richard H Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.

[13] Toby Collins and Adrien Bartoli. Locally planar and affine deformable surface reconstruction from video. In *VMV Workshop*, pages 339–346, 2010.

[14] J. Costeira and T. Kanade. A multi-body factorization method for motion analysis. In *ICCV*, 1995.

[15] Gruia Călinescu, Cristina G. Fernandes, Ulrich Finkler, and Howard Karloff. A better approximation algorithm for finding planar subgraphs. In *Proceedings of the seventh annual ACM-SIAM symposium on Discrete algorithms*, SODA '96, pages 16–25, Philadelphia, PA, USA, 1996. Society for Industrial and Applied Mathematics.

[16] Yuchao Dai, Hongdong Li, and Mingyi He. A simple prior-free method for non-rigid structure-from-motion factorization. In *CVPR*, 2012.

[17] Mark de Berg, Otfried Cheong, Marc van Kreveld, and Mark Overmars. *Computational Geometry: Algorithms and Applications*. Springer, 2008.

[18] Joao Fayad, Lourdes Agapito, and Alessio Del Bue. Piecewise quadratic reconstruction of non-rigid surfaces from monocular sequences. In *ECCV*, 2010.

[19] Joao Fayad, Alessio Del Bue, Lourdes Agapito, and Pedro M.Q. Aguiar. Non-rigid structure from motion using quadratic deformation models. In *BMVC*, 2009.

[20] P. Gotardo and A.M. Martinez. Kernel non-rigid structure from motion. In *ICCV*, 2011.

[21] P. Gotardo and A.M. Martinez. Non-rigid structure from motion with complementary rank-3 spaces. In *CVPR*, 2011.

[22] Paulo F. U. Gotardo and Aleix M. Martinez. Computing smooth time trajectories for camera and deformable shape in structure from motion with occlusion. *PAMI*, 33:2051–2065, October 2011.

[23] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[24] T. S. Huang and C. H. Lee. Motion and structure from orthographic projections. *PAMI*, 1989.

[25] Peter Huber. Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35:73–101, 1964.

[26] H. Ishikawa. Higher-order clique reduction in binary graph cut. In *CVPR*, 2009.

[27] H. Ishikawa. Higher-order gradient descent by fusion-move graph cut. In *ICCV*, 2009.

[28] Hiroshi Ishikawa. Transformation of general binary mrf minimization to the first order case. *PAMI*, 2011.

[29] A.D. Jepson, D.J. Fleet, and T.F. El-Maraghi. Robust on-line appearance models for visual tracking. *PAMI*, 2003.

[30] Gunnar Johannsson. Visual motion perception. *Scientific American*, 1975.

[31] Gunnar Johansson and Gunnar Jansson. Perceived rotary motion from changes in a straight line. *Perception & Psychophysics*, 4:165–170, 1968.

[32] Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923, 1976.

[33] J. Kennedy and R. Eberhart. Particle swarm optimization. In *International Conference on Neural Networks*, 1995.

[34] Vladimir Kolmogorov and Ramin Zabih. What energy functions can be minimized via graph cuts? In *PAMI*, 2004.

[35] V. Lempitsky, C. Rother, S. Roth, and A.ion Blake. Fusion moves for Markov random field optimization. *PAMI*, 32(8):1392–1405, aug 2010.

[36] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *International Joint Conference on Artificial Intelligence*, 1981.

[37] S. Olsen and A. Bartoli. Using priors for improving generalization in non-rigid structure-from-motion. In *BMVC*, 2007.

[38] M. Paladini, A. Del Bue, M. Stošic, M. Dodig, J. Xavier, and L. Agapito. Factorization for non-rigid and articulated structure using metric projections. In *CVPR*, 2009.

[39] Mathieu Perriollat, Richard Hartley, and Adrien Bartoli. Monocular template-based reconstruction of inextensible surfaces. In *BMVC*, 2008.

[40] Vincent Rabaud and Serge Belongie. Re-thinking non-rigid structure from motion. In *CVPR*, 2008.

[41] Vincent Rabaud and Serge Belongie. Linear embeddings in non-rigid structure from motion. In *CVPR*, 2009.

[42] Carsten Rother, Vladimir Kolmogorov, Victor Lempitsky, and Martin Szummer. Optimizing binary MRFs via extended roof duality. In *CVPR*, 2007.

[43] Chris Russell, Joao Fayad, and Lourdes Agapito. Energy based multiple model fitting for non-rigid structure from motion. In *CVPR*, 2011.

[44] M. Salzmann, R. Hartley, and P. Fua. Convex optimization for deformable surface 3-d tracking. In *ICCV*, 2007.

[45] Mathieu Salzmann and Pascal Fua. Reconstructing sharply folding surfaces: A convex formulation. In *CVPR*, pages 1054–1061, 2009.

[46] Mathieu Salzmann, Julien Pilet, Slobodan Ilic, and Pascal Fua. Surface deformation models for nonrigid 3D shape recovery. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29:1481–1487, August 2007.

[47] Mathieu Salzmann, Raquel Urtasun, and Pascal Fua. Local deformation models for monocular 3D shape recovery. In *CVPR*, 2008.

[48] Nicol N. Schraudolph and Dmitry Kamenetsky. Efficient exact inference in planar Ising models. In *NIPS*, volume 21, Cambridge, MA, 2009. MIT Press.

[49] Jonathan Taylor, Allan D. Jepson, and Kiriakos N. Kutulakos. Non-rigid structure from locally-rigid motion. In *CVPR*, 2010.

[50] Yuandong Tian and Srinivasa G. Narasimhan. A globally optimal data-driven approach for image distortion estimation. In *CVPR*, 2010.

[51] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography: A factorization method. *IJCV*, 9:137–154, 1992.

[52] Lorenzo Torresani, Aaron Hertzmann, and Chris Bregler. Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. *PAMI*, 30:878–892, May 2008.

[53] Lorenzo Torresani, Aaron Hertzmann, and Christoph Bregler. Learning non-rigid 3D shape from 2d motion. In *NIPS*, 2003.

[54] Lorenzo Torresani, Danny B. Yang, Eugene J. Alexander, and Christoph Bregler. Tracking and modeling non-rigid objects with rank constraints. In *CVPR*, pages 493–500, 2001.

[55] Bill Triggs, Philip Mclauchlan, Richard Hartley, and Andrew Fitzgibbon. Bundle adjustment - a modern synthesis. *Vision Algorithms: Theory and Practics, LNCS*, 2000.

[56] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *CVPR*, 2007.

[57] S. Ullman. The interpretation of structure from motion. *Proceedings of the Royal Society of London. Series B, Containing papers of a Biological character. Royal Society (Great Britain)*, 203(1153):405–426, January 1979.

[58] Shimon Ullman. Maximizing rigidity: The incremental recovery of 3-d structure from rigid and rubbery motion. *Perception*, 13:255–274, 1983.

[59] A. Varol, M. Salzmann, E.Tola, and P. Fua. Template-free monocular reconstruction of deformable surfaces. In *IEEE International Conference on Computer Vision*, Kyoto, Japan, 2009.

[60] Sara Vicente and Lourdes Agapito. Soft inextensibility constraints for template-free non-rigid reconstruction. In *ECCV*, 2012.

[61] R Vidal, R Tron, and R Hartley. Multiframe motion segmentation with missing data using power factorization and GPCA. *IJCV*, 79:85–105, 2008.

[62] Jing Xiao, Jinxiang Chai, and Takeo Kanade. A closed-form solution to non-rigid shape and motion recovery. In *ECCV*, 2004.

[63] Jingyu Yan and Marc Pollefeys. A factorization-based approach to articulated motion recovery. In *CVPR*, pages 815–821, 2005.

[64] Jingyu Yan and Marc Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In *ECCV*, pages 94–106, 2006.