Local Non-Rigid Structure-from-Motion from Diffeomorphic Mappings

Shaifali Parashar, Mathieu Salzmann, Pascal Fua

Local Non-Rigid Structure-from-Motion from Diffeomorphic Mappings

Shaifali Parashar, Mathieu Salzmann and Pascal Fua
CVLAB, EPFL, Switzerland
shaifali.parashar,mathieu.salzmann,pascal.fua@epfl.ch

Abstract

We propose a new formulation to non-rigid structure-from-motion that only requires the deforming surface to preserve its differential structure. This is a much weaker assumption than the traditional ones of isometry or conformality. We show that it is nevertheless sufficient to establish local correspondences between the surface in two different images and therefore to perform point-wise reconstruction using only first-order derivatives. To this end, we formulate differential constraints and solve them algebraically using the theory of resultants. We will demonstrate that our approach is more widely applicable, more stable in noisy and sparse imaging conditions and much faster than earlier ones, while delivering similar accuracy. The code is available at https://github.com/cvlab-epfl/diff-nrsfm/.

1. Introduction

Reconstructing the 3D shape of deformable objects from monocular images, known as Non-Rigid Structure-from-Motion (NRSfM), has applications in domains ranging from entertainment [23] to medicine [20]. It was introduced in [4] by expressing shapes in terms of a low-rank shape-basis. Many variants of this idea have since been proposed with a view to improve reconstruction stability [5, 2, 32, 11, 21, 14]. Over the last decade, physically-inspired NRSfM models [33, 34, 30, 8, 9, 18, 24, 25] have emerged as an attractive alternative. They exploit local surface properties to draw constraints, can handle large deformations and outperform techniques relying on low-rank priors. Unfortunately, most methods in both categories become prohibitively slow as the number of images increases, due of their non-linear complexity, and cannot handle missing data. This makes them impractical for real-world scenarios.

[24, 25] buck this trend. By expressing isometry or conformality constraints in terms of differential properties, local reconstruction constraints can be established between the deforming surface as seen in two different images. Thus, the surface 3D shape in any frame can be obtained by pairing that frame with the rest and the complexity only grows linearly with the number of images. Furthermore, missing data for example due to occlusions, can be easily handled by using a parametric image registration warp. While effective in theory, this approach suffers from two main drawbacks: *i)* it requires the second-order derivatives of the image registration warps, which are usually noisy, and sometimes even downright wrong when given only semi-dense correspondences to compute them. In the first case, an expensive warp refinement [26] must be performed and in the second the approach simply becomes impractical. *ii)* a deformation model must be chosen *a priori*, which precludes using this method for surfaces of unknown properties. In this paper, we introduce a framework that overcomes these drawbacks. To this end, we leverage the assumption that the deforming surface is locally diffeomorphic, that is, that the deformation preserves the local differential structure of the surface, which is a much more generic model than isometry or conformality and encompasses both as well as equiareality. We will show that it suffices to establish local reconstruction constraints between pairs of surfaces without requiring second-order derivatives or *a priori* knowledge about the surface properties. This makes our approach immune to the difficulties described above. Furthermore, if knowledge about the surface properties is available, the corresponding metric-preserving constraints can be incorporated.

We will show that, when the deformations are equiareal instead of conformal and the correspondences semi-dense, our approach delivers good results whereas [24, 25] cannot be used. Furthermore, in the conformal case with dense correspondences, our approach delivers a similar accuracy to [24, 25] and it is $10\times$ faster. In addition, we require only first order derivatives, which is at least $20\times$ faster than computing second order derivatives for [24, 25]. We also compare with some of the best performing methods in state of the art and show that we outperform most of them in terms of both accuracy and computation time.

2. Related Work

NRSfM methods can be grouped into three broad classes depending on how deformations are modeled.

Low-Dimensional Deformations. These methods [4, 6, 1, 14, 10, 21] produce a global 3D shape by jointly reconstructing the points in all frames. This is an ill-posed problem that is solved by constraining the deformations to lie in a low-dimensional space. This makes these methods ill-suited to model complex deformations and to handle missing correspondences. Furthermore, it usually requires the shape-space dimension to be decided *a priori*.

Global Physical Deformations. These methods [30, 34, 33, 9, 18] aim to preserve physical properties of surfaces. Most of them assume deformations to be isometric (distance-preserving) but they model an approximation of isometry such as inextensibility [9, 18], piece-wise inextensibility [30, 34] or piece-wise rigidity [33]. They usually find a globally optimal solution by solving for constraints over all the points altogether. They usually require a computationally expensive optimization which makes them impractical for handling large number of images.

Local Physical Deformations. Fewer methods only characterize local deformations. These methods formulate and solve isometric constraints locally. [8] formulates isometry as local rigidity and [24, 25] formulate the exact constraint for isometry using differential properties of surfaces. [24, 25] showed that their complexity scales linearly with the number of images, unlike that of the methods discussed above, which grows much faster. This is because they use differential properties that are preserved under isometry up to a change of variables. Due to this, they show that adding images does not increase the number of variables. In practice, [24, 25] yield faster and more accurate reconstructions than existing methods. However, as discussed in the introduction, they rely on second-order derivatives, which are computationally expensive to compute and therefore, impractical. [24, 25] assume that the second-order derivatives are provided with the input. We discuss the problems with obtaining the second-order derivatives in our experiments. Furthermore, they still impose strong constraints on what the surface deformations may be. In this paper, we seek the minimalistic deformations constraints. Given that the surface deformations in nature are at least locally diffeomorphic, we show that it provides sufficient constraints to perform reconstruction. Thus we show that any deformation stronger than local diffeomorphism, (isometry, conformality and equareality), is thus solvable. In addition, we get rid off the second-order derivatives and thus obtain a highly reliable, fast and practical solution for NRSfM, just with the assumption of local diffeomorphism.

3. Method Outline

Fig 1 depicts our setup when using only two images $\mathcal{I}$ and $\overline{\mathcal{I}}$ acquired by a calibrated camera. In each one, we denote the deforming surface as $\mathcal{S}$ and $\overline{\mathcal{S}}$, respectively, and

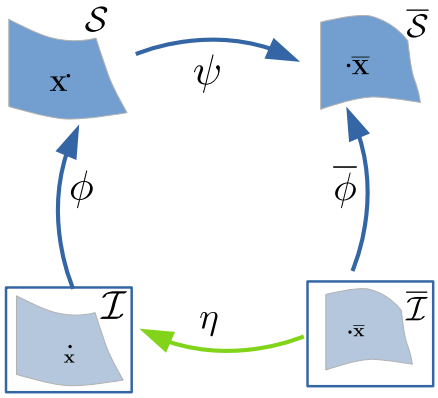


Figure 1: A 2-view model for Diff-NRSfM. Assuming ψ to be locally diffeomorphic, our goal is to find ϕ, $\overline{\phi}$ given that η is known.

model it in terms of functions $\phi, \overline{\phi} : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ that associate a surface point to an image point. Let us assume that we are given an image registration function $\eta : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ that associates points in the first image to points in the second. In practice, it can be computed using standard image matching techniques such as optical flow [29, 28] or SIFT [22].

These functions can be composed to create a mapping $\psi : \mathbb{R}^3 \rightarrow \mathbb{R}^3$, which we assume to be locally diffeomorphic, from 3D surface points seen in the two images. We use a parametric representation of η and ϕ using a spline [3], which allows us to accurately obtain first-order derivatives of these functions. Any other approach such as finite difference methods, can be used alternatively.

At the heart of our approach is the fact that, under the assumption that the two surfaces are locally diffeomorphic, some differential properties of corresponding 3D points should match. These properties can be expressed in terms of *connections*. They are generic properties of a differentiable surface that express intrinsic relationship between a point on the surface and its local neighborhood [7, 16]. In particular, the well known first and second fundamental forms on surfaces can be derived from them. Crucially, they are preserved under diffeomorphism [25], which we prove formally. Furthermore, assuming the surfaces to be locally planar, we show that we can use connections computed using *only* first derivatives. Thus, we can express depth and its derivatives at $\overline{\mathcal{S}}$ in terms of the same quantities at $\mathcal{S}$. As a consequence, the 3D coordinates of corresponding points on the surface are strongly constrained, thanks to multi-view constraints and the 3D reconstruction problem becomes sufficiently constrained. This approach has several strengths:

- Because all the constraints can be expressed in terms of first derivatives of ϕ, $\overline{\phi}$ and η, which, unlike the second derivatives that are required by the formulation of [25],

can be estimated even if the points for which we have correspondences are relatively sparse.

- If we happen to know that the deformation is isometric, conformal, or equiareal, we can easily incorporate these additional constraints into our framework.
- If we take $\mathcal{S}$ to be the reference image in which we wish to recover the shape, we can write the constraints for as many surfaces $\overline{\mathcal{S}}$ as we want to increase robustness and the cost only grows linearly with the number of such images. The shape at $\overline{\mathcal{S}}$ can then be expressed in terms of the recovered shape at $\mathcal{S}$.

We now define connections and then show their usage.

4. Connections and Local Diffeomorphisms

In this section, we formalize connections and show their invariance under diffeomorphic deformations. We will use these concepts in Section 5 to implement our Diff-NRSfM framework. We use the notation introduced at the beginning of Section 3 and depicted by Fig. 1.

Moving Frames. Given the projection $\mathbf{x} = [u, v]^T$ in $\mathcal{I}$ and the corresponding 3D point $\mathbf{X}$ on $\mathcal{S}$, we write

$$\mathbf{X}=\phi(\mathbf{x}),\ E(\phi)=(\mathbf{e}_1=\frac{\partial\phi}{\partial u},\mathbf{e}_2=\frac{\partial\phi}{\partial v},\mathbf{e}_3=\mathbf{e}_1\times\mathbf{e}_2). \quad (1)$$

$E(\phi)$ is a moving reference frame for $\mathcal{S}$, and we define $E(\overline{\phi})$ similarly for $\overline{\mathcal{S}}$.

Connections. We now define *connections* that encode differential surface properties that are invariant under diffeomorphic deformations and are at the heart of our approach. Assuming $\mathcal{S}$ to be locally planar, we can rewrite $\phi(u, v)$ as $\beta(u, v)[u, v, 1]^\top$, where β is a linear function representing depth, within a small neighborhood around the projection of any surface point $\mathbf{x}$. Injecting this definition in Eq. 1 yields

$$\begin{aligned}\mathbf{e}_1 &= \beta(u,v)[1+ux_1, vx_1, x_1]^\top,\\ \mathbf{e}_2 &= \beta(u,v)[ux_2, 1+vx_2, x_2]^\top,\\ \mathbf{e}_3 &= \beta(u,v)^2[-x_1, -x_2, 1+ux_1+vx_2]^\top,\end{aligned} \quad (2)$$

where $x_1 = \frac{1}{\beta(u,v)}\frac{\partial\beta}{\partial u}$ and $x_2 = \frac{1}{\beta(u,v)}\frac{\partial\beta}{\partial v}$. The connections Γ^i_{jk} are then taken to be the solutions of the linear system

$$\begin{aligned}\frac{\partial\mathbf{e}_j}{\partial u} &= \Gamma^1_{j1}\mathbf{e}_1+\Gamma^2_{j1}\mathbf{e}_2+\Gamma^3_{j1}\mathbf{e}_3, \quad j=[1,2,3]\\ \frac{\partial\mathbf{e}_j}{\partial v} &= \Gamma^1_{j2}\mathbf{e}_1+\Gamma^2_{j2}\mathbf{e}_2+\Gamma^3_{j2}\mathbf{e}_3.\end{aligned} \quad (3)$$

Because β is assumed to be linear, its partial derivatives that appear in the definition of x_1 and x_2 in Eq. 2 are constant and its second order derivatives are 0. Thus, solving the linear system of Eq. 3 yields

$$\begin{pmatrix}\Gamma^1_{11} & \Gamma^2_{11} & \Gamma^3_{11}\\ \Gamma^1_{21} & \Gamma^2_{21} & \Gamma^3_{21}\\ \Gamma^1_{31} & \Gamma^2_{31} & \Gamma^3_{31}\end{pmatrix} = \frac{\beta^3}{D}\begin{pmatrix}2\beta T_8x_1 & -2\beta T_3x_1^2 & -2x_1^2\\ \beta T_6x_2 & \beta T_5x_1 & -2x_1x_2\\ 2\beta^2T_1x_1 & 2\beta^2T_2x_1 & \beta(2T_8+T_5)x_1\end{pmatrix}, \quad (4)$$

$$\begin{pmatrix}\Gamma^1_{12} & \Gamma^2_{12} & \Gamma^3_{12}\\ \Gamma^1_{22} & \Gamma^2_{22} & \Gamma^3_{22}\\ \Gamma^1_{32} & \Gamma^2_{32} & \Gamma^3_{32}\end{pmatrix} = \frac{\beta^3}{D}\begin{pmatrix}\beta T_6x_2 & \beta T_5x_1 & -2x_1x_2\\ -2\beta T_4x_2^2 & 2\beta T_7x_2 & -2x_2^2\\ 2\beta^2T_1x_2 & 2\beta^2T_2x_2 & \beta(2T_6+T_7)x_2\end{pmatrix},$$

where

$$\begin{aligned}D &= \beta^4(x_1^2+x_2^2+(1+ux_1+vx_2)^2)\ ,\\ T_1 &= x_1+vx_1x_2-ux_2^2, T_2 = x_2+ux_1x_2-vx_1^2\ ,\\ T_3 &= v+(1+v^2)x_2+uvx_1, T_4 = u+(1+u^2)x_1+uvx_2\ ,\\ T_5 &= 1+(1+u^2)x_1^2-(1+v^2)x_2^2+2ux_1\ ,\\ T_6 &= 1+(1+v^2)x_2^2-(1+u^2)x_1^2+2vx_2\ ,\\ T_7 &= 1+(1+u^2)x_1^2+2ux_1+vx_2+uvx_1x_2\ ,\\ T_8 &= 1+(1+v^2)x_2^2+ux_1+2vx_2+uvx_1x_2\ .\end{aligned}$$

From the above equation we can verify that $\Gamma^i_{jk} = \Gamma^i_{kj}$ always holds and that $\Gamma^3_{3k} = \Gamma^1_{1k}+\Gamma^2_{2k}$. This leaves us with a set of 13 distinct $\Gamma^i_{jk}(\phi)$ expressed in terms of image observations, depths, and the depth first-order derivatives. Their formulation may seem complex but this is the price to pay to achieve invariance to diffeomorphic deformations, which we prove below.

Invariance under Local Diffeomorphism. In the previous paragraph, we defined the connections $\Gamma^i_{jk}(\phi)$. We can similarly define the connections $\Gamma^i_{jk}(\overline{\phi})$, which we will denote as $\overline{\Gamma^i_{jk}}$, for $\overline{\mathcal{S}}$. We now discuss their invariance to diffeomorphic deformations, that is,

$$\Gamma^i_{jk}(\phi\circ\eta) = \Gamma^i_{jk}(\overline{\phi})\ . \quad (5)$$

As can be seen in Fig. 1, $\overline{\phi} = \psi\circ\phi\circ\eta$. We show in the supplementary material that it follows that

$$\overline{E}(\overline{\phi}) = \text{diag}(\lambda_1,\lambda_2,\lambda_3)\mathbf{R}E(\phi)\text{diag}(\mathbf{J}_\eta, |\mathbf{J}_\eta|), \quad (6)$$

where $\mathbf{J}_\eta = \begin{pmatrix}\frac{\partial u}{\partial\overline{u}} & \frac{\partial u}{\partial\overline{v}}\\ \frac{\partial v}{\partial\overline{u}} & \frac{\partial v}{\partial\overline{v}}\end{pmatrix}$ is the Jacobian of η, λ_i are scalars and $\mathbf{R}$ is a rotation matrix. As we also show in the supplementary material, injecting Eq. 6 into the definition of the $\overline{\Gamma^i_{jk}}$ yields

$$\begin{pmatrix}\overline{\Gamma}^1_{11} & \overline{\Gamma}^2_{11} & \overline{\Gamma}^3_{11}\\ \overline{\Gamma}^1_{21} & \overline{\Gamma}^2_{21} & \overline{\Gamma}^3_{21}\\ \overline{\Gamma}^1_{31} & \overline{\Gamma}^2_{31} & \overline{\Gamma}^3_{31}\end{pmatrix} = \begin{pmatrix}\mathbf{J}_\eta & 0\\ 0 & |\mathbf{J}_\eta|\end{pmatrix}^{-1}\begin{pmatrix}\frac{\partial\mathbf{J}_\eta}{\partial\overline{u}} & 0\\ 0 & \frac{\partial|\mathbf{J}_\eta|}{\partial\overline{u}}\end{pmatrix}+$$
$$\frac{\partial u}{\partial\overline{u}}\begin{pmatrix}\mathbf{J}_\eta & 0\\ 0 & |\mathbf{J}_\eta|\end{pmatrix}^{-1}\begin{pmatrix}\Gamma^1_{11} & \Gamma^2_{11} & \Gamma^3_{11}\\ \Gamma^1_{21} & \Gamma^2_{21} & \Gamma^3_{21}\\ \Gamma^1_{31} & \Gamma^2_{31} & \Gamma^3_{31}\end{pmatrix}\begin{pmatrix}\mathbf{J}_\eta & 0\\ 0 & |\mathbf{J}_\eta|\end{pmatrix}+$$
$$\frac{\partial v}{\partial\overline{u}}\begin{pmatrix}\mathbf{J}_\eta & 0\\ 0 & |\mathbf{J}_\eta|\end{pmatrix}^{-1}\begin{pmatrix}\Gamma^1_{12} & \Gamma^2_{12} & \Gamma^3_{12}\\ \Gamma^1_{22} & \Gamma^2_{22} & \Gamma^3_{22}\\ \Gamma^1_{32} & \Gamma^2_{32} & \Gamma^3_{32}\end{pmatrix}\begin{pmatrix}\mathbf{J}_\eta & 0\\ 0 & |\mathbf{J}_\eta|\end{pmatrix},$$

$$\begin{pmatrix} \overline{\Gamma}^1_{12} & \overline{\Gamma}^2_{12} & \overline{\Gamma}^3_{12} \\ \overline{\Gamma}^1_{22} & \overline{\Gamma}^2_{22} & \overline{\Gamma}^3_{22} \\ \overline{\Gamma}^1_{32} & \overline{\Gamma}^2_{32} & \overline{\Gamma}^3_{32} \end{pmatrix} = \begin{pmatrix} \mathbf{J}_\eta & 0 \\ 0 & |\mathbf{J}_\eta| \end{pmatrix}^{-1} \begin{pmatrix} \frac{\partial \mathbf{J}_\eta}{\partial \overline{v}} & 0 \\ 0 & \frac{\partial |\mathbf{J}_\eta|}{\partial \overline{v}} \end{pmatrix} + \frac{\partial u}{\partial \overline{v}} \begin{pmatrix} \mathbf{J}_\eta & 0 \\ 0 & |\mathbf{J}_\eta| \end{pmatrix}^{-1} \begin{pmatrix} \Gamma^1_{11} & \Gamma^2_{11} & \Gamma^3_{11} \\ \Gamma^1_{21} & \Gamma^2_{21} & \Gamma^3_{21} \\ \Gamma^1_{31} & \Gamma^2_{31} & \Gamma^3_{31} \end{pmatrix} \begin{pmatrix} \mathbf{J}_\eta & 0 \\ 0 & |\mathbf{J}_\eta| \end{pmatrix} + \frac{\partial v}{\partial \overline{v}} \begin{pmatrix} \mathbf{J}_\eta & 0 \\ 0 & |\mathbf{J}_\eta| \end{pmatrix}^{-1} \begin{pmatrix} \Gamma^1_{12} & \Gamma^2_{12} & \Gamma^3_{12} \\ \Gamma^1_{22} & \Gamma^2_{22} & \Gamma^3_{22} \\ \Gamma^1_{32} & \Gamma^2_{32} & \Gamma^3_{32} \end{pmatrix} \begin{pmatrix} \mathbf{J}_\eta & 0 \\ 0 & |\mathbf{J}_\eta| \end{pmatrix}. \quad (7)$$

The above relation shows that connections are preserved up to a change of variable. In other words, we can compute the connections of $\overline{\mathcal{S}}$ from those of $\mathcal{S}$ using η. In the next section, we exploit this to perform NRSfM.

5. Surface Reconstruction under Local Diffeomorphism

In this section, we use connections and their preservation relations (7) to derive reconstruction equations. We first express the depth and its derivatives at $\overline{\mathcal{S}}$ in terms of the ones at $\mathcal{S}$ and η. We show that this helps in constraining the complexity of the problem. Then we derive constraints to perform reconstruction from a local diffeomorphism and other metric-preserving mappings.

5.1. Relating Depths

(7) expresses $\Gamma^i_j(\overline{\phi})$ in terms of $\Gamma^i_j(\phi)$ and the first- and the second-order derivatives of η. The expanded expressions are shown in the supplementary material. However, not all of the $\overline{\Gamma}^i_j$ depend on the second-order derivatives. In particular, some of the non-diagonal $\overline{\Gamma}^i_j$ are expressed only in terms of Γ^i_j and of the first-order derivatives of η. By considering only these and equating their definition from (4) with that from (7), we can write

$$\begin{aligned} \overline{\Gamma}^3_{11} &\equiv \frac{\overline{x}_1^2 \overline{\beta}^3}{\overline{D}} = \frac{\beta^3 t_1^2}{|\mathbf{J}_\eta| D}, \\ \overline{\Gamma}^3_{12} = \overline{\Gamma}^3_{21} &\equiv \frac{\overline{x}_1 \overline{x}_2 \overline{\beta}^3}{\overline{D}} = \frac{\beta^3 t_1 t_2}{|\mathbf{J}_\eta| D}, \\ \overline{\Gamma}^3_{22} &\equiv \frac{\overline{x}_2^2 \overline{\beta}^3}{\overline{D}} = \frac{\beta^3 t_2^2}{|\mathbf{J}_\eta| D}, \end{aligned} \quad (8)$$

$$\begin{aligned} \overline{\Gamma}^1_{31} &\equiv \frac{\overline{\beta}^5 (\overline{x}_1^2 + \overline{v}\overline{x}_1^2 \overline{x}_2 - \overline{u}\overline{x}_1 \overline{x}_2^2)}{\overline{D}} = \frac{\beta^5 t_1}{D} t_3, \\ \overline{\Gamma}^2_{31} &\equiv \frac{\overline{\beta}^5 (\overline{x}_1 \overline{x}_2 + \overline{u}\overline{x}_1^2 \overline{x}_2 - \overline{v}\overline{x}_1^3)}{\overline{D}} = \frac{\beta^5 t_1}{D} t_4, \\ \overline{\Gamma}^1_{32} &\equiv \frac{\overline{\beta}^5 (\overline{x}_1 \overline{x}_2 + \overline{v}\overline{x}_1 \overline{x}_2^2 - \overline{u}\overline{x}_2^3)}{\overline{D}} = \frac{\beta^5 t_2}{D} t_3, \\ \overline{\Gamma}^2_{32} &\equiv \frac{\overline{\beta}^5 (\overline{x}_2^2 + \overline{u}\overline{x}_1 \overline{x}_2^2 - \overline{v}\overline{x}_1^2 \overline{x}_2)}{\overline{D}} = \frac{\beta^5 t_2}{D} t_4, \end{aligned} \quad (9)$$

$$\text{where } t_1 = \frac{\partial u}{\partial \overline{u}} x_1 + \frac{\partial v}{\partial \overline{u}} x_2, t_2 = \frac{\partial u}{\partial \overline{v}} x_1 + \frac{\partial v}{\partial \overline{v}} x_2,$$

$$t_3 = \left(t_2(vx_1 - ux_2) + \frac{\partial v}{\partial \overline{v}} x_1 - \frac{\partial u}{\partial \overline{v}} x_2 \right),$$
$$t_4 = \left(t_1(ux_2 - vx_1) - \frac{\partial v}{\partial \overline{u}} x_1 + \frac{\partial u}{\partial \overline{u}} x_2 \right).$$

Computing $(\overline{x}_1^2, \overline{x}_1 \overline{x}_2, \overline{x}_2^2)$ from the first three equations in (8), and substituting the results in (9) yields

$$\begin{aligned} \overline{\Gamma}^1_{31} &\equiv \overline{\beta}^2 (t_1 + (\overline{v} t_1 - \overline{u} t_2) \overline{x}_2) = \beta^2 |\mathbf{J}_\eta| t_3, \\ \overline{\Gamma}^2_{31} &\equiv \overline{\beta}^2 (t_2 + (\overline{u} t_2 - \overline{v} t_1) \overline{x}_1) = \beta^2 |\mathbf{J}_\eta| t_4, \\ \overline{\Gamma}^1_{32} &\equiv \overline{\beta}^2 (t_1 + t_2 (\overline{v}\overline{x}_1 - \overline{u}\overline{x}_2)) = \beta^2 |\mathbf{J}_\eta| t_3, \\ \overline{\Gamma}^2_{32} &\equiv \overline{\beta}^2 (t_2 + t_1 (\overline{u}\overline{x}_2 - \overline{v}\overline{x}_1)) = \beta^2 |\mathbf{J}_\eta| t_4. \end{aligned} \quad (10)$$

Multiplying $\overline{\Gamma}^1_{32}$ and $\overline{\Gamma}^2_{32}$ with t_1 and t_2, respectively, and adding the results yields

$$\overline{\beta}^2 (t_1^2 + t_2^2) = \beta^2 |\mathbf{J}_\eta|^2 (x_1^2 + x_2^2) = \beta^2 |\mathbf{J}_\eta| (t_3 t_1 + t_4 t_2). \quad (11)$$

Computing $\frac{\overline{\beta}}{\beta}$ from (11) and substituting the result in the definitions of $\overline{\Gamma}^1_{31}, \overline{\Gamma}^2_{31}$ of (10) gives

$$\bar{x}_1 = t_1 \frac{N_1}{N_2} \text{ and } \bar{x}_2 = t_2 \frac{N_1}{N_2}, \quad (12)$$

where $N_1 = (t_1 t_4 - t_2 t_3)$ and $N_2 = (t_1 t_3 + t_2 t_4)(\overline{u} t_2 - \overline{v} t_1)$. In (11) and (12), we express the unknowns, $(\frac{\overline{\beta}}{\beta}, \overline{x_1}, \overline{x_2})$ on $\overline{\mathcal{S}}$, in terms of (x_1, x_2) on $\mathcal{S}$ and a known η. Therefore, the unknown quantities, depth and its derivatives, for any additional surface can be expressed in terms of the unknown quantities of $\mathcal{S}$. Thus adding additional views yields additional constraints without increasing the number of unknowns, which limits the complexity of the system.

5.2. Reconstruction Equations

We first derive the reconstruction equations under the assumption of local diffeomorphism. Then, we show how to add the explicit metric-preserving constraints of isometry, conformality and equiareality.

Diffeomorphic NRSfM. In the last section, we used few of the connections to derive $(\frac{\overline{\beta}}{\beta}, \overline{x_1}, \overline{x_2})$ in terms of (x_1, x_2). These relations were expressed with only first-order derivatives of η. The remaining connections are sufficient to obtain reconstruction constraints but they all contain second-order derivatives of η and therefore, we cannot use them directly. According to [26], the image of two given planes should satisfy 2D Schwarzian equations given by

$$\frac{\partial u}{\partial \overline{u}} \frac{\partial^2 v}{\partial \overline{u}^2} - \frac{\partial v}{\partial \overline{u}} \frac{\partial^2 u}{\partial \overline{u}^2} = 0, \frac{\partial v}{\partial \overline{v}} \frac{\partial^2 u}{\partial \overline{v}^2} - \frac{\partial u}{\partial \overline{v}} \frac{\partial^2 v}{\partial \overline{v}^2} = 0\,. \quad (13)$$

The above expressions appear in $(\overline{\Gamma}^1_{22}, \overline{\Gamma}^2_{11})$ and setting them to zero cancels out second-order derivatives. Since we

assume surfaces to be locally planar, we use (13) to write

$$\overline{\Gamma}^2_{11} \equiv \frac{\overline{x}_1^2\overline{\beta}^4\overline{T}_3}{\overline{D}} = \frac{\beta^4 t_1^2}{|\mathbf{J}_\eta| D}\left(\frac{\partial u}{\partial \overline{u}}T_3 - \frac{\partial v}{\partial \overline{u}}T_4\right),$$
$$\overline{\Gamma}^1_{22} \equiv \frac{\overline{x}_2^2\overline{\beta}^4\overline{T}_4}{\overline{D}} = \frac{\beta^4 t_2^2}{|\mathbf{J}_\eta| D}\left(-\frac{\partial u}{\partial \overline{v}}T_3 + \frac{\partial v}{\partial \overline{v}}T_4\right). \quad (14)$$

Computing $(\overline{x}_1^2, \overline{x}_1\overline{x}_2, \overline{x}_2^2)$ from (8), as before, and substituting the resulting expressions in (14) yields

$$\overline{\beta T}_3 = \beta\left(\frac{\partial u}{\partial \overline{u}}T_3 - \frac{\partial v}{\partial \overline{u}}T_4\right), \overline{\beta T}_4 = -\beta\left(\frac{\partial u}{\partial \overline{v}}T_3 - \frac{\partial v}{\partial \overline{v}}T_4\right) \quad (15)$$

Substituting $(\overline{x}_1, \overline{x}_2)$ from (12) to the above equation gives

$$\overline{\beta}(\overline{v}N_2 + ((1+\overline{v}^2)t_2 + \overline{u}\overline{v}t_1)N_1) = \beta N_2\left(\frac{\partial u}{\partial \overline{u}}T_3 - \frac{\partial v}{\partial \overline{u}}T_4\right)$$
$$\overline{\beta}(\overline{u}N_2 + ((1+\overline{u}^2)t_1 + \overline{u}\overline{v}t_2)N_1) = -\beta N_2\left(\frac{\partial u}{\partial \overline{v}}T_3 - \frac{\partial v}{\partial \overline{v}}T_4\right) \quad (16)$$

Squaring (16) and substituting $\left(\frac{\overline{\beta}}{\beta}\right)^2$ from (11) gives

$$|\mathbf{J}_\eta|^2(t_1^2+t_2^2)(\overline{v}N_2 + ((1+\overline{v}^2)t_2 + \overline{u}\overline{v}t_1)N_1)^2 = N_2^2(x_1^2+x_2^2)\left(\frac{\partial u}{\partial \overline{u}}T_3 - \frac{\partial v}{\partial \overline{u}}T_4\right)^2,$$
$$|\mathbf{J}_\eta|^2(t_1^2+t_2^2)(\overline{u}N_2 + ((1+\overline{u}^2)t_1 + \overline{u}\overline{v}t_2)N_1)^2 = N_2^2(x_1^2+x_2^2)\left(-\frac{\partial u}{\partial \overline{v}}T_3 + \frac{\partial v}{\partial \overline{v}}T_4\right)^2. \quad (17)$$

Thus we obtain two polynomials in two variables. With the solution to (x_1, x_2), we can write the normals at $\mathcal{S}$ using (2). We can then obtain $(\overline{x}_1, \overline{x}_2)$ at $\overline{\mathcal{S}}$ using (x_1, x_2) in (12). Thus, we obtain normal on all the surfaces. We can obtain an up-to-scale depth by integrating the normals.

Solving Polynomial Equations. We solve (17) by using resultants [12] to convert these equations to univariate polynomials, which can then be easily solved. A resultant is defined as an expression written in terms of the coefficients of two polynomials. If the polynomials have a common root, their resultant evaluates to zero. We write the equations (17) as $A(x_1, x_2) = 0$ and $B(x_1, x_2) = 0$. Their resultant with respect to x_1 is given by $R(x_2)$. Since these equations must bear a common root, we get $R(x_2) = 0$. $R(x_2)$ is a univariate equation of degree 10. We show the structure of $R(x_2)$ in the supplementary equation. We substitute x_2 into $A(x_1, x_2)$ and $B(x_1, x_2)$ and solve for x_1.

Obtaining a Unique Solution. $R(x_2)$ is a degree 10 polynomial. Hence, it is highly likely to have multiple real solutions for (x_1, x_2) for a pair of images. We then need at least one more image to disambiguate the existing solutions. Thus, we obtain a unique solution from ≥ 3 images.

Metric Preservation from Local Diffeomorphism. Under a local diffeomorphism, the moving frames $\overline{E}$ and E are related by (6). Since all metric-preserving deformations (including isometry, conformity and equiareality), fall under the category of locally diffeomorphic mappings, we write

$$\overline{E}^\top\overline{E} = \text{diag}(\lambda_1^2, \lambda_2^2, \lambda_3^2)\text{diag}(\mathbf{J}_\eta, |\mathbf{J}_\eta|)^\top E^\top E \text{diag}(\mathbf{J}_\eta, |\mathbf{J}_\eta|),$$

where,

$$\text{Isometry} \equiv \lambda_1 = \lambda_2 = \lambda_3 = 1,$$
$$\text{Conformity} \equiv \lambda_1 = \lambda_2 = \lambda_3 = \lambda,$$
$$\text{Equiareality} \equiv \lambda_3 = \lambda_1\lambda_2 = 1. \quad (18)$$

Using (2) in the above equation lets us write

$$\overline{\beta}^2\text{diag}(\overline{\mathbf{G}}, \overline{\beta}^2|\overline{\mathbf{G}}|) = \beta^2\text{diag}(\lambda_1^2, \lambda_2^2, \lambda_3^2)\text{diag}(\mathbf{P}, \beta^2|\mathbf{J}_\eta|^2|\mathbf{G}|), \text{with}$$
$$\mathbf{P} = \mathbf{J}_\eta^\top\mathbf{G}\mathbf{J}_\eta$$
$$\mathbf{G} = \begin{pmatrix} \epsilon x_1^2+1+2ux_1 & \epsilon x_1x_2+ux_2+vx_1 \\ \epsilon x_1x_2+ux_2+vx_1 & \epsilon x_2^2+1+2vx_2 \end{pmatrix},$$
$$\overline{\mathbf{G}} = \begin{pmatrix} \overline{\epsilon}\overline{x}_1^2+1+2\overline{u}\overline{x}_1 & \overline{\epsilon}\overline{x}_1\overline{x}_2+\overline{u}\overline{x}_2+\overline{v}\overline{x}_1 \\ \overline{\epsilon}\overline{x}_1\overline{x}_2+\overline{u}\overline{x}_2+\overline{v}\overline{x}_1 & \overline{\epsilon}\overline{x}_2^2+1+2\overline{v}\overline{x}_2 \end{pmatrix},$$
$$\epsilon = 1+u^2+v^2, \overline{\epsilon} = 1+\overline{u}^2+\overline{v}^2. \quad (19)$$

Substituting $(\overline{x}_1, \overline{x}_2)$ from (12) in the equation above gives

$$\overline{\beta}^2\text{diag}(\overline{\mathbf{G}}, \overline{\beta}^2|\overline{\mathbf{G}}|) = \beta^2\text{diag}(\lambda_1^2, \lambda_2^2, \lambda_3^2)\text{diag}(\mathbf{P}, \beta^2|\mathbf{J}_\eta|^2|\mathbf{G}|), \text{with}$$
$$\overline{\mathbf{G}}[1,1] = \left(\frac{1}{N_2}\right)^2(\overline{\epsilon}t_1^2N_1^2 + N_2^2 + 2\overline{u}t_1N_1N_2)$$
$$\overline{\mathbf{G}}[1,2] = \overline{\mathbf{G}}[2,1] = \left(\frac{1}{N_2}\right)^2(\overline{\epsilon}t_1t_2N_1^2 + \overline{u}t_2N_1N_2 + \overline{v}t_1N_1N_2)$$
$$\overline{\mathbf{G}}[2,2] = \left(\frac{1}{N_2}\right)^2(\overline{\epsilon}t_2^2N_1^2 + N_2^2 + 2\overline{v}t_2N_1N_2) \quad (20)$$

Our Diff-NRSfM solution in (17) was derived using only the assumption of locally diffeomorphic deformations. In a scenario where we know *a priori* the specific surface properties, we can incorporate the corresponding constraints explicitly in Diff-NRSfM using (19).

Conformal Constraints. We obtain the constraints by taking the ratios of the components of $\mathbf{P}$ and $\overline{\mathbf{G}}$ in (20) to remove λ and $\frac{\overline{\beta}}{\beta}$. The equations are given by

$$\frac{\overline{\mathbf{G}}[1,1]}{\overline{\mathbf{G}}[2,2]} = \frac{P[1,1]}{P[2,2]}, \frac{\overline{\mathbf{G}}[1,2]}{\overline{\mathbf{G}}[2,2]} = \frac{P[1,2]}{P[2,2]} \quad (21)$$

Thus we obtain two relations of degree 9 in two variables.

Equiareal Constraints. We obtain the constraints by comparing areas, which are given by determinants, in (20). Substituting $\left(\frac{\overline{\beta}}{\beta}\right)^2$ from (11) gives

$$(t_1^2+t_2^2)^2|\overline{\mathbf{G}}| = (x_1^2+x_2^2)^2|\mathbf{G}|. \quad (22)$$

Thus we obtain a relation of degree 8 in two variables.

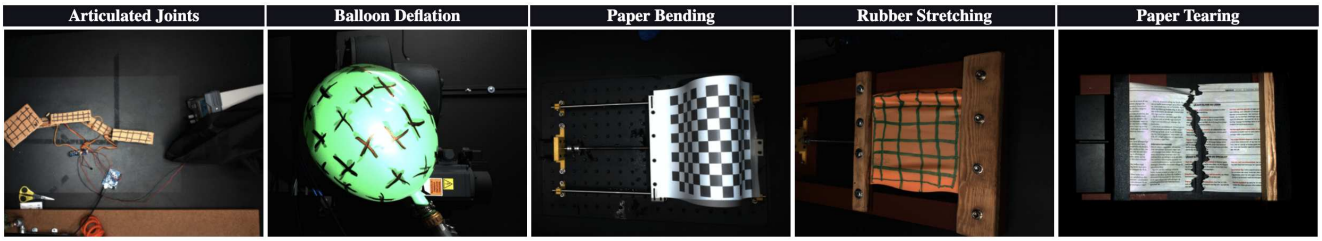


Figure 2: NRSfM challenge dataset.

Isometric Constraints. Isometry is expressed as a combination of conformity and equiareality. We use (21) and (22) to define isometric constraints.

5.3. Algorithm

Let $\{\mathbf{x}_j^i\}, i \in [1, M], j \in [1, N]$, denote a set of N point correspondences between M images. Our goal is to find the 3D point and the normal corresponding to each $\mathbf{x}_j^i$. We take an arbitrary image $\mathcal{I}$ as our reference and use a standard algorithm such as optical flow [29, 28] or SIFT [22] to compute an η mapping between each remaining images and $\mathcal{I}$. Our point-wise Diff-NRSfM has the following steps:

- *Solve for* x_2. For each image paired with the reference, compute the resultant $R(x_2)$ from $A(x_1, x_2) = 0$, $B(x_1, x_2) = 0$, defined in (17). Find x_2 by minimizing the sum of squares of $R(x_2) = 0$ computed over all available image pairs.

- *Solve for* x_1. Substitute x_2 obtained from the previous step into $(A(x_1, x_2) = 0, B(x_1, x_2) = 0)$ and find x_1 by minimizing their sum of squares.

- *(optional) Add metric preserving constraints.* If the deformation model is known *a priori*, minimize the sum of squares of (21), or (22), or both to add conformal, equiareal, or isometric constraints. Use (x_1, x_2) obtained from previous steps to initialize this solution.

- *Find local normals.* Use (x_1, x_2) to express local normals as in (2).

After obtaining a local normal for each $\mathbf{x}_j^i$, we integrate them to compute depth up to a scale factor.

6. Experiments

We show results on NRSfM challenge dataset [17], one synthetic and three real datasets. We denote our Diff-NRSfM as **Diff** and its variants with isometric, conformal or equiareal constraints as **DiffI**, **DiffC** and **DiffE**. We compare it to NRSfM methods that assume isometry/conformity, **Pa17** [24, 25], inextensibility, **Ch17** [9], soft-inextensibility, **Vi12** [34], local rigidity **Ch14** [8] and low-rank constraints **Go11** [13]. We report mean shape E_s and mean depth E_d errors computed as RMSEs between reconstructed and ground-truth normals and 3D points.

NRSfM Challenge Dataset (perspective)					
Method	Articulated	Balloon	Paper	Stretch	Tearing
Vi12	69.11	41.76	53.91	48.98	45.93
Ch17	91.55	58.00	66.54	62.49	53.93
Go11	44.49	27.94	36.06	29.23	23.01
Best	40.74	27.94	35.75	29.23	23.01
Diff	**21.28**	**27.70**	**31.01**	**17.35**	**15.73**

NRSfM Challenge Dataset (orthographic)					
Method	Articulated	Balloon	Paper	Stretch	Tearing
Vi12	61.43	36.75	47.41	45.56	37.87
Ch17	88.66	58.27	66.98	66.27	56.67
Go11	35.63	24.88	31.96	24.25	**17.59**
Best	35.51	**14.55**	66.98	18.30	**17.59**
Diff	**18.67**	28.55	**31.82**	**17.12**	19.71

Figure 3: NRSfM challenge results. **Best** represents the state of the art. In the perspective case, it is **Go11** for balloon, stretch and tearing. For articulated and paper, it is [31] and [15], respectively. In the orthographic case, it is **Ch17** and **Go11** for balloon and tearing and [19] for the rest.

NRSfM Challenge Dataset. It consists of 5 image sequences depicted by Fig. 2. They feature 5 kinds of non-rigid motions: articulated (piecewise-rigid), balloon (confomal), paper bending (isometric), rubber (elastic), and paper being torn. The dataset features images from 6 different camera motions and provides image points captured assuming both a perspective and an orthographic projection. It provides only one ground-truth surface for each of the sequences. The correspondences are sparse and not well-distributed across the images. Fig. 3 compares the performance of **Diff** with that of other methods in terms of E_d, measured in mm, with **Best** being the one that does best to date and that we beat by a large margin in the perspective case. Note that **Diff** still does well in the orthographic case, even though it is explicitly designed for the perspective case. Note also that **Pa17** does not appear in this table, presumably because the sparsity of the correspondences are not ideal to compute the second derivatives that it needs.

Fig. 4 shows qualitative results on the stretch sequence with flyby and semicircle motions. **Diff** reconstructs the surface well even with orthographic images. The stretching of the rubber is well-preserved, except in some corners. This is because **Diff** does not require strong constraints such as isometry, and thus can reconstruct a wide range of deformations, including, up to an extent, elasticity.

Cylinder Dataset. It consists of 400 points tracked across 10 images (640× 480p) of a cylindrical surface deforming isometrically created synthetically. We add a gaussian noise of 3 pixels to the image points. The results are shown in Figure 5a. **Pa17** shows the best performance, with our method, **Diff**, being close to it. The results are slightly improved with the additional constraints of **DiffI**, **DiffC** and **DiffE**. The constraints obtained for **DiffI**, **DiffC** and **DiffE** are high order polynomials which do not lead to strong constraints in practical terms. **Ch14** and **Vi12** also show a decent performance, but it is not consistent throughout the dataset. **Ch17** does not perform well on this dataset, it yields a very high normal error which indicates flattening. This is because it requires the constraints to be very distinct especially if the images are as few as 10. We do not show the results of **Go11** as it is a low-rank method and requires a lot more than 10 images to give a stable performance.

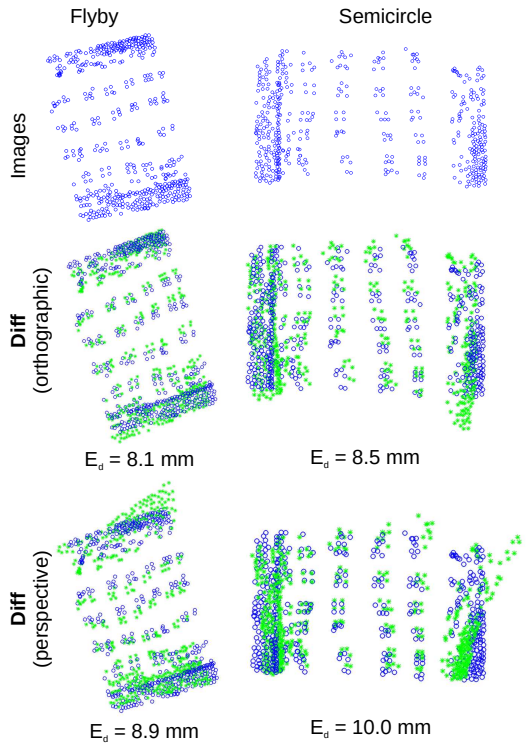


Figure 4: Results on the stretch sequence. Blue is the ground truth and green is the reconstruction using **Diff**.

Table 1: RMSE for all methods.

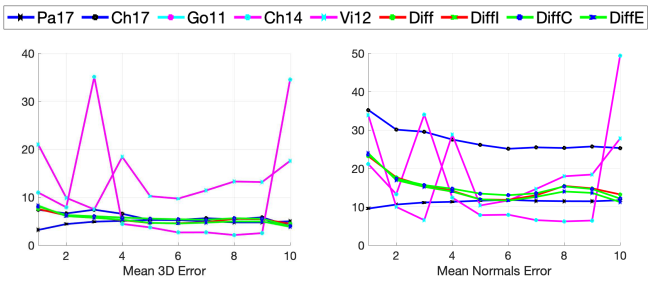
Method	Kinect paper		Rug		Tshirt	
	Es	Ed	Es	Ed	Es	Ed
Go11	18.65	28.88	19.87	99.89	X	X
Vi12	X	X	X	X	18.01	20.67
Ch14	X	X	X	X	16.52	16.06
Ch17	**6.40**	**5.38**	18.67	63.54	28.93	3.67
Pa17	10.50	8.26	**16.70**	52.40	**11.22**	**8.94**
Diff	19.41	10.15	21.49	**46.05**	27.05	12.96
DiffI	19.20	9.87	22.55	48.90	27.27	13.07
DiffC	19.14	9.88	21.59	46.80	27.12	13.03
DiffE	19.10	9.80	22.00	48.00	27.07	13.00

When varying the noise on the Cylinder dataset from 1 to 5 pixels, the mean shape errors obtained by **Diff** are $9.8-12.3$ degrees. For [24], which relies on second order derivatives, the mean shape errors in the same experiment lie in the range of $9.5-14.7$ degrees. Thus, **Diff** yields better normals in the presence of noise, even if it uses weaker constraints than isometry or conformality, thanks to its reliance on first order warp derivatives only.

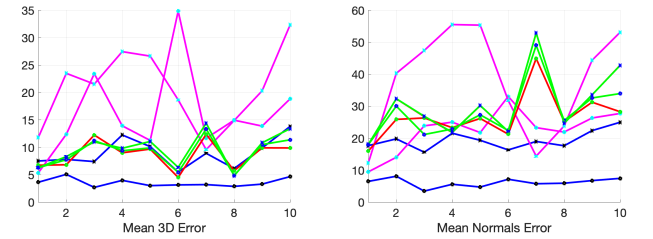
Tshirt Dataset. [8] It consists of 85 manually computed points correspondences across 10 images of a tshirt deforming isometrically from very different views. Figure 5b shows the results. **Ch17** shows the best performance. **Pa17** also has a very stable performance. The performance of our methods, **Diff**, **DiffI**, **DiffC** and **DiffE**, is also very stable. **Ch14** and **Vi12** do not show a consistent performance but they get decent results for most of the images.

Paper Dataset. [27] It consists of 190 images of a paper deforming isometrically with 1500 point correspondences on them. For such a high number of images, **Ch14** and **Vi12** are highly impractical. We used only 150 point correspondences on this dataset. Figure 5c shows the 3D reconstruction error for all compared methods. Amongst our methods, we show the results of **Diff** only as the remaining ones perform very close to it, as seen in Table 1. **Ch17** shows the best performance. The performance of our methods is very close to **Pa17**. **Go11** shows the worst performance.

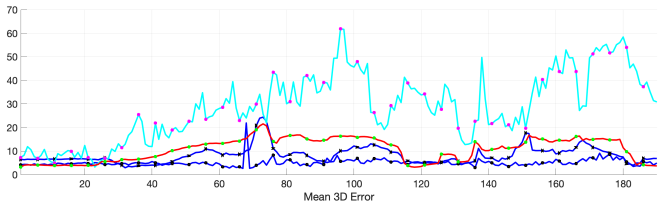
Rug Dataset. [24] It consists of 160 images of a rug deforming isometrically with 1500 point correspondences obtained using optical flow [29]. We consider 350 points for evaluating all methods. This data contains correspondence errors arising from the lack of texture on the object. Figure 5d shows the comparison of the 3D error of **Diff** with the rest. We see that **Diff** performs the best on this dataset. **Pa17**'s performance is degraded due to poor computation of second-order derivatives in noisy conditions. **Ch17** degrades even more and its performance is worse than **Diff** and **Pa17**. **Go11** does not perform well.



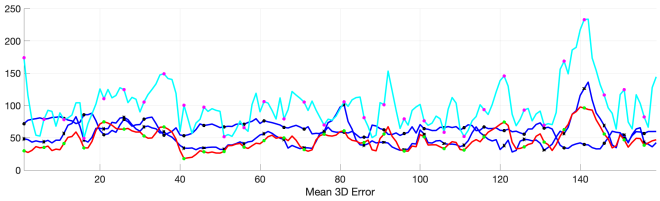
(a) Cylinder dataset



(b) Tshirt dataset



(c) Paper dataset



(d) Rug dataset

Figure 5: Experiments on 1 synthetic and 3 real datasets.

Discussion. In short, when the correspondences are sparse, as in the NRSfM Challenge dataset, our approach dominates, as shown in Fig. 2, and when they are dense and of high quality, **Pa17** does best but our approach comes very close behind in terms of accuracy. However, when met with noisy data, such as in the Rug dataset, **Pa17** degrades due to high error in the second order derivatives and **Diff** becomes noticeably better, as shown in Table 1.

Computation Time. **Ch14** and **Vi12** usually take 15-20 minutes to reconstruct 10 images. Therefore, we did not evaluate them on large datasets. **Ch17** takes almost 15 minutes for 30 images, therefore, we split the large sequences into sets of 30 images and evaluated this method. **Pa17** is a local method with linear complexity, like ours, but it uses an expensive polynomial solver which takes 1.5 seconds to evaluate normals at a point constrained from 10 images. In similar conditions, **Diff** takes only 10 ms, thus, it is 10× faster. In addition, it only requires first order derivatives which can be computed within 20-30 ms for each image pair. Refined second-order derivatives that are required by **Pa17** may take anything between 2-5 seconds.

Parameter Setting. We require a minimum of 3 images for reconstruction. However, the accuracy increases with the number of images, stabilizing after 5 or 6. In practice, we typically use more images (between 8-10). This ensures that sufficient constraints are available even when dealing with images with small visual motion, which is common in short-baseline data. The only parameters to be tuned are the Bicubic B-Spline (BBS) hyper-parameters, which are related to warp control centers and smoothness. We fixed them for all our experiments and do not anticipate having to change them for other data.

Failure Cases. Our methods rely on image correspondences to formulate reconstruction constraints. They will fail in the absence of image texture as image correspondences cannot be computed anymore with the existing image matching methods. Furthermore, the application of our methods is limited to smooth objects as we assume the surfaces to be smooth manifolds.

7. Conclusions and Future Directions

In this paper, we have explored the limiting case for NRSfM: What minimum assumptions on surface deformations are required to solve the problem. We have shown that the assumption of local diffeomorphism, which is a generic surface property of surfaces, yields enough constraints to perform reconstruction. Our experiments have validated our theoretical formulation and demonstrated that it compares favorably to methods that rely on much stronger constraints such as isometry or conformality. Furthermore, it is faster and applicable in more general settings because it only requires first-order derivatives instead of second-order. In the future, we intend to exploit the strengths of this method to develop refinement methods for NRSfM, to which Diff-NRSfM can serve as an initialization. In addition, we plan to extend our method to non-smooth objects by decomposing a non-smooth object into smooth parts, which can be reconstructed independently and stitched together.

Acknowledgements. This work was supported in part by the Swiss Innovation Agency.

References

[1] I. Akhter, Y. Sheikh, and S. Khan. In Defense of Orthonormality Constraints for Nonrigid Structure from Motion. In *Conference on Computer Vision and Pattern Recognition*, June 2009. 2

[2] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Nonrigid Structure from Motion in Trajectory Space. In *Advances in Neural Information Processing Systems*, December 2008. 1

[3] F.L. Bookstein. Principal Warps: Thin-Plate Splines and the Decomposition of Deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585, 1989. 2

[4] C. Bregler, A. Hertzmann, and H. Biermann. Recovering Non-Rigid 3D Shape from Image Streams. In *Conference on Computer Vision and Pattern Recognition*, 2000. 1, 2

[5] A. Del Bue, X. Llad, and L. Agapito. Non-Rigid Metric Shape and Motion Recovery from Uncalibrated Images Using Priors. In *Conference on Computer Vision and Pattern Recognition*, 2006. 1

[6] A. Del Bue, F. Smeraldi, and L. Agapito. Non-Rigid Structure from Motion Using Non-Parametric Tracking and Non-Linear Optimization. In *Conference on Computer Vision and Pattern Recognition*, 2004. 2

[7] É Cartan. Sur les variétés à connexion affine, et la théorie de la relativité généralisée (premier partie). *Annales Scientifiques de l'École Normale Supérieure*, 40:325–412, 1923. 2

[8] A. Chhatkuli, D. Pizarro, and A. Bartoli. Non-Rigid Shape-From-Motion for Isometric Surfaces Using Infinitesimal Planarity. In *British Machine Vision Conference*, 2014. 1, 2, 6, 7

[9] A. Chhatkuli, D. Pizarro, T. Collins, and A. Bartoli. Inextensible Non-Rigid Structure-From-Motion by Second-Order Cone Programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2017. 1, 2, 6

[10] Y. Dai, H. Li, and M. He. A Simple Prior-Free Method for Non-Rigid Structure from Motion Factorization. In *Conference on Computer Vision and Pattern Recognition*, 2012. 2

[11] Y. Dai, H. Li, and M. He. A Simple Prior-Free Method for Non-Rigid Structure-From-Motion Factorization. *International Journal of Computer Vision*, 107(2):101–122, 2014. 1

[12] I. M. Gelfand, M.M. Kapranov, and A.V. Zelevinsky. *Discriminants, Resultants, and Multidimensional Determinants*. Boston: Birkhauser, 1994. 5

[13] P.F.U. Gotardo and A.M. Martinez. Kernel Non-Rigid Structure from Motion. In *International Conference on Computer Vision*, 2011. 6

[14] P. F. Gotardo and A. M. Martinez. Computing Smooth Time Trajectories for Camera and Deformable Shape in Structure from Motion with Occlusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):2051–2065, 2011. 1, 2

[15] O. C. Hamsici, P. F. Gotardo, and A. M. Martinez. Learning Spatially-Smooth Mappings in Non-Rigid Structure from Motion. In *European Conference on Computer Vision*, 2012. 6

[16] T. A. Ivey and J. M. Landsberg. *Cartan for Beginners: Differential Geometry via Moving Frames and Exterior Differential Systems, Second Edition*. 2016. 2

[17] S. H. N. Jensen, A. Del Bue, M. E. B. Doest, and H. Aanæs. A Benchmark and Evaluation of Non-Rigid Structure from Motion. In *arXiv Preprint*, 2018. 6

[18] P. Ji, H. Li, Y. Dai, and I. Reid. Maximizing Rigidity Revisited: A Convex Programming Approach for Generic 3D Shape Reconstruction from Multiple Perspective Views. In *International Conference on Computer Vision*, 2017. 1, 2

[19] S. Kumar, Y. Dai, and H. Li. Spatio-Temporal Union of Subspaces for Multi-Body Non-Rigid Structure-From-Motion. *Pattern Recognition*, 71:428–443, 2017. 6

[20] J. Lamarca, S. Parashar, A. Bartoli, and J. M. M. Montiel. DefSLAM: Tracking and Mapping of Deforming Scenes from Monocular Sequences, 2019. 1

[21] M. Lee and S. Cho, J.and Oh. Consensus of Non-Rigid Reconstructions. In *Conference on Computer Vision and Pattern Recognition*, 2016. 1, 2

[22] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 20(2):91–110, November 2004. 2, 6

[23] S. Parashar and A. Bartoli. 3DVFX: 3D Video Editing Using Non-Rigid Structure-From-Motion. In *Eurographics*, 2019. 1

[24] S. Parashar, D. Pizarro, and A. Bartoli. Isometric Non-Rigid Shape-From-Motion with Riemannian Geometry Solved in Linear Time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 1, 2, 6, 7

[25] S. Parashar, D. Pizarro, and A. Bartoli. Local Deformable 3D Reconstruction with Cartan's Connections. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 1, 2, 6

[26] D. Pizarro, R. Khan, and A. Bartoli. Schwarps: Locally Projective Image Warps Based on 2D Schwarzian Derivatives. *International Journal of Computer Vision*, 119(2):93–109, 2016. 1, 4

[27] M. Salzmann, R. Hartley, and P. Fua. Convex Optimization for Deformable Surface 3D Tracking. In *International Conference on Computer Vision*, October 2007. 7

[28] D. Sun, X. Yang, M. Liu, and J. Kautz. PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume. In *CVPR*, 2018. 2, 6

[29] N. Sundaram, T. Brox, and K. Keutzer. Dense point trajectories by gpu-accelerated large displacement optical flow. In *European Conference on Computer Vision*, 2010. 2, 6, 7

[30] J. Taylor, A. D. Jepson, and K. N. Kutulakos. Non-Rigid Structure from Locally-Rigid Motion. In *Conference on Computer Vision and Pattern Recognition*, June 2010. 1, 2

[31] L. Torresani, A. Hertzmann, and C. Bregler. Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):878–892, 2008. 6

[32] L. Torresani, D. B. Yang, E.J. Alexander, and C. Bregler. Tracking and Modeling Non-Rigid Objects with Rank Constraints. In *Conference on Computer Vision and Pattern Recognition*, pages 493–500, 2001. 1

[33] A. Varol, M. Salzmann, E. Tola, and P. Fua. Template-Free Monocular Reconstruction of Deformable Surfaces. In *International Conference on Computer Vision*, September 2009. 1, 2

[34] S. Vicente and L. Agapito. Soft Inextensibility Constraints for Template-Free Non-Rigid Reconstruction. In *European Conference on Computer Vision*, 2012. 1, 2, 6