

High Dimensional Space Model for Dense Monocular Surface Recovery*

Vladislav Golyanik
Didier Stricker
Department of Computer Science, University of Kaiserslautern
Department Augmented Vision, DFKI

Abstract

Dense surface reconstruction from monocular image sequences — known as *Non-Rigid Structure from Motion (NRSfM)* — is a highly ill-posed inverse problem. The objective of *NRSfM* is to learn 3D shapes from 2D point tracks in an unsupervised manner. While existing methods rely on low-rank models, we propose the concept of *High Dimensional Space Model (HDSM)*. In *HDSM*, time-varying geometry is encoded by a high-dimensional static structure projected into different metric subspaces. To express non-rigid deformations, instead of directly modelling in the 3D space, we gradually increase space dimensionality as the complexity of the scene increases. *HDSM* allows for a compact representation with deformation localisation and can be interpreted as a generalisation of the previously proposed models for *NRSfM*. Relying on *HDSM*, we develop an algorithm for dense monocular surface recovery. Experiments show that the proposed method achieves high accuracy while allowing for the fine-grained control.

1. Introduction

Templateless monocular surface recovery or *Non-Rigid Structure from Motion (NRSfM)* exploits motion and deformation cues for unsupervised learning of 3D shapes from 2D point correspondences. Three models for *NRSfM* have been proposed so far, *i.e.*, *Low-Rank Space Model (LRSM)* [12], *Trajectory Space Model (TSM)* [9] and *Force Space Model (FSM)* [2]. The common for all of them is the modelling of shapes or point trajectories as a linear combination of basis elements. Both the recovered shapes and the basis reside in 3D space. It has been shown that *LRSM*, *TSM* and *FSM* are dual to each other [2, 5].

In this paper, we propose to model time-varying 3D surface geometry by projecting a multi-dimensional *static* structure into different 3D subspaces. The resulting lifted representation — *High-Dimensional Space Model (HDSM)* — generalises previously proposed models for *NRSfM* and

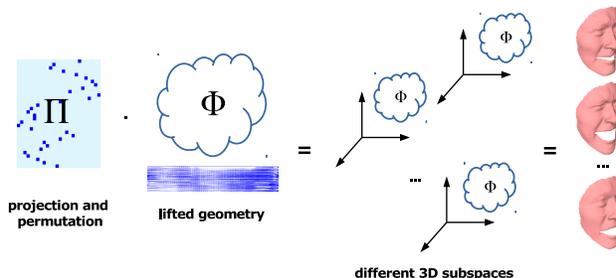


Figure 1: An overview of the main idea of the paper. We propose to model different 3D states of a non-rigid shape by projecting a high-dimensional structure Φ into different 3D subspaces. Π is a permutation matrix which relates projective 3D subspaces with the input frames.

allows designing an algorithm with unique properties. Further on, we refer to this high-dimensional rigid structure Φ as *lifted geometry*. The main motivation for *HDSM* is to stay as close as possible to the raw data representation.

HDSM is tightly related to compressed scene representations which follows from the main idea of *HDSM*, see Fig. 1. The compression in *HDSM* can be implemented on multiple levels (*e.g.*, frame-to-frame or point trajectory levels in the lifted space). Thus, *HDSM* is furnished with *lift-compress* and *decompress-expand* operators. *Lift-compress* allows to pass into the lifted space, detect frame-to-frame redundancies and suppress small local deformations and noise whereas *decompress-expand* generates human-interpretable 3D representations. The lifted geometry provides cues for deformation localisation (by analysing point trajectory patterns) and segmentation from deformation (segmenting the scene into rigid and non-rigid regions). A coarse-to-fine or wavelet-decomposition-like effect is also theoretically possible when applying multiple *lift-compress* operators.

To the best of our knowledge, the idea of compressible representations in the context of *NRSfM* remained unexplored in the literature so far. We believe, however, that especially in the dense case compression is important, as plain representations can occupy gigabytes of memory. In embedded systems, a several-times reduction of transmitted or saved data is particularly relevant. Due to the factorisation

*This work was supported by the BMBF project DYNAMICS (01IW15003). The authors thank Antonio Agudo for helpful remarks.

into camera pose and compact lifted geometry occurring in our approach, the reconstructed scene can be naturally compressed with the compression ratio of 10 and higher.

The introduction of HDSM and initial investigation of its properties is the main contribution and focus of this paper. We show that HDSM can be interpreted as a generalisation of the previously proposed models for NRSfM (Sec. 4). Based on HDSM, we design a new energy-based template-free method for monocular surface recovery — Lifted Coherent Depth Fields (L-CDF, Sec. 5). Our method includes two terms — the data term and the coherency term (CT) — a new spatial regularizer [20]. L-CDF requires a minimal number of priors. The low-rank constraint in 3D space is imposed implicitly, *i.e.*, we do not require any terms accounting for the low-rank nature of deformations in 3D space. We achieve this property by maintaining an optimal non-redundant lifted geometry in every optimisation step.

CT originates in the motion coherence theory [41] and imposes a soft constraint of coherent displacements on adjacent points in the reproducing kernel Hilbert space. Besides the properties inherited from HDSM, L-CDF is well parallelisable, robust and easy to implement due to CT.

2. Related Work

Since the first approach for monocular templateless non-rigid surface recovery was proposed [12], multiple successor methods elaborated on the idea of constraining surfaces to lie in the linear subspaces spanned by a few basis shapes [11, 39, 7, 29]. The main difficulty in modelling with an explicit basis is the need to decide about the optimal basis cardinality K since it varies across datasets exhibiting different degrees of deformations and non-linearities. Moreover, there is no guarantee that an optimal K exists which captures all shape variation while discarding the noise [39].

Due to high ill-posedness of the problem, multiple priors were proposed over the last years such as basis [40], shape prior [13] or temporal smoothness [42, 9, 22] constraints. Torresani *et al.* [39] constrain 3D shapes to lie near a linear subspace in a probabilistic manner, with a Gaussian prior on every basis shape. To reduce the number of unknowns, several methods rely on a known trajectory basis [9] or model surface deformations with physics-based priors [3, 2, 6]. Several methods minimise the rank of the stacked shape matrix through a nuclear norm term [15, 18, 19]. Dai *et al.* [15] proposed an energy-based method with a minimal number of priors which was extended for dense reconstructions in [18]. To handle large and long occlusions, the recent enhancement imposes a Gaussian prior on the surface geometry guided by a per-pixel occlusion tensor [19]. Similarly, our formulation does not include unknowns accounting for the basis, and we perform a low-rank approximation of the expanded shape matrix.

Localised modelling. To more accurately model non-

linear deformations, Rabaud *et al.* proposed local LRSM where basis shapes are valid for a shape neighbourhood [30]. Related approaches are based on union of subspaces, *i.e.*, different 3D shapes lie in a disjunction of linear subspaces [43, 23, 24]. Localised modelling is another way to deal with complex deformations. A recent approach of Agudo *et al.* [4] segments surfaces and models deformations of non-overlapping segments with different LRSM.

A particular type of nonlinearities arises in scenarios with large deformations. Piecewise methods [37, 16, 25] address this scenario well. Most of them follow the pipeline of splitting the scene into overlapping patches, reconstructing the patches and eventually imposing global policies to assemble individual patches into the meaningful reconstructions. The local methods, however, do not allow deformation localisation in space, as our approach does. Due to the structure compression, we automatically detect regions with the highest deformability and cluster the structure into the rigid (up to the noise level prior) and non-rigid segments. In computer graphics, a similar technique is used to model localised and human interpretable shape deformations. One example is the work of Neumann *et al.* [28].

Energy-based methods. Energy-based methods increasingly gain the attention in NRSfM [30, 15, 3]. Different energy terms directly express assumptions and soft constraints such as temporal smoothness [30] or spatial regularisation [18]. We propose a new spatial regulariser — coherency term. Similarly to the total variation (TV) [18, 19], it requires a point adjacency data structure but allows for the minimisation at a less computational cost. CT parameterises 3D deformations by a 2D function and imposes coherency on the point depths. The particular form of CT used in this paper is most closely related to the variational regulariser encountered in non-rigid point set registration [27]. In the introductory work [20], we describe CT as applied to dense monocular surface recovery and the core CDF method (without lifting) in greater detail. The emphasis of [20] lies on the handling of large and long occlusions. In contrast, the emphasis of this work lies on HDSM. The proposed HDSM, lifting and expansion operations can be implemented on top of other NRSfM approaches.

Compressed and compact representations. Zhu *et al.* [42] combine LRSM and a compact shape representation in the sense of compressive sensing [17], where every shape is represented as a sparse linear combination of basis shapes (with few nonzero coefficients). Due to the sparsity constraints on the coefficients, the recovered basis shapes were shown to be closer to the elements of the subspace they span. A similar idea is pursued in a more recent work. Assuming 3D structure compressibility, [23] pose NRSfM as a block sparse dictionary learning problem involving L^0 -norm constraint on the coefficient vector. Agudo *et al.* proposed a deformation model with a reduced basis and a di-

rect physical interpretation [1]. Their approach involves spectral analysis on point distance matrices and computes eigenmodes of deformation which are used to model point displacements relative to the estimated shape at rest.

In this paper, we mean structure compression in the sense of compact representation and the theory of data compression [33]. Especially in the case of dense reconstructions, the size of the reconstructions can be prohibitive for embedded and interactive applications. In contrast to the sparse settings, this aspect can not be ignored.

Coarse-to-fine recovery. [10] proposes to estimate basis shapes one at a time imposing the constraint on every new mode to express as much of remaining (more and more high frequent) deformations as possible. The method computes the mean shape and iteratively adds deformation modes. The search for a new mode leads to the energy decrease as much as possible, and this is equivalent to the coarse-to-fine effect. In the proposed L-CDF method, the coarse-to-fine effect is attained through the iterative thresholding. Also, the method can influence the desirable deformation scale. At the same time, we do not have explicit unknowns accounting for the deformation modes.

Geometry lifting. Several approaches employ manifold learning [30, 31, 36] and nonlinear dimensionality reduction (NLDR) [22] techniques. [22] proposed NLDR with a kernel trick. The method defines a nonlinear mapping into a high-dimensional space and uses a set of radial basis functions to locally approximate high non-linearities in 3D. Compared to the manifold-learning and NLDR, a more precise designation for HDSM is *lifting*, and the main motivation is to stay as close as possible to the raw data. HDSM allows to express unions of subspaces and local LRSMs (though, modelling non-linear deformations is not in the scope of this paper); frame-to-frame redundancies can also be modelled (and detected by our method automatically) — all by projecting into different 3D subspaces. One example of applying lifting in computer vision is a framework for tightest convex relaxations of nonconvex energies with a TV term [26]. As applied to NRSfM, the idea of lifted representations remained unexplored in the literature so far, to the best of our knowledge.

3. Notations

For the convenience, we provide in Table 1 a non-exhaustive list of symbols used in this paper. We use the same symbols in Secs. 4 and 5 to denote the same structures, though they refer to different mathematical derivations.

4. High Dimensional Space Model

In this section, we explain the concept of HDSM starting from the rigid orthographic case and subsequently interpret it for the non-rigid case.

Symbol/s	Signification
\mathbf{W}	dense correspondences, measurement matrix
F	number of frames
N, p	number of tracked points
\mathbf{R}, \mathbf{S}	relative camera pose and 3D shape
$\mathbf{\Pi}, \mathbf{P}$	permutation matrix
Φ	lifted geometry (l dimensional)
$\mathbf{U}, \Sigma, \mathbf{V}$	$\text{svd}(\mathbf{X}) = \mathbf{U}\Sigma\mathbf{V}^T$, \mathbf{X} is an arbitrary matrix
$\mathbf{P}(\cdot)$	permutation or shape dimension entangling operator
$\hat{\mathbf{S}}$	shape in the Fourier space

Table 1: A non-exhaustive list of symbols used in the paper.

4.1. Rigid factorisation

Rigid factorisation-based structure from motion [38] is the problem of recovering a rigid 3D scene from 2D observations (point tracks). Such reconstruction represents the same shape in a different pose for every frame. Suppose $\mathbf{W}_{2 \times p}$ is the measurement matrix with p points per frame, $\mathbf{S}_{3 \times p}$ is the observed 3D structure, and $\mathbf{R}_{2 \times 3}$ is the orthographic camera matrix. Then, $\mathbf{W}_{2 \times p} = \mathbf{R}_{2 \times 3}\mathbf{S}_{3 \times p}$. Applying the relative camera pose to \mathbf{S} leads to the complete transfer of the pose to the object:

$$\mathbf{S}(t) = \mathbf{R}'_{3 \times 3}\mathbf{S}_{3 \times p}. \quad (1)$$

We can always use Eq. (1), because of the ambiguity between the camera and object poses. \mathbf{R}' is obtained from \mathbf{R} by adding a third orthonormal row (a cross product of the first two rows). *Our key observation is that the rotated rigid structure $\mathbf{S}(t)$ can be lifted into the 4D space and interpreted as multiple projections of a 4D rigid body Φ into the 3D space under different angles of view (by different 4D to 3D orthographic projection matrices $\mathbf{R}_{3 \times 4}$ in this case).* Formally, we can write:

$$\mathbf{S}(t) = \mathbf{R}_{3 \times 4}\Phi_{4 \times p}. \quad (2)$$

The rigid 4D shape Φ can be found using the similar principles as in the rigid factorisation approach, going from multiple observations in 3D to 4D. A natural question arises, whether it is possible to go from multiple 2D observations directly to a 4D Φ . The further analysis suggests:

$$\mathbf{W}_{2 \times p} = \mathbf{P}_{2 \times 3}\mathbf{R}_{3 \times 4}\Phi_{4 \times p}, \quad (3)$$

where $\mathbf{P}_{2 \times 3} = \mathbf{I}_{2 \times 3}$ is a projection matrix. Factorising $\mathbf{W}_{2 \times p}$ with svd leads to

$$\text{svd}(\mathbf{W}_{2 \times p}) = \mathbf{U}_{2 \times p}\Sigma_{p \times p}^{\frac{1}{2}}\Sigma_{p \times p}^{\frac{1}{2}}\mathbf{V}_{p \times p}^T, \quad (4)$$

with $\Phi = \Sigma_{4 \times p}^{\frac{1}{2}}\mathbf{V}_{p \times p}^T$ and $\mathbf{P}_{2 \times 3}\mathbf{R}_{3 \times 4} = \mathbf{U}_{2 \times p}\Sigma_{4 \times 4}^{\frac{1}{2}}$. From Eq. (4) follows that Φ is ambiguous. Similar to the 3D case, a corrective transformation \mathbf{Q} is required, since an invertible matrix can be inserted between $\mathbf{M} = \mathbf{P}\mathbf{R}$ and Φ :

$$\mathbf{W}_{2 \times p} = \underbrace{\mathbf{P}_{2 \times 3}\mathbf{R}_{3 \times 4}}_{\mathbf{M}}\mathbf{Q}_{4 \times 4}\mathbf{Q}_{4 \times 4}^{-1}\Phi_{4 \times p}. \quad (5)$$

In the rigid case, recovering a 4D representation does not bring advantages. Eq. (5) provides the first evidence in favour of HDSM: *a 4D rigid body may exist so that its projections into the 3D space under different angles can encode rigid transformations of a 3D rigid body.*

4.2. Non-rigid factorisation

In the non-rigid case, HDSM instantiates as follows:

$$\mathbf{W} = \underbrace{\begin{bmatrix} \mathbf{R}_{2 \times 3}^1 & 0 & \dots & 0 \\ 0 & \mathbf{R}_{2 \times 3}^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{R}_{2 \times 3}^F \end{bmatrix}}_{\mathbf{R}_{2F \times 3F}} \underbrace{\begin{bmatrix} \mathbf{P}_{3 \times l}^1 \\ \mathbf{P}_{3 \times l}^2 \\ \vdots \\ \mathbf{P}_{3 \times l}^F \end{bmatrix}}_{\mathbf{\Pi}_{3F \times l}} \underbrace{\begin{bmatrix} \mathbf{\Phi} \\ \vdots \\ \vdots \end{bmatrix}}_{l \times p}. \quad (6)$$

In Eq. (6), $\mathbf{\Phi}$ denotes the high-dimensional structure with l rows and p points, $\mathbf{P}_{3 \times l}^i$ are permutation matrices stacked into the block-matrix $\mathbf{\Pi}$, and $\mathbf{R}_{2 \times 3}^i$ are orthographic projection matrices stacked into \mathbf{R} (quasi-block-diagonal matrix).

$\mathbf{P}_{3 \times l}^i$ always fetch three rows of $\mathbf{\Phi}$. As long as the structure remains unaltered, $\mathbf{\Phi}$ preserves its dimensionality and $\mathbf{P}_{3 \times l}^i$ fetches the same 3D subspace over multiple frames. If deformations occur, $\mathbf{\Phi}$ is expanded, *i.e.*, three additional rows accounting for the new state need to be added to $\mathbf{\Phi}$ in the general case. This might lead to a redundancy as $\mathbf{\Phi}$ may already contain the newly observed structure. On the contrary, most of the newly observed deformations are localised, *i.e.*, they do not affect the whole 3D surface but rather its regions (*e.g.*, some areas can be rigid throughout the entire image sequence). The latter observation allows for further compression of $\mathbf{\Phi}$ (both lossless and lossy).

4.3. LRSM, TSM, FSM and HDSM

Consider the relation between LRSM — the most widely used model for NRSfM — and HDSM. In LRSM, an observed \mathbf{W}_i can be written as [11]

$$\mathbf{W}_i = [c_{i1}\mathbf{R}_i \ c_{i2}\mathbf{R}_i \ \dots \ c_{ik}\mathbf{R}_i][\mathbf{B}_1\mathbf{B}_2 \dots \mathbf{B}_k]^\top = \mathbf{R}_i(c_{i1}\mathbf{B}_1 + c_{i2}\mathbf{B}_2 + \dots + c_{ik}\mathbf{B}_k) = \mathbf{R}_i\mathbf{S}_i, \quad (7)$$

with k basis shapes \mathbf{B}_i and scalar coefficients c_{ij} , $j \in \{1, \dots, k\}$. Stacking all \mathbf{W}_i together leads to

$$\mathbf{W}_i = \mathbf{I}_{2 \times 3}\mathbf{R}_{3 \times 3F}\mathbf{\Pi}_{3F \times 3F}\mathbf{S}_{3F \times p} \Rightarrow \quad (8)$$

$$\begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \\ \vdots \\ \mathbf{W}_F \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{2 \times 3} & 0 & \dots & 0 \\ 0 & \mathbf{I}_{2 \times 3} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{I}_{2 \times 3} \end{bmatrix} \begin{bmatrix} \mathbf{R}_{3 \times 3F}^1 & 0 & \dots & 0 \\ 0 & \mathbf{R}_{3 \times 3F}^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{R}_{3 \times 3F}^F \end{bmatrix} \begin{bmatrix} \mathbf{I}_{3F \times 3F} \\ \mathbf{I}_{3F \times 3F} \\ \vdots \\ \mathbf{I}_{3F \times 3F} \end{bmatrix} \begin{bmatrix} \mathbf{S}_1 \\ \mathbf{S}_2 \\ \vdots \\ \mathbf{S}_F \end{bmatrix}, \quad (9)$$

where $l = 3F$ and $\mathbf{R}_{3 \times 3F}^1 = [\mathbf{R}_1 | 0 \dots 0]$, $\mathbf{R}_{3 \times 3F}^2 = [0 | \mathbf{R}_2 | 0 \dots 0]$. In other words, *a set of 3D shapes stacked together forms a high dimensional rigid body; its projections into different 3D subspaces lead to the initial 3D shapes.* Thus, LRSM can be written and interpreted in terms of HDSM. The form in Eq. (9) represents a special case and will not hold when designing a method based on HDSM.

Akhter *et al.* revealed that TSM is a dual representation of LRSM [9]. In support of this relation, Agudo *et al.* showed that LRSM, TSM and the recently proposed FSM are all dual to each other [2, 5]. *Thus, the connection of HDSM to all three previous models is established.* Besides, it would be perhaps more correct to describe it not as equivacency, but rather a generalisation, since shapes and camera poses in LRSM can always be rearranged to agree with HDSM, but the opposite does not generally hold.

Similarly, the formulation of variational approach (VA) [18] agrees with HDSM, where individual shapes are stacked together into the combined shape matrix \mathbf{S} :

$$\mathbf{W} = \underbrace{\begin{bmatrix} \mathbf{R}_{2 \times 3}^1 & 0 & \dots & 0 \\ 0 & \mathbf{R}_{2 \times 3}^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{R}_{2 \times 3}^f \end{bmatrix}}_{(\mathbf{R}\mathbf{\Pi})_{2F \times 3F}} \underbrace{\begin{bmatrix} \mathbf{S}_1 \\ \mathbf{S}_2 \\ \vdots \\ \mathbf{S}_F \end{bmatrix}}_{\mathbf{\Phi}_{3F \times N}} \quad (10)$$

In VA, the low-rank constraint is imposed in every optimisation step by minimising the nuclear norm of \mathbf{S} . In contrast to VA, we initialise $\mathbf{\Phi}$ without duplication and keep it compact and compressed in every optimisation step.

5. Monocular Surface Recovery with HDSM

We assume that dense correspondences for an image batch are given, and the translation is resolved (an object is centred relative to the camera). To reconstruct a deformable structure from F uncalibrated views, we propose to minimise the energy functional of the form

$$\mathbf{E}(\mathbf{R}, \mathbf{\Pi}, \mathbf{\Phi}, l) = \frac{1}{2} \|\mathbf{W} - \mathbf{R}\mathbf{\Pi}\mathbf{C}^{-1}(\mathbf{\Phi})\|_{\mathcal{F}}^2 + \frac{\lambda}{2} \int_{\mathbb{R}^2} \frac{|\hat{\mathbf{S}}(s)|^2}{\hat{G}(s)} ds, \text{ subject to } \text{rank}(\mathbf{P}(\mathbf{\Pi}\mathbf{C}^{-1}(\mathbf{\Phi}))) = \tau, \quad (11)$$

with C^{-1} standing for an expansion operator, $\hat{\mathbf{S}}$ denoting a Fourier transformed decompressed and expanded surface geometry $\Pi C^{-1}(\Phi)$, and \hat{G} being a Fourier transformed reproducing kernel; \mathcal{F} denotes Frobenius norm, τ and s are a natural number (a parameter) and a Fourier space variable respectively. Let $\mathbf{S} = \Pi C^{-1}(\Phi)$. The operator P entangles dimensions of the expanded shapes:

$$P(\mathbf{S}_{3F \times 3N}) = [\mathbf{S}'_1 \mathbf{S}'_2 \dots \mathbf{S}'_F]_{F \times 3N}^\top, \quad (12)$$

with $\mathbf{S}'_i = (\text{vec}(\mathbf{S}_i^\top))^\top$. Note that we combine discrete and continuous notations in our formulation.

The objective (11) with the rank constraint on $P(\Pi C^{-1}(\Phi))$ is a nonconvex multi-dimensional optimisation problem. As no closed-form solution to this type of problems exists, we minimise it alternately.

At the beginning of every optimisation step, we perform decompression-expansion of Φ . Afterwards, we alternately optimise for \mathbf{R} and \mathbf{S} while fixing \mathbf{S} and \mathbf{R} respectively. Optimisation for \mathbf{R} is performed in a closed form by projecting an unconstrained update to the $\text{SO}(3)$ group by normal equations. Optimisation for \mathbf{S} is carried out by an introduction of an auxiliary variable $\bar{\mathbf{S}}$ and splitting the objective into two subproblems:

$$\min_{\mathbf{S}} \mathbf{E}_1(\mathbf{S}) = \min_{\mathbf{S}} \frac{1}{2\theta} \|\mathbf{S} - \bar{\mathbf{S}}\|_{\mathcal{F}}^2 + \frac{\lambda}{2} \int_{\mathbb{R}^2} \frac{|\hat{\mathbf{S}}(s)|^2}{\hat{G}(s)} ds, \quad (13)$$

$$\begin{aligned} \min_{\bar{\mathbf{S}}} \mathbf{E}_2(\bar{\mathbf{S}}) &= \min_{\bar{\mathbf{S}}} \frac{1}{2\theta} \|\mathbf{S} - \bar{\mathbf{S}}\|_{\mathcal{F}}^2 + \frac{1}{2} \|\mathbf{W} - \mathbf{R}\bar{\mathbf{S}}\|_{\mathcal{F}}^2, \\ \text{s.t. } \text{rank}(P(\bar{\mathbf{S}})) &= \tau. \end{aligned} \quad (14)$$

The subproblem (13) results in a convolution equation which is solved with respect to \mathbf{S} :

$$\mathbf{S} = \mathcal{F}^{-1} \left(\mathcal{F}(\bar{\mathbf{S}}) \circ \frac{\mathcal{F}(G)}{\lambda\theta \mathbf{J}_{m \times n} + \mathcal{F}(G)} \right), \quad (15)$$

where $\mathbf{J}_{m \times n}$ is an all-one matrix and \circ is an elementwise multiplication.

The subproblem (14) is solved for $\bar{\mathbf{S}}$. First, an unconstrained $\bar{\mathbf{S}}'$ is obtained as

$$\bar{\mathbf{S}}' = \left(\frac{1}{\theta} + \mathbf{R}^\top \mathbf{R} \right)^{-1} (\mathbf{R}^\top \mathbf{W} + \frac{1}{\theta} \mathbf{S}). \quad (16)$$

We subsequently impose the rank constraint on $\bar{\mathbf{S}}'$ by preserving τ largest singular values of $P(\bar{\mathbf{S}}')$ and reassembling the matrix. The whole algorithm is summarised in Alg. 1. It includes lifting-compression as well as decompression-expansion operations which are explained further in this section. Please refer to Appendix A for the details on the derivation of Eq. (15).

Lifting-compression of \mathbf{S} . In every iteration, we perform geometry lifting and compression. Taking the expanded updates of \mathbf{S} , we automatically detect frame-to-frame redundancies, *i.e.*, generate an ordered map of pairs

Algorithm 1 Lifted Coherent Depth Fields (L-CDF)

Input: measurements \mathbf{W} , parameters $\lambda, \theta, \tau, \sigma, \mu, \epsilon$

Output: permutation matrix Π and lifted geometry Φ

- 1: **Initialisation:** $[\Pi, \Phi] = C(\mathbf{S}_{\text{rigid}})$ (Alg. 2),
 - 2: $l = 3, \Pi = \mathbf{I}_{3 \times 3} \otimes [1 \ 1 \ 1 \ \dots]^\top \in \mathbb{R}^{F \times 1}$,
 - 3: $\mathbf{R}_{2F \times 3F} = \text{diag} [\mathbf{R}_{2 \times 3}^{1, \text{rigid}} \ \mathbf{R}_{2 \times 3}^{2, \text{rigid}} \ \dots \ \mathbf{R}_{2 \times 3}^{F, \text{rigid}}]$
 - 4: **while** not converge **do**
 - 5: **decompression, Alg. 3:** $\mathbf{S} = \Pi C_\mu^{-1}(\Phi)$
 - 6: **fix \mathbf{S} , update \mathbf{R} :**
 - 7: $\text{svd}(\mathbf{W}\mathbf{S}(\mathbf{S}\mathbf{S}^\top)^{-1}) = U\Sigma V^\top$
 - 8: $\mathbf{R} = UCV^\top$, where
 $C = \text{diag}(1, 1, \dots, 1, \text{sign}(\det(UV^\top)))$
 - 9: **fix \mathbf{R} , update \mathbf{S} :**
 - 10: initialise $\bar{\mathbf{S}} = \mathbf{S}$
 - 11: **while** not converge **do**
 - 12: $U\Sigma V^\top = \text{svd}(P((\frac{1}{\theta}\mathbf{I} + \mathbf{R}^\top \mathbf{R})^{-1}(\frac{1}{\theta}\mathbf{S} + \mathbf{R}^\top \mathbf{W})))$
 - 13: $\bar{\mathbf{S}} = P^{-1}(U\Sigma_{\text{trunc}} V^\top)$
 - 14: $\mathbf{S} = \mathcal{F}^{-1} \left(\mathcal{F}(\bar{\mathbf{S}}) \circ \frac{\mathcal{F}(G)}{\lambda\theta \mathbf{J}_{m \times n} + \mathcal{F}(G)} \right)$
 - 15: **end while**
 - 16: **compression, Alg. 2:** $[\Pi, \Phi] = C_\epsilon(\mathbf{S})$
 - 17: **end while**
-

$x = (\|\mathbf{S}_i\|_{\mathcal{F}}, \mathbf{S}_i)$. Next, we sort x and compare Frobenius norms. If the difference in Frobenius norms for the consecutive shapes is below a scalar value μ , one of the shapes is considered as redundant and not added into Φ' . The correct assignment of projective 3D subspaces and the shapes \mathbf{S}_i is ensured by constantly updating Π_i . Note that the same principles can be applied to the separate dimensions (imagine the case when two shapes share the same x and y coordinates, and differ solely in the z coordinate). Moreover, if only parts of the shape change, the structure can be split into several regions. To obtain \mathbf{S} from Φ' , we need to compute $\Pi \Phi'$.

To compress along the temporal direction, we analyse point trajectories and apply state-to-state compression. Since the shapes are sorted (either in ascending or descending order according to the Frobenius norms), the point trajectories in Φ' are temporally smooth with a high probability. Thus, we save only non-redundant values, up to an ϵ value. The resulting Φ is a sparsified matrix. By saving only non-zero elements, high compression ratios are possible. We denote compression by the operator $C(\cdot)$. An overview of the lift-compress operator is provided in Alg. 2.

Since in our approach, rigid transformations are decoupled from shape deformations, trajectory analysis provides a cue for clustering the points into rigid, nearly rigid and non-rigid subsets. By analysing groups of trajectories, we can perform a *segmentation-from-deformation* and detect localised and correlated deformations.

Decompression-Expansion of Φ . The reverse operator to lift-compress is decompress-expand. It replicates the seed (non-zero) values along the point trajectories in the

Algorithm 2 Lifting-compression of \mathbf{S}

Input: \mathbf{S} , frame-to-frame threshold μ , granularity parameter ϵ **Output:** $[\mathbf{\Pi}, \Phi] = C_\mu(\mathbf{S})$

```

1: Initialisation:  $\mathbf{\Pi} = \mathbf{0}$ 
2: step 1, generate an uncompressed  $\Phi'$ :
3:  $x = (\|\mathbf{S}_i\|_{\mathcal{F}}, \mathbf{S}_i), i \in \{1, \dots, F\}$ 
4: sort  $x$  in ascending order based on the Frobenius norms
5: insert non-redundant frames into  $\Phi'$ , generate  $\mathbf{\Pi}$ 
6: step 2, compress point trajectories:
7: for points  $p \in \{1, \dots, N\}$  do
8:   initialise the seed:  $s = \Phi'(p, 1), \Phi(p, 1) = \Phi'(p, 1)$ 
9:   for point trajectory  $\Phi'(p, j), j \in \{1, \dots, F\}$  do
10:    if  $\|s\| - \|\Phi'(p, j)\| < \epsilon$  then
11:       $\Phi(p, j) = \mathbf{0}_{3 \times 1}$ 
12:    else
13:       $s = \Phi'(p, j), \Phi(p, 1) = \Phi'(p, 1)$ 
14:    end if
15:  end for
16: end for

```

lifted space (decompression) and applies $\mathbf{\Pi}$ to Φ' (expansion). The entire procedure is summarised in Alg. 3.

Algorithm 3 Decompression-Expansion of Φ

Input: compressed lifted geometry Φ , permutation matrix $\mathbf{\Pi}$ **Output:** expanded shape matrix $\mathbf{S} = \mathbf{\Pi} C^{-1}(\Phi)$

```

1: Initialisation:  $\Phi' = \mathbf{0}$ 
2: step 1, point trajectory completion,  $\Phi' = C^{-1}(\Phi)$ :
3: for points  $p \in \{1, \dots, N\}$  do
4:   initialise the seed:  $s = \Phi(p, 1)$ 
5:   for point trajectory  $\Phi(p, j), j \in \{1, \dots, F\}$  do
6:     if  $\Phi(p, j)$  is not a seed then
7:        $\Phi'(p, j) = s$ 
8:     else
9:        $s = \Phi(p, j), \Phi'(p, j) = \Phi(p, j)$ 
10:    end if
11:  end for
12: end for
13: step 2, expansion:  $\mathbf{S} = \mathbf{\Pi} \Phi'$ 

```

Initialisation. We initialise \mathbf{S} under rigidity assumption with [38]. \mathbf{P} initially always fetches \mathbf{S}_1 , *i.e.*, the only available element. The threshold ϵ is a scalar user-specified setting reflecting the noise level expectation, but also serving as a granularity level parameter.

6. Results

The primary purpose of this section consists in showing the validity of HDSM and evaluating L-CDF on synthetic and real world image sequences. All experiments are performed on a server with 32 GB RAM and Intel i7-6700K/4 GHz CPU. We do not require dedicated circuits, though multiple steps of the algorithm can be accelerated on parallel hardware (*e.g.*, matrix multiplications, Fourier trans-

Table 2: Joint average e_{3D} and σ_e for the synthetic faces [18].

	TB [8]	MP [29]	VA [18]	L-CDF
e_{3D} / σ_e	9.24 / 5.37	8.81 / 6.15	3.22 / 0.55	8.03 / 0.98

forms, lift-compress and decompress-expand operators).

Synthetic face sequences. We conduct quantitative evaluation on four synthetic face sequences from [18]. Sequences 1 and 2 depict different facial expressions and are ten frames long each. Sequences 3 and 4 represent interpolated transitions between ten facial expressions and are both 99 frames long. Table 2 reports joint average root-mean-square errors (RMSE) e_{3D} and standard deviation σ_e for the synthetic face sequences and several approaches [29, 9, 18] supporting dense reconstruction, including the proposed L-CPD. RMSE is defined as $e_{3D} = \frac{1}{F} \sum_{f=1}^F \frac{\|\mathbf{S}_f^{ref} - \mathbf{S}_f\|_{\mathcal{F}}}{\|\mathbf{S}_f^{ref}\|_{\mathcal{F}}}$,

with \mathbf{S}_f^{ref} denoting the ground truth surfaces. Since all methods reconstruct the relative camera poses, we register the reconstructions and corresponding \mathbf{S}_f^{ref} with Procrustes analysis. As the point ordering in reconstructions differs across the methods, we use non-rigid point set registration techniques [21] to establish correspondences between reconstructions and \mathbf{S}_f^{ref} . L-CDF achieves the second best result, after VA [18]. Every reconstruction contains $2.9 \cdot 10^4$ points per frame. With 10 alternations (between estimation of \mathbf{R} and \mathbf{S}) and 10 inner primal-dual iterations, the runtime of L-CDF amounts to 985 seconds, with a potential for improvement. In this experiment, we set $\epsilon = 0$ which is equivalent to disabling compression of Φ .

Remark. CDF can explicitly regularise depth values which makes it especially robust against inaccurate correspondences. This comes at the cost of reduced e_{3D} in the cases with clean correspondences. See [20] for further details.

For the case with the activated compression, we set $\lambda = 10^{-1}$, $\theta = 10^{-2}$, $\sigma = 10^{-5}$, $\tau = 15$ and $\mu = 10^{-4}$ in further experiments with the sequences 3 and 4. The value of ϵ varies, and we measure its influence on e_{3D} and compression ratio $\epsilon = \frac{v_{uncomp.}}{v_{comp.}}$, with $v_{uncomp.}$ and $v_{comp.}$ denoting sizes of the uncompressed and compressed dynamic reconstructions respectively. The results are summarised in Fig. 2- (left, middle). With the increase in ϵ from 0.0 to $1.6 \cdot 10^{-3}$, e_{3D} does not noticeably drop. At the same time, ϵ exceeds 2.1 for the sequence 4. Thus, a nearly lossless compression with $\epsilon = 2.1$ is observed. With further increase of ϵ , compression ratios of up to 100 are possible, accompanied by noticeable compression artefacts and, as a result, the drop of e_{3D} . In Fig. 1, several meshed instances of the synthetic face sequence 3 are shown. Compression artifacts in the form of curves on the reconstructed surfaces can be seen with zooming in.

Properties of L-CDF can be appreciated on real and naturalistic image sequences depicting real-world objects.

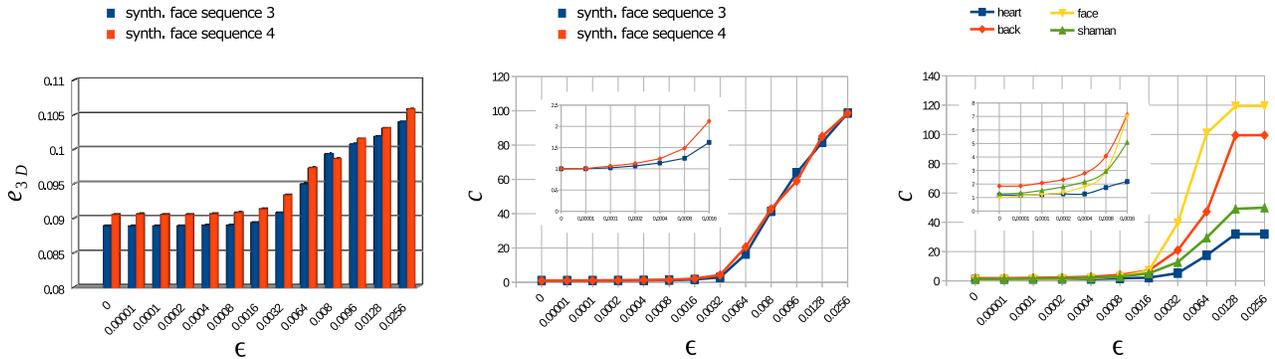


Figure 2: e_{3D} and c values as functions of ϵ , in the experiment with the synthetic faces [18] (left, middle) and real image sequences (right).

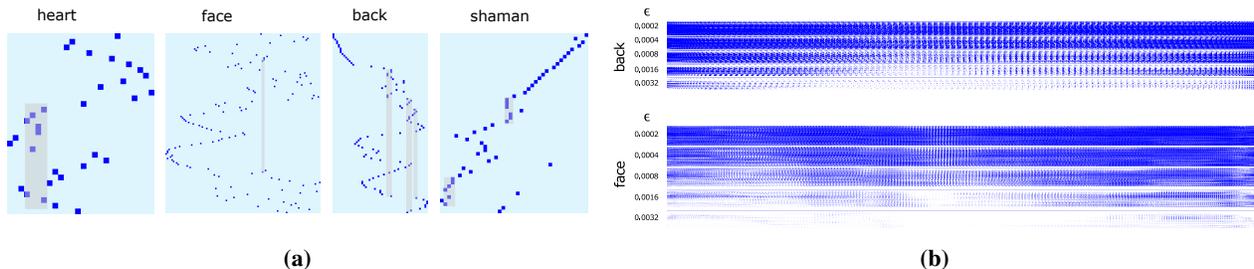


Figure 3: Visualisations of (a): final permutation matrices Π (qualitative evaluation); (b): series of final Φ_s for the face [18] and back [32] sequences.

Those are often prone to frame-to-frame redundancies (either consecutive or repetitive) and self-occlusions. Moreover, often only parts of a surface deform, whereas remaining areas are transformed rigidly. We evaluate L-CDF on three real image sequences, *i.e.*, *face* [18], *back* [32], *heart* [34] as well as *shaman* sequence from the SINTEL dataset [14]. In all cases, correspondences are computed with [35] in a single pass without an explicit occlusion handling. Fig. 2-(right) shows compression ratios as the functions of ϵ , for all four sequences. Because of the real setting, more redundancies — frame-to-frame as well as localised in space — are detected for the same ϵ , compared to the synthetic face sequences. As a result, higher compression ratios for lower ϵ values are observed. An exception is the *heart* sequence, since it deforms entirely up to a few frames in the diastolic phase. However, due to the periodic nature of the deformations and a fixed camera, frame-level compression is possible (recall that similar states do not necessarily need to be consecutive). Furthermore, *back* achieves the highest c , since rigid motion is dominant and non-rigid deformations are rather scarce in this sequence. Compared to the synthetic faces [18], the real *face* and the naturalistic *shaman* achieve somewhat higher c values, as expected. Reconstructions of all sequences are of the same order of magnitude in scale. This allows comparison of the compression ratios achieved for the same ϵ values.

Fig. 3a visualises the final Π matrices for all four image sequences and $\epsilon = 8 \cdot 10^{-4}$. The cyan marks zero entries, and blue stays for $\mathbf{I}_{3 \times 3}$ matrices. Thus, every Π fetches

a $3 \times N$ submatrix of Φ . Note the irregular structure of Π s. Since Φ is a compact lifted representation, Π serves as a key for the assignment of projective 3D subspaces to the frames. Besides, Π s carry information about the frame-level compression of the sequence. The grey bars mark areas with multiple reconstructions originating from the same 3D projections (and thus, detected as redundant by L-CDF, up to the μ parameter). Fig. 3b shows the series of visualised compressed Φ_s for different ϵ values, with the white and blue areas marking zero and non-zero values respectively. Note, first, the repetitive structure of Φ_s , and, second, how Φ_s are sparsified with increasing ϵ . Every point trajectory in Φ reveals how often the respective point remains rigid in the sequence, up to the ϵ parameter. Moreover, by analysing the trajectory patterns, it is possible to detect the areas undergoing similar deformations.

Finally, Fig. 4 shows exemplary reconstructions from all four sequences. Reconstructions are of a high quality, with the advantage of the reduced storage requirement. For the *face* sequence, we visualise point-to-point distances between the uncompressed and compressed reconstructions. With moderately high compression ratios, compression artifacts become more noticeable. Those, however, do not disrupt the general perception of the scene if $c \in \{5, \dots, 30\}$. In the case of $c \approx 80$ and higher, the dynamic scene reduces to almost a single state, with the most of the deformations lost. It is intuitive that in a sequence with, suppose 100 frames, the compression ratio of 100 would imply a single most dominant or an average state preserved.

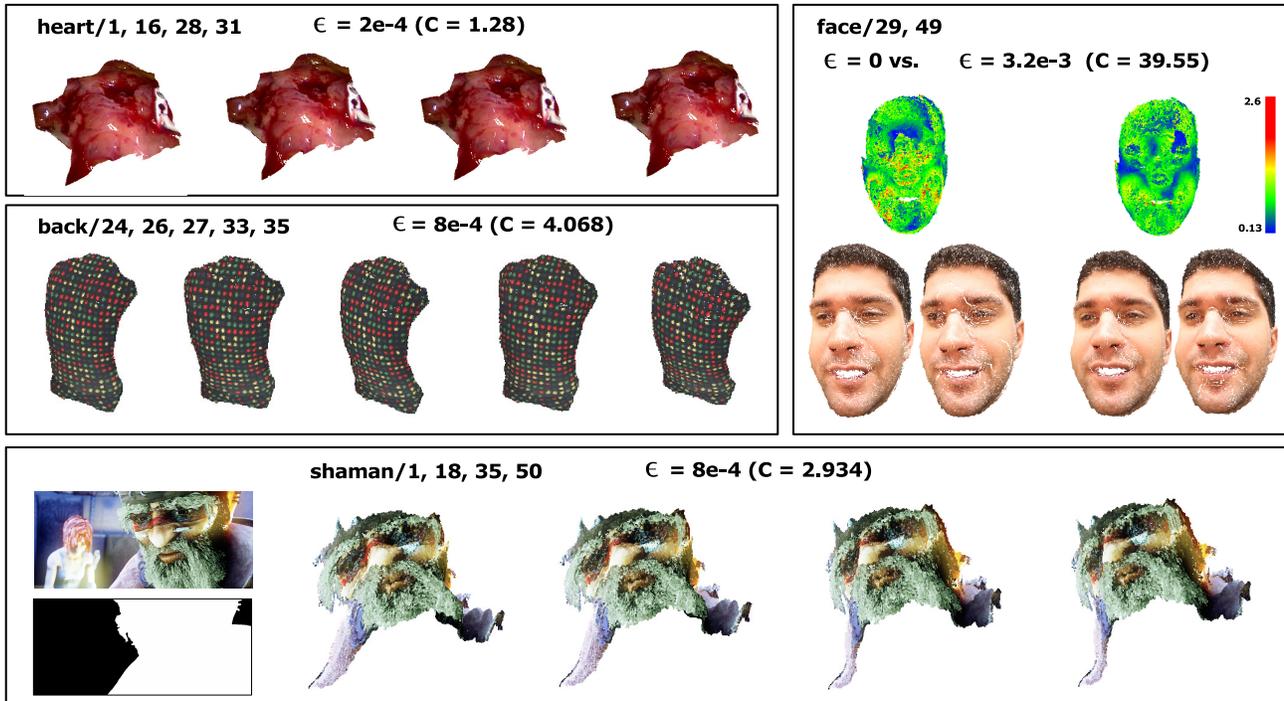


Figure 4: Exemplary reconstructions of real and synthetic image sequences. Frame numbers (comma separated), compression thresholds ϵ and the achieved compression ratios c are listed for every sequence. For the *shaman* sequence [14], the reference frame and the mask are shown on the left. For the *face* [18], point-to-point distances between the uncompressed and compressed reconstructions are visualised using Red>Yellow>Green>Blue colour code.

7. Discussion and Conclusion

We propose a new expressive model for monocular surface recovery with geometry lifting — the HDSM. Besides generalising the previously proposed models for NRSfM — which was shown theoretically — it can serve as a foundation for a practical approach enabling compression of dense non-rigid reconstructions. The proposed variational energy-based L-CDF approach with the new spatial regularizer (coherency term) achieves high reconstruction accuracy and enables nearly lossless compression with the compression ratios of $\approx 5 - 7$ on real image sequences.

Not all properties of HDSM were demonstrated experimentally in this paper. The successor paper will address the remaining parts. A limitation of the current approach is handling of large deformations. HDSM can support those, but this is currently not envisaged and experimentally verified. Thus, HDSM can naturally model a union of subspaces. Another step towards the solution could be an initialisation policy which more accurately approximates the local states. An important direction of the future work is designing an algorithm which does not require expansion, *i.e.*, inferring Φ directly from 2D observations or, in other words, performing updates in the lifted space. A further interesting avenue is to test compressed representations for shape recognition.

A. Appendix: Derivation of Eq. (15)

In this complementary section, we provide details for the minimisation of the energy functional (13). First, we rewrite (13) as

$$\frac{1}{2\theta} \int_{\Omega} |\mathbf{S}(x) - \bar{\mathbf{S}}(x)|^2 dx + \frac{\lambda}{2} \int_{\Omega} \frac{|\hat{\mathbf{S}}(s)|^2}{\hat{G}(s)} ds, \quad (17)$$

with the set of points in the ROI $x \in \Omega$. Note that 1) $\mathbf{S}(x)$, $\bar{\mathbf{S}}(x)$ parameterise 3D deformations by a 2D function, 2) functional (17) updates only the point depths and 3) $\bar{\mathbf{S}}$ is fixed. Next, we perform Fourier transform (FT) of \mathbf{S} :

$$\frac{1}{2\theta} \int_{\Omega} \left| \int_{\mathbb{R}^2} \hat{\mathbf{S}}(s) e^{2\pi i \langle x, s \rangle} ds - \bar{\mathbf{S}}(x) \right|^2 dx + \frac{\lambda}{2} \int_{\Omega} \frac{|\hat{\mathbf{S}}(s)|^2}{\hat{G}(s)} ds. \quad (18)$$

To optimise the energy functional (18) w.r.t. $\hat{\mathbf{S}}$, we take a variational derivative of \mathbf{E} w.r.t. $\hat{\mathbf{S}}(t)$ and equate it to zero:

$$\frac{\partial \mathbf{E}(\hat{\mathbf{S}})}{\partial \hat{\mathbf{S}}(t)} = \frac{1}{\theta} (\mathbf{S}(t) - \bar{\mathbf{S}}(t)) e^{2\pi i \langle x, t \rangle} + \lambda \frac{\hat{\mathbf{S}}(-t)}{\hat{G}(t)} \stackrel{!}{=} 0. \quad (19)$$

Applying inverse FT leads to the convolution equation

$$\mathbf{S}(z) = \frac{1}{\lambda\theta} G(z) * (\bar{\mathbf{S}} - \mathbf{S})(z) \quad (20)$$

which is solved w.r.t. $\hat{\mathbf{S}}$, and the solution after discretisation is given by Eq. (15).

References

- [1] A. Agudo, J.M.M.Montiel, B. Calvo, and F. Moreno-Noguer. Mode-shape interpretation: Re-thinking modal space for recovering deformable shapes. In *Winter Conference on Applications of Computer Vision (WACV)*, 2016. 3
- [2] A. Agudo and F. Moreno-Noguer. Learning shape, motion and elastic models in force space. In *International Conference on Computer Vision (ICCV)*, 2015. 1, 2, 4
- [3] A. Agudo and F. Moreno-Noguer. Simultaneous pose and non-rigid shape with particle dynamics. In *Computer Vision and Pattern Recognition (CVPR)*, 2015. 2
- [4] A. Agudo and F. Moreno-Noguer. Global model with local interpretation for dynamic shape reconstruction. In *Winter Conference on Applications of Computer Vision (WACV)*, 2017. 2
- [5] A. Agudo and F. Moreno-Noguer. Force-based representation for non-rigid shape and elastic model estimation. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2018, to appear. 1, 4
- [6] A. Agudo, F. Moreno-Noguer, B. Calvo, and J. M. M. Montiel. Sequential non-rigid structure from motion using physical priors. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2016. 2
- [7] I. Akhter, Y. Sheikh, and S. Khan. In defense of orthonormality constraints for nonrigid structure from motion. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1534–1541, 2009. 2
- [8] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Nonrigid structure from motion in trajectory space. *Neural Information Processing Systems (NIPS)*, 2008. 6
- [9] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Trajectory space: A dual representation for nonrigid structure from motion. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(7):1442–1456, 2011. 1, 2, 4, 6
- [10] A. Bartoli, V. Gay-Bellile, U. Castellani, J. Peyras, S. Olsen, and P. Sayd. Coarse-to-fine low-rank structure-from-motion. In *Computer Vision and Pattern Recognition (CVPR)*, 2008. 3
- [11] M. Brand. A direct method for 3d factorization of nonrigid motion observed in 2d. In *Computer Vision and Pattern Recognition (CVPR)*, pages 122–128, 2005. 2, 4
- [12] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *Computer Vision and Pattern Recognition (CVPR)*, pages 690–696, 2000. 1, 2
- [13] A. D. Bue. A factorization approach to structure from motion with shape priors. In *Computer Vision and Pattern Recognition (CVPR)*, 2008. 2
- [14] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision (ECCV)*, pages 611–625, 2012. 7, 8
- [15] 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. 2
- [16] J. Fayad, L. Agapito, and A. Del Bue. Piecewise quadratic reconstruction of non-rigid surfaces from monocular sequences. In *European Conference on Computer Vision (ECCV)*, pages 297–310, 2010. 2
- [17] S. Foucart and H. Rauhut. *A Mathematical Introduction to Compressive Sensing*. Birkhäuser Basel, 2013. 2
- [18] R. Garg, A. Roussos, and L. Agapito. Dense variational reconstruction of non-rigid surfaces from monocular video. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1272–1279, 2013. 2, 4, 6, 7, 8
- [19] V. Golyanik, T. Fetzner, and D. Stricker. Accurate 3d reconstruction of dynamic scenes from monocular image sequences with severe occlusions. In *Winter Conference on Applications of Computer Vision (WACV)*, 2017. 2
- [20] V. Golyanik, T. Fetzner, and D. Stricker. Introduction to coherent depth fields for dense monocular surface recovery. In *British Machine Vision Conference (BMVC)*, 2017. 2, 6
- [21] V. Golyanik, B. Taetz, G. Reis, and D. Stricker. Extended coherent point drift algorithm with correspondence priors and optimal subsampling. In *Winter Conference on Applications of Computer Vision (WACV)*, 2016. 6
- [22] P. F. U. Gotardo and A. M. Martinez. Kernel non-rigid structure from motion. In *International Conference on Computer Vision (ICCV)*, pages 802–809, 2011. 2, 3
- [23] C. Kong and S. Lucey. Prior-less compressible structure from motion. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 2
- [24] S. Kumar, Y. Dai, and H. Li. Spatio-temporal union of subspaces for multi-body non-rigid structure-from-motion. *Pattern Recognition*, 2017. 2
- [25] M. Lee, J. Cho, and S. Oh. Consensus of non-rigid reconstructions. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 2
- [26] T. Mollenhoff, E. Laude, M. Moeller, J. Lellmann, and D. Cremers. Sublabel-accurate relaxation of nonconvex energies. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 3
- [27] A. Myronenko and X. Song. Point-set registration: Coherent point drift. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2010. 2
- [28] T. Neumann, K. Varanasi, S. Wenger, M. Wacker, M. Magnor, and C. Theobalt. Sparse localized deformation components. *ACM Trans. Graph.*, 32(6):179:1–179:10, 2013. 2
- [29] M. Paladini, A. Del Bue, J. Xavier, L. Agapito, M. Stosić, and M. Dodig. Optimal metric projections for deformable and articulated structure-from-motion. *International Journal of Computer Vision (IJCV)*, 96(2):252–276, 2012. 2, 6
- [30] V. Rabaud and S. Belongie. Re-thinking non-rigid structure from motion. In *Computer Vision and Pattern Recognition (CVPR)*, 2008. 2, 3
- [31] V. Rabaud and S. Belongie. Linear embeddings in non-rigid structure from motion. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2427–2434, 2009. 3
- [32] C. Russell, J. Fayad, and L. Agapito. Energy based multiple model fitting for non-rigid structure from motion. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3009–3016, 2011. 7
- [33] D. Salomon. *Data Compression: The Complete Reference*. Springer-Verlag New York, Inc., 2006. 3

- [34] D. Stoyanov. Stereoscopic scene flow for robotic assisted minimally invasive surgery. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 479–486, 2012. 7
- [35] B. Taetz, G. Bleser, V. Golyanik, and D. Stricker. Occlusion-aware video registration for highly non-rigid objects. In *Winter Conference on Applications of Computer Vision (WACV)*, 2016. 7
- [36] L. Tao and B. J. Matuszewski. Non-rigid structure from motion with diffusion maps prior. In *Computer Vision and Pattern Recognition (CVPR)*, 2013. 3
- [37] J. Taylor, A. D. Jepson, and K. N. Kutulakos. Non-rigid structure from locally-rigid motion. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2761–2768, 2010. 2
- [38] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision (IJCV)*, 9:137–154, 1992. 3, 6
- [39] L. Torresani, A. Hertzmann, and C. Bregler. Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 30(5):878–892, 2008. 2
- [40] J. Xiao, J. Chai, and T. Kanade. A closed-form solution to non-rigid shape and motion recovery. *International Journal of Computer Vision (IJCV)*, 67(2):233–246, 2006. 2
- [41] A. Yuille and N. Grzywacz. The motion coherence theory. In *International Conference on Computer Vision (ICCV)*, 1988. 2
- [42] S. Zhu, L. Zhang, and B. M. Smith. Model evolution: An incremental approach to non-rigid structure from motion. In *Computer Vision and Pattern Recognition (CVPR)*, 2010. 2
- [43] Y. Zhu, D. Huang, F. De La Torre, and S. Lucey. Complex non-rigid motion 3d reconstruction by union of subspaces. In *Computer Vision and Pattern Recognition (CVPR)*, 2014. 2