Robust Spatio-Temporal Clustering and Reconstruction of Multiple Deformable Bodies

Antonio Agudo and Francesc Moreno-Noguer

Abstract—In this paper we present an approach to reconstruct the 3D shape of multiple deforming objects from a collection of sparse, noisy and possibly incomplete 2D point tracks acquired by a single monocular camera. Additionally, the proposed solution estimates the camera motion and reasons about the spatial segmentation (i.e., identifies each of the deforming objects in every frame) and temporal clustering (i.e., splits the sequence into motion primitive actions). This advances competing work, which mainly tackled the problem for one single object and non-occluded tracks. In order to handle several objects at a time from partial observations, we model point trajectories as a union of spatial and temporal subspaces, and optimize the parameters of both modalities, the non-observed point tracks, the camera motion, and the time-varying 3D shape via augmented Lagrange multipliers. The algorithm is fully unsupervised and does not require any training data at all. We thoroughly validate the method on challenging scenarios with several human subjects performing different activities which involve complex motions and close interaction. We show our approach achieves state-of-the-art 3D reconstruction results, while it also provides spatial and temporal segmentation.

Index Terms—Non-Rigid Structure from Motion, Union of Subspaces, Spatio-Temporal Clustering, Augmented Lagrange Multipliers.

1 Introduction

The problem of Non-Rigid Structure from Motion (NRSfM) involves simultaneously recovering 3D geometry and camera motion from 2D point tracks. When considering deformable or articulated bodies as seen from a monocular camera, many different shape configurations may yield similar image projections. This makes NRSfM a highly ambiguous problem, which requires introducing prior knowledge in order to be solved. The most standard priors include the use of low-rank subspaces constraining the solution space of either the entire shape [3], [40], [56], the 3D point trajectories [10], [50] or the force patterns that induce the deformations [7].

All these previous approaches, consider one single low-rank modality at a time (namely shape, trajectory or force). There are situations, though, that may require models with higher levels of expressiveness, e.g., to represent deformations with complex point trajectories or when dealing with multiple objects, each performing different types of deformations and motions. We here particularly address both these situations with the additional difficulty of partially occluded observations.

There exist previous works addressing in part these scenarios. For the rigid case, for instance, the shape of multiple moving objects can be retrieved by first segmenting the objects from the input 2D tracks and then applying a rigid SfM algorithm to each of them [47], [51], [61]. However, this strategy depends on the accuracy of the initial segmentation which, for the case of non-rigid and overlapping objects is prone to fail. Note also that since the deformations occur in 3D space, in practice it is more intuitive to recover both spatial and temporal segmentations in 3D instead of inferring them on 2D. Regarding the non-rigid case, there has been recent attempts at reconstructing complex dynamics by encoding the time-varying deformation as a sparse 3D shape [38], or by modeling motion as a union of temporal subspaces [62]. Interestingly, [62] also performs temporal clustering. These approaches, however, have been only applied to one single object, and rely on continuous and fully observed 2D point tracks.

In order to reconstruct multiple non-rigid objects with complex motions from partial 2D observations, this paper introduces a novel optimization framework that combines spatial and temporal clustering in a unified manner. The two types of (soft) clustering are performed through affinity matrices, which encode the temporal similarity among the sequence frames and the spatial similarity of the data points within each frame. These matrices are jointly learned, in conjunction with the 3D non-rigid shape, the camera motion and the missing entries, using an efficient Augmented Lagrange Multiplier (ALM) scheme. Hard clustering can be then trivially estimated by applying spectral clustering over the affinity matrices. The overall approach is fully unsupervised and needs no initialization about the deformation and segmentation. Moreover, no a priori knowledge about the dimensionality of the subspaces or which data points belong to which subspace is required.

We extensively evaluate the method on sequences with up to four subjects performing complex actions and interacting with each other. As shown in Fig. 1 the outcome of our algorithm is the spatial segmentation of each frame, which is likely to correspond to each of the subjects, a temporal clustering corresponding to motion

• *The authors are with the Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Barcelona, 08028, Spain. Email: {aagudo, fmoreno}@iri.upc.edu.*

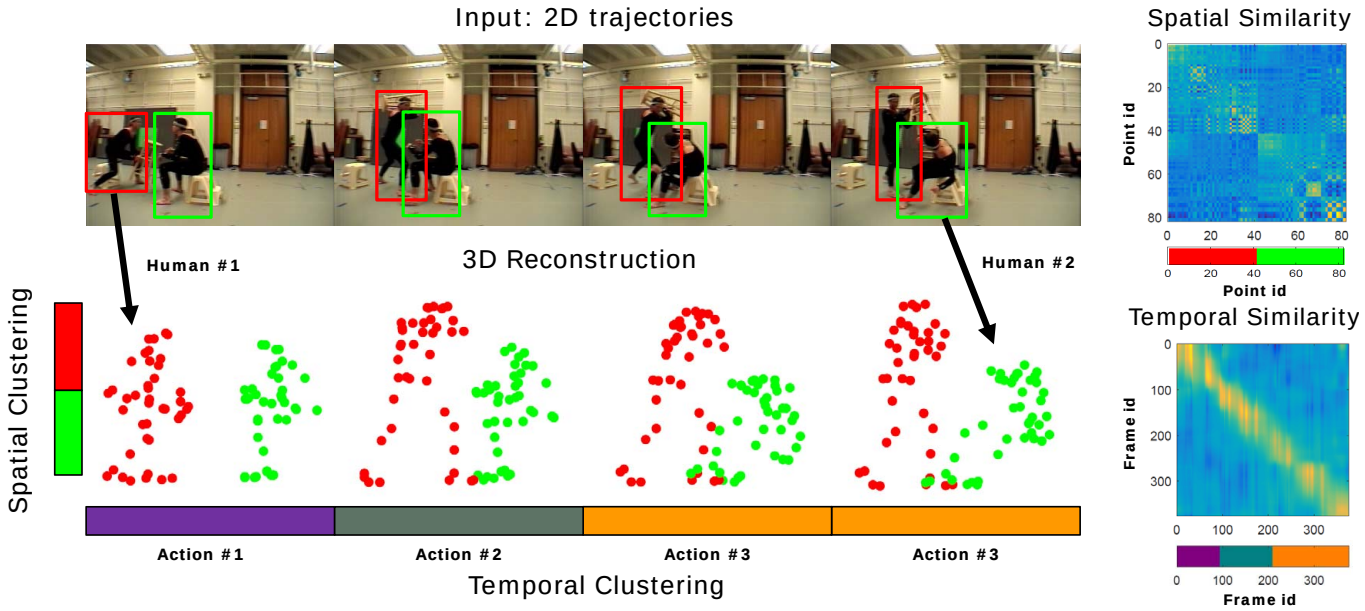

Fig. 1. **Simultaneous 3D non-rigid reconstruction, camera motion, spatial segmentation and temporal clustering from incomplete 2D point tracks. Top-left:** Example of real images from the CMU MoCap dataset. We assume 2D point tracks are provided, although the number of object and point membership is unknown. Point tracks also affected by partial occlusions and strong object overlapping. **Right:** Retrieved spatial and temporal similarity matrices. Each entry in these matrices expresses the spatial/temporal pairwise affinity between points or frames, respectively. Clusters are directly discovered by applying spectral clustering on these matrices. **Bottom-left:** 3D shape reconstruction together with the temporal and spatial clustering results. In this example, spatial segmentation yields two objects, represented by red and green points. Temporal clusters identify three motion primitives which have a clear semantic meaning. In this case, they correspond to 'two subjects sitting down' (magenta), 'one subject standing up an threatening the second one' (green), and 'one subject attacks the other that falls down' (orange). Camera motion is not represented in this figure, but it is also an outcome of our algorithm.

primitives (three action primitives are retrieved for the example shown in the figure), plus the 3D reconstruction of each individual and the corresponding camera motion. We are not aware of any other approach solving the four problems simultaneously solely from incomplete 2D point tracks in a monocular video. Furthermore, as we will show in the results, the accuracy of the 3D reconstructions we obtain, improves that of state-of-the-art NRSfM methods (which do not provide any kind of clustering) by a considerable margin.

A preliminary version of this work was presented in [5], in which we showed our approach to be suitable for jointly recovering time-varying 3D shape, and the spatio-temporal clustering, all of them, from 2D point trajectories in a monocular video. In this work, we extend our contribution enforcing temporal consistency, and to estimate in the same loop all model parameters we consider, including the missing point tracks and the camera motion parameters. Additionally, besides extending the battery of results to further emphasize the advantages of our approach against artifacts such as noisy and missing measurements, we also perform experiments on real data where an unknown number of humans are performing complex tasks, moving, deforming, and even interacting between them (see again Fig. 1).

2 Related Work

The most standard approach to address the inherent ambiguity of the NRSfM problem is by enforcing the underlying 3D shape to lie in a low-rank manifold. In order to estimate such low-rank model, factorization-based approaches have been typically used [6], [15], [26], [31], [34], [48], [56]. Other approaches impose the low-rank constraints by means of robust PCA-like formulations which seek to minimize the rank of a matrix representing the shape. These type of methods either assume the data lies in a single low dimensional space [24], [30], [32], in a union of temporal subspaces [62], or in a manifold defined by a combination of distinct atoms [38]. Piecewise [29] and part-based [41] formulations were also proposed in this context to retrieve strong deformations. On top of these shape models, additional spatial [40] or temporal [2], [12], [42] smoothness constraints have also been considered. Low-rank models were also proposed in the temporal domain, by fitting point trajectories to a series of predefined DCT basis [10], [50], [57] or to an over-complete dictionary learned from human motion data [63], in shape-and-temporal composite domains [35], [36], [54], and in the space of forces that induce the deformations [7].

There exist also a series of works that, instead of relying on low-rank models, reduce the size of the solution space through other types of constraints. One of the most frequently used strategy consists in enforcing inextensibility between every pair of neighboring points [22], [49], [59]. This constrain, though, limits the applicability of these approaches to only isometric deformations. More general deformations (e.g., elastic warps or articulated

bodies) can be recovered through physics-based models [4], [8].

In any event, all previous approaches, have been focused on retrieving the shape of single objects. Most of them, indeed, are not directly applicable to the multi-object scenario we contemplate in this paper, because they rely on a single linear subspace assumption that is not rich enough to model the variability occurring on scenarios with multiple objects performing different actions. Trajectory-based methods [10], [50], [57], can potentially tackle this type of scenarios because the low-rank is applied per point coordinate on the temporal domain. However, as we will show in the results section, a high sensitivity on to the dimension of the low-rank penalizes the accuracy of the reconstructions they provide. Furthermore, none of the previous methods is intended to provide full temporal and spatial segmentation from incomplete 2D point tracks.

Most existing works in multi-object reconstruction from 2D point tracks are applied to rigid objects, and follow a two-step pipeline. First the 2D motion tracks are segmented into several objects using a subspace clustering approach [27], [43]; and then rigid SfM techniques [55] are separately applied to each of the objects [23], [51], [61]. The technique in [47] is able to perform simultaneous segmentation and reconstruction, but it is still only applicable to rigid cases. One interesting exception is the work [53] which assumes the object to be represented as overlapping rigid parts, and simultaneously segments and reconstructs these parts using piecewise rigid models. However, while this approach provides dense (spatial) segmentation and 3D reconstruction, it suffers from the relative low expressiveness of the piecewise models, which limits the applicability to scenes with mild deformations. More recently, the sparse subspace clustering algorithm [28] was extended to solve segmentation and reconstruction by means of a union of spatio-temporal subspaces [39]. Yet, this approach can not handle scenarios with partial object occlusions, limiting its applicability in real cases.

The formulation we introduce in this paper goes a step further from existing approaches in that it simultaneously retrieves 3D non-rigid shape, spatial and temporal clustering, camera motion, and the estimation of missing tracks. To the best of our knowledge, no previous approach has jointly addressed all these problems in a unified framework, and from incomplete 2D trajectories acquired with a monocular camera. Additionally, the spatio-temporal model we propose, allows dealing with objects undergoing complex motions and point track patterns with a high degree of overlapping, in a completely unsupervised manner.

Table 1 summarizes a qualitative comparison of our approach and the aforementioned NRSfM techniques.

Meth. \ Qua.	Rank required	Occlusion handing	Multiple objects	Temporal clustering	Shape clustering
[7], [56], [26]	–	✓	–	–	–
[35], [36]	–	✓	✓	–	–
[24], [32], [38]	✓	–	–	–	–
[30], [40], [41]	✓	✓	–	–	–
[63]	✓	✓	✓	–	–
[62]	✓	–	✓	✓	–
[39]	✓	–	✓	✓	✓
Ours	✓	✓	✓	✓	✓

TABLE 1

Qualitative comparison of our approach against competing NRSfM techniques. The method described in this paper is the only that can jointly provide 3D non-rigid reconstruction, camera motion, temporal clustering, shape segmentation and missing entries estimation. Interestingly, it can naturally handle complex scenarios with multiple interacting objects, without requiring to manually adjust the rank of the basis. Note that when the rank of the basis is required, it usually turns to be a very sensitive parameter for accuracy of the method. Note also that [30] performs shape clustering directly from 2D (rather than 3D), as an independent and separate task previous to the shape reconstruction. Additionally, [63] requires large amounts of 3D human motion data to build a trajectory basis.

3 Revisiting NRSfM

We next revisit the basics on NRSfM which will be used later to build our model to represent non-rigid shape as a union of spatial and temporal subspaces.

Let us consider a time-varying set of N 3D points observed along F frames. We denote by $\mathbf{x}_n^f = [x_n^f, y_n^f, z_n^f]^\top$ the 3D locations of the n-th point at frame f, and by $\tilde{\mathbf{p}}_n^f = [u_n^f, v_n^f]^\top$ its 2D orthographic projection in the image plane. To simplify subsequent formulation, the camera translation $\mathbf{t}^f = \sum_n \tilde{\mathbf{p}}_n^f / N$ is subtracted from the 2D projections, i.e., we consider $\mathbf{p}_n^f = \tilde{\mathbf{p}}_n^f - \mathbf{t}^f$.

We can then write the projection of the 3D points $\{1, \dots, N\}$ onto the sequence of images $\{1, \dots, F\}$ through the following linear system:

$$\underbrace{\begin{bmatrix} \mathbf{p}_1^1 & \dots & \mathbf{p}_N^1 \\ \vdots & \ddots & \vdots \\ \mathbf{p}_1^F & \dots & \mathbf{p}_N^F \end{bmatrix}}_{\hat{\mathbf{P}}} = \underbrace{\begin{bmatrix} \mathbf{R}^1 & & \\ & \ddots & \\ & & \mathbf{R}^F \end{bmatrix}}_{\mathbf{G}} \underbrace{\begin{bmatrix} \mathbf{x}_1^1 & \dots & \mathbf{x}_N^1 \\ \vdots & \ddots & \vdots \\ \mathbf{x}_1^F & \dots & \mathbf{x}_N^F \end{bmatrix}}_{\hat{\mathbf{X}}}, \quad (1)$$

where $\hat{\mathbf{P}}$ is a $2F \times N$ matrix storing the 2D point measurements arranged column-wise, $\mathbf{G}$ is a $2F \times 3F$ block diagonal matrix, made of the F truncated 2×3 camera rotations $\mathbf{R}^f$, and $\hat{\mathbf{X}}$ is a $3F \times N$ matrix with the 3D positions of the points along the whole sequence, also arranged column-wise. The NRSfM problem can then be formulated as that of recovering the non-rigid structure $\hat{\mathbf{X}}$ and camera motion $\mathbf{G}$ from 2D point trajectories $\hat{\mathbf{P}}$.

Early NRSfM solutions [10], [25], [35], [60] based on the factorization method [15], constrained the matrix $\hat{\mathbf{X}}$ to be low rank. These methods solved the factorization by through distinct types of basis elements, such as shape basis [25], [60], pre-defined trajectory basis [10], shape-trajectory formulations [35] and force-induced basis [7]. For a given number K of bases, it was shown that rank($\hat{\mathbf{X}}$) was $\leq 3K$. The time-varying shape could then be estimated applying a rank-$3K$ factorization over $\hat{\mathbf{P}}$, in combination with constraints enforcing rotation

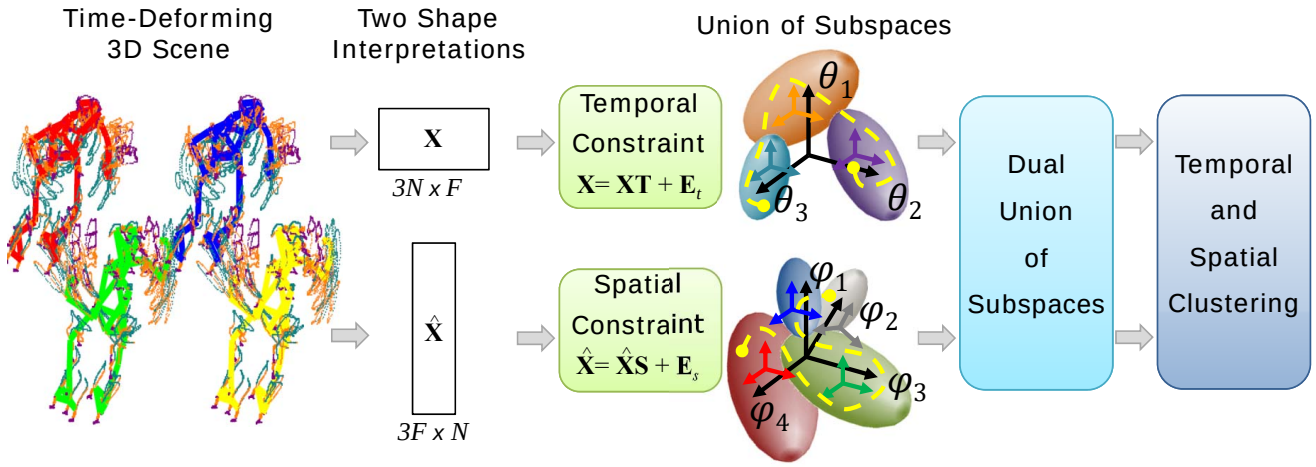

Fig. 2. **Dual Union of Spatio-Temporal Subspaces.** Schematic representation of a scenario with four subjects that are non-rigidly moving and interacting. Recall that the spatial segmentation and number of subjects is initially unknown. The 4D information can be encoded using two different interpretations given by matrices $\mathbf{X}$ and $\hat{\mathbf{X}}$ (see section 4). Post-multiplying these matrices by affinities $\mathbf{T}$ and $\mathbf{S}$, respectively, allows introducing temporal and spatial constraints, and obtaining the corresponding spatio-temporal clustering by means of spectral analysis. Additionally, these matrices are enforced to be low rank, being the rank of every subspace also unknown. This means that each temporal and spatial cluster (depicted by color ellipsoids and vectors), is in turn represented by a union of subspaces (indicated by black vectors $\boldsymbol{\theta}_i$ and $\boldsymbol{\varphi}_i$). The proposed Dual Union of Spatio-Temporal Subspaces (DUST) model, combines the two types of subspaces, and it can encode a wider solution space in both temporal and spatial domains, as shown by the yellow line.

orthonormality [9]. However, despite their popularity, these methods are very sensitive to the value of the rank, which needs to be carefully chosen to obtain accurate results.

More recently, several approaches have enforced the low-rank constraint of the time-varying shape by applying nuclear norm minimization directly over the matrix encoding the 3D point positions [24], [30], [32]. Following [24], [37], the elements of $\hat{\mathbf{X}}$ in Eq. (1) can be rearranged into a new $3N \times F$ matrix $\mathbf{X}$ encoding the x, y and z coordinates in different rows:

$$\mathbf{X} = \begin{bmatrix} x_1^1 & \dots & x_N^1 & y_1^1 & \dots & y_N^1 & z_1^1 & \dots & z_N^1 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_1^F & \dots & x_N^F & y_1^F & \dots & y_N^F & z_1^F & \dots & z_N^F \end{bmatrix}^\top .$$

Observe that the components of $\mathbf{X}$ and $\hat{\mathbf{X}}$ are exactly the same, but in a different manner. The interest of using matrix $\mathbf{X}$ arrangement is that under a low-rank representation with K shape bases, it retains the rank K (in contrast to $\hat{\mathbf{X}}$, which was $3K$). Therefore, $\mathbf{X}$ naturally captures the fact that it is represented by a K-order linear model and avoids spurious degrees of freedom while allowing to learn redundancies between frames.

In the following sections, both $\mathbf{X}$ and $\hat{\mathbf{X}}$ matrices will be used. In order to map one matrix to the other we define a function q such that $\hat{\mathbf{X}} = q(\mathbf{X}) = (\mathbf{I}_3 \otimes \mathbf{X}^\top)\mathbf{A}_{3D}$, where $\mathbf{A}_{3D}$ is a $9N \times N$ binary matrix, $\mathbf{I}_3$ is an identity matrix of rank 3, and $\otimes$ represents the Kronecker product operator. Similarly, we define the inverse mapping by means of $\mathbf{X} = q^{-1}(\hat{\mathbf{X}}) = (\hat{\mathbf{X}}^\top \otimes \mathbf{I}_3)\mathbf{B}_{3D}$, where $\mathbf{B}_{3D}$ is a $9F \times F$ binary matrix.

In a similar way, we will find useful to define a function d to relate the 2D measurement matrix $\hat{\mathbf{P}}$ in Eq. (1), with a new arrangement $\mathbf{P} \in 2N \times F$, such that $\hat{\mathbf{P}} = d(\mathbf{P}) = (\mathbf{I}_2 \otimes \mathbf{P}^\top)\mathbf{A}_{2D}$, where $\mathbf{A}_{2D}$ is a $4N \times N$ binary matrix. The inverse mapping is defined as $\mathbf{P} = d^{-1}(\hat{\mathbf{P}}) = (\hat{\mathbf{P}}^\top \otimes \mathbf{I}_2)\mathbf{B}_{2D}$, where $\mathbf{B}_{2D}$ is a $4F \times F$ binary matrix.

4 Dual Union of Spatio-Temporal Subspaces

As we have just described, a time-varying shape can be either represented by the matrices $\mathbf{X}$ or $\hat{\mathbf{X}}$. Even though the two matrices are made by exactly the same elements, they permit two different types of interpretations. From one side, following [45], [62], when considering the $\mathbf{X}$-arrangement, we can define a temporal clustering over the shapes through a temporal affinity $F \times F$ matrix $\mathbf{T}$:

$$\mathbf{X} = \mathbf{X}\mathbf{T} + \mathbf{E}_t \,, \tag{2}$$

where the affinity matrix $\mathbf{T}$ measures the similarity between image frames. As we shall see later, once this matrix is learned from data, spectral clustering algorithms [20] can be applied on it to discover and match different motion primitives within the video sequence. $\mathbf{E}_t$ is a residual noise, which is essential to avoid the trivial solution $\mathbf{T} \equiv \mathbf{I}_F$.

On the other hand, we can also consider performing spatial segmentation by means of a so-called spatial affinity $N \times N$ matrix $\mathbf{S}$, which in this case, is applied on the matrix $\hat{\mathbf{X}}$:

$$\hat{\mathbf{X}} = \hat{\mathbf{X}}\mathbf{S} + \mathbf{E}_s \,, \tag{3}$$

where $\mathbf{E}_s$ is a residual noise. In this case, the affinity matrix encodes point similarity, and again, once it its

learned, we can use spectral clustering on it to spatially segment the data and split it into different objects.

Equations (2) and (3) can be interpreted as a representation of the time-varying 3D points using a union of temporal and spatial subspaces, respectively. In the following section we will simultaneously apply the two types of representations, i.e., we will jointly merge two unions of subspaces, and hence the name of *Dual Union of Spatio-Temporal subspaces* (DUST) we give to our approach. A schematic representation of our model is illustrated in Fig. 2.

5 Enforcing Temporal Smoothness

As we shall see in the following section, the formulation we propose allows easily introducing smoothness constraints to the reconstructed shapes. For this purpose, we consider the hard constraint $\mathbf{XQ} = \mathbf{0}$, where $\mathbf{Q}$ is a $F \times F$ matrix encoding temporal smoothness priors. Specifically, we use second-order central differences, i.e., we enforce $2\mathbf{x}_n^f - \mathbf{x}_n^{f-1} - \mathbf{x}_n^{f+1} \approx \mathbf{0}$, which in matrix form can be written as:

$$\mathbf{Q}_{kj} = \begin{cases} 2 & \text{if } j = k, k = \{2, \dots, F-1\} \\ 1 & \text{if } j = k, k = \{1, F\} \\ -1 & \text{if } j|k = k|j+1, k|j = \{1, \dots, F-1\} \\ 0 & \text{if otherwise} \end{cases} \quad (4)$$

In the results section, we will see this constraint yields remarkable benefits, with almost no additional cost.

6 3D Shape and Spatio-Temporal Clustering from 2D Tracks

In this section we combine the geometric projection constraint described in Section 3, together with Eqs. (2) and (3) enforcing temporal and spatial clustering, respectively, in order to simultaneously segment the 2D trajectories into different objects, estimate their time-varying 3D shape and camera motion, and cluster their motion into a series of primitives. It is worth pointing that [62] already presented an approach to perform reconstruction and temporal grouping of one single object. Here, we introduce the multi-object capability, a strategy to recover the camera motion, and the possibility to handle occluded 2D tracks. As it will be shown, this involves having to deal with a considerably more complex loss function and a more elaborate optimization strategy than that considered in [62].

6.1 Problem Formulation

Let $\bar{\mathbf{P}}$ be a possibly incomplete 2D measurement matrix, and $\mathbf{O}$ its corresponding $F \times N$ observation matrix with $\{1, 0\}$ entries indicating whether the coordinates of a point in a specific frame are observed or not.

We can specifically formulate our problem as follows: given the incomplete 2D tracks $\bar{\mathbf{P}}$ and the observation matrix $\mathbf{O}$, we seek to retrieve the temporal 3D location of all points $\hat{\mathbf{X}}$, the affinity matrices associated to soft temporal $\mathbf{T}$ and spatial $\mathbf{S}$ clustering for spatio-temporal segmentation, the matrix $\hat{\mathbf{P}}$ of complete 2D tracks, and the matrix $\mathbf{G}$ of camera rotations. Let us denote by $\mathbf{\Theta} \equiv \{\hat{\mathbf{P}}, \mathbf{G}, \mathbf{T}, \mathbf{S}, \mathbf{X}, \mathbf{E}_t, \mathbf{E}_s\}$ the set of all model parameters.

For estimating these parameters we introduce a cost function that incorporates the spatio-temporal model described previously and enforces the model matrices to lie in low-rank subspaces [1]. Since rank minimization is a non-convex NP-hard problem [52], the nuclear norm is used as a convex relaxation [19], [21]. In order to be able to deal with data corrupted by noise and outliers, we use l_1-norm regularization as the convex relaxation of the l_∞-norm [46]. Finally, our problem can therefore be written as follows:

$$\begin{aligned} \underset{\mathbf{\Theta}}{\arg\min} \; & \| (\mathbf{O} \otimes \mathbf{1}_2) \odot \left(\hat{\mathbf{P}} - \bar{\mathbf{P}} \right) \|_F^2 + \beta \|\hat{\mathbf{P}}\|_* + \phi \|\mathbf{T}\|_* \\ & + \phi \|\mathbf{S}\|_* + \gamma \|\mathbf{X}\|_* + \lambda_t \|\mathbf{E}_t\|_1 + \lambda_s \|\mathbf{E}_s\|_1 \quad (5) \\ \text{subject to} \; & \hat{\mathbf{P}} = \mathbf{G}\hat{\mathbf{X}} \\ & \mathbf{I}_{2F} = \mathbf{G}\mathbf{G}^\top \\ & \mathbf{X} = \mathbf{X}\mathbf{T} + \mathbf{E}_t \\ & \hat{\mathbf{X}} = \hat{\mathbf{X}}\mathbf{S} + \mathbf{E}_s \\ & \mathbf{X}\mathbf{Q} = \mathbf{0} \end{aligned}$$

where $\odot$ represents the Hadamard product and $\mathbf{1}$ is a vector of ones. $\|\cdot\|_*$ is the nuclear norm, $\|\cdot\|_1$ is the convex approximation to sparse error and $\|\cdot\|_F$ indicates the Frobenius norm. $\{\beta, \phi, \gamma, \lambda_t, \lambda_s\}$ are penalty term parameters.

In this paper we will introduce two algorithms to minimize the cost function in Eq. (5). First, following most state-of-the-art factorization techniques [10], [26], [35], [24], [40], we will propose an approximated three-step strategy in which: 1) the partially observed measurement matrix $\hat{\mathbf{P}}$ is completed, 2) the camera rotations matrix $\mathbf{G}$ is estimated and, 3) the shape $\mathbf{X}$ and clustering parameters $\mathbf{T}$ and $\mathbf{S}$ are simultaneously recovered. This algorithm is a temporally-smoothed version of our early approach DUST [5], hence denoted DUST-TS, and will be described in section 6.2. The second approach we propose is also an iterative formulation, but in this case all model parameters are jointly estimated in the same loop, instead of having to perform three different steps. This algorithm (denoted "DUST2-TS") will be described in section 6.3.

6.2 DUST-TS: A Three-Step Factorization Strategy

We next describe the three main steps of the DUST-TS algorithm.

1. The low rank constraint needs to be enforced to both the affinity matrices $\mathbf{T}$ and $\mathbf{S}$ and to the shape matrix $\mathbf{X}$. Enforcing low rank only to the affinity matrices does not guarantee the shape matrices will be low rank. For instance, in the temporal case we have that $\text{rank}(\mathbf{X}) \equiv \text{rank}(\mathbf{XT} + \mathbf{E}_t) \leq \text{rank}(\mathbf{XT}) + \text{rank}(\mathbf{E}_t) = \min\{\text{rank}(\mathbf{X}), \text{rank}(\mathbf{T})\} + \text{rank}(\mathbf{E}_t)$. That is, even if $\mathbf{T}$ is low rank, $\mathbf{X}$ may not be due to the presence of the noise term $\mathbf{E}_t$, and we need also so explicitly enforce the low rank on $\mathbf{X}$.

input : Possibly incomplete 2D trajectories $\bar{\mathbf{P}}$ and parameters $\{\lambda_t, \lambda_s, \gamma\}$
output: 3D reconstruction $\hat{\mathbf{X}}$ or $\mathbf{X}$, camera rotation $\mathbf{G}$, spatial $\mathbf{S}$ and temporal $\mathbf{T}$ clustering, and full 2D trajectories $\hat{\mathbf{P}}$

/* Complete 2D Trajectories, Eq. (7) */
if $\hat{\mathbf{P}} \neq \bar{\mathbf{P}}$ **then**
$\hat{\mathbf{P}} = \min \| (\mathbf{O} \otimes \mathbf{1}_2) \odot \left(\hat{\mathbf{P}} - \bar{\mathbf{P}}\right) \|_F^2 + \beta \|\hat{\mathbf{P}}\|_*$
else $\hat{\mathbf{P}} \equiv \bar{\mathbf{P}}$
/* Camera Rotation $\mathbf{G}$, Eq. (8) */
/* ALM Optimization of Eq. (10) */
while *not converged* **do**
/* Update Model Parameters */
$\mathbf{J} = \min \frac{1}{\alpha}\|\mathbf{J}\|_* + \frac{1}{2}\|\mathbf{J} - (\mathbf{T} + \frac{\mathbf{L}_6}{\alpha})\|_F^2$
$\mathbf{T} = (\mathbf{X}^\top\mathbf{X} + \mathbf{I}_F)^{-1}(\mathbf{X}^\top(\mathbf{X} - \mathbf{E}_t) + \mathbf{J} + \frac{\mathbf{X}^\top\mathbf{L}_1 - \mathbf{L}_6}{\alpha})$
$\mathbf{K} = \min \frac{1}{\alpha}\|\mathbf{K}\|_* + \frac{1}{2}\|\mathbf{K} - (\mathbf{S} + \frac{\mathbf{L}_7}{\alpha})\|_F^2$
$\mathbf{S} = (\hat{\mathbf{X}}^\top\hat{\mathbf{X}} + \mathbf{I}_N)^{-1}(\hat{\mathbf{X}}^\top(\hat{\mathbf{X}} - \mathbf{E}_s) + \mathbf{K} + \frac{\hat{\mathbf{X}}^\top\mathbf{L}_3 - \mathbf{L}_7}{\alpha})$
$\mathbf{X} = \min \frac{\gamma}{\alpha}\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X} - \big((\mathbf{E}_t - \frac{\mathbf{L}_1}{\alpha})\mathbf{N}^\top + q^{-1}(\hat{\mathbf{X}} - \frac{\mathbf{L}_4}{\alpha}) - (\frac{\mathbf{L}_5}{\alpha})\mathbf{Q}^\top\big)(\mathbf{N}\mathbf{N}^\top + \mathbf{I}_F + \mathbf{Q}\mathbf{Q}^\top)^{-1}\|_F^2$
$\mathbf{C} = \mathbf{G}^\top(\hat{\mathbf{P}} + \frac{\mathbf{L}_2}{\alpha}) + (\mathbf{E}_s - \frac{\mathbf{L}_3}{\alpha})(\mathbf{I}_N - \mathbf{S}^\top) + \frac{\mathbf{L}_4}{\alpha} + q(\mathbf{X})$
$\text{vec}(\hat{\mathbf{X}}) = \big(\mathbf{I}_N \otimes (\mathbf{G}^\top\mathbf{G} + \mathbf{I}_H) + \mathbf{H}^\top \otimes \mathbf{I}_H\big)^{-1}\text{vec}(\mathbf{C})$
$\hat{\mathbf{X}} = \text{mat}(\text{vec}(\hat{\mathbf{X}}))$
$\mathbf{E}_t = \min \frac{\lambda_t}{\alpha}\|\mathbf{E}_t\|_1 + \frac{1}{2}\|\mathbf{E}_t - (\mathbf{X} - \mathbf{X}\mathbf{T} + \frac{\mathbf{L}_1}{\alpha})\|_F^2$
$\mathbf{E}_s = \min \frac{\lambda_s}{\alpha}\|\mathbf{E}_s\|_1 + \frac{1}{2}\|\mathbf{E}_s - (\hat{\mathbf{X}} - \hat{\mathbf{X}}\mathbf{S} + \frac{\mathbf{L}_3}{\alpha})\|_F^2$
/* Update Lagrange Multipliers */
$\mathbf{L}_1 = \mathbf{L}_1 + \alpha(\mathbf{X} - \mathbf{X}\mathbf{T} - \mathbf{E}_t)$
$\mathbf{L}_2 = \mathbf{L}_2 + \alpha(\hat{\mathbf{P}} - \mathbf{G}\hat{\mathbf{X}})$
$\mathbf{L}_3 = \mathbf{L}_3 + \alpha(\hat{\mathbf{X}} - \hat{\mathbf{X}}\mathbf{S} - \mathbf{E}_s)$
$\mathbf{L}_4 = \mathbf{L}_4 + \alpha(q(\mathbf{X}) - \hat{\mathbf{X}})$
$\mathbf{L}_5 = \mathbf{L}_5 + \alpha(\mathbf{X}\mathbf{Q} - \mathbf{0})$
$\mathbf{L}_6 = \mathbf{L}_6 + \alpha(\mathbf{T} - \mathbf{J})$
$\mathbf{L}_7 = \mathbf{L}_7 + \alpha(\mathbf{S} - \mathbf{K})$
/* Update Penalty Weights */
$\alpha = \min(\rho\alpha, 10^{12})$
/* Check Convergence */
$\|\mathbf{X} - \mathbf{X}\mathbf{T} - \mathbf{E}_t\|_\infty < \epsilon$
$\|\hat{\mathbf{P}} - \mathbf{G}\hat{\mathbf{X}}\|_\infty < \epsilon$
$\|\hat{\mathbf{X}} - \hat{\mathbf{X}}\mathbf{S} - \mathbf{E}_s\|_\infty < \epsilon$
$\|q(\mathbf{X}) - \hat{\mathbf{X}}\|_\infty < \epsilon$
$\|\mathbf{X}\mathbf{Q} - \mathbf{0}\|_\infty < \epsilon$
$\|\mathbf{T} - \mathbf{J}\|_\infty < \epsilon$
$\|\mathbf{S} - \mathbf{K}\|_\infty < \epsilon$
end
Not.: $\mathbf{H} = (\mathbf{I}_N - \mathbf{S})(\mathbf{I}_N - \mathbf{S}^\top)$, $H = 3T$, $\mathbf{N} = \mathbf{I}_F - \mathbf{T}$.
Hyper-parameters: $\rho = 1.1$, $\epsilon = 10^{-7}$, $\alpha = 10^{-2}$.
Matrices $\mathbf{L}_c$, $c = \{1, \ldots, 7\}$, are initially set to zero.

Algorithm 1: DUST-TS algorithm for optimizing Eq. (5). vec$(\cdot)$ and mat$(\cdot)$ are vectorization and matrization operators.

6.2.1 Completing Missing Entries

To complete the unobserved tracks identified as zeros within the observation matrix $\mathbf{O}$, we separately optimize $\hat{\mathbf{P}}$ taking the first two terms of Eq. (5) and enforcing a low-rank constraint about the measurement matrix:

$$\min_{\hat{\mathbf{P}}} \quad \| (\mathbf{O} \otimes \mathbf{1}_2) \odot \left(\hat{\mathbf{P}} - \bar{\mathbf{P}}\right) \|_F^2 + \beta \|\hat{\mathbf{P}}\|_* \quad . \tag{6}$$

As it was shown in [11], [16], [17], this type of low-rank minimizations with the nuclear norm acting as a regularizer can be optimized with a bilinear factorization $\hat{\mathbf{P}} = \mathbf{U}\mathbf{V}^\top$ and applying Augmented Lagrange Multipliers (ALM) [14]. By doing this, we obtain the following augmented Lagrangian function:

$$\begin{aligned}\underset{\hat{\mathbf{P}},\mathbf{U},\mathbf{V}}{\arg\min} \; & \| (\mathbf{O} \otimes \mathbf{1}_2) \odot \left(\hat{\mathbf{P}} - \bar{\mathbf{P}}\right) \|_F^2 + \frac{\beta}{2}\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right) \\ & + \langle \mathbf{L}, \hat{\mathbf{P}} - \mathbf{U}\mathbf{V}^\top \rangle + \frac{\alpha}{2}\|\hat{\mathbf{P}} - \mathbf{U}\mathbf{V}^\top\|_F^2 \quad , \end{aligned} \tag{7}$$

where $\mathbf{L}$ is the $2F \times N$ Lagrange multiplier and $\alpha > 0$ a penalty parameter. The Euclidean inner product between two matrices is defined as $\langle \mathbf{X}, \mathbf{Z} \rangle = \text{tr}(\mathbf{X}^\top\mathbf{Z})$, where tr$(\cdot)$ is the trace of a matrix. We solve this optimization following the algorithm described in [17] (see Algorithm #1 in [17]).

6.2.2 Estimating Camera Rotation

Given the full matrix of point tracks $\hat{\mathbf{P}}$ estimated in subsection 6.2.1, the camera rotation matrices $\mathbf{R}$, i.e., the matrix $\mathbf{G}$, can be estimated independently from the rest of model parameters by factorization. For this purpose we write the following non-convex optimization problem which accounts for both 3D-to-2D projection consistency plus orthonormality constraints on the estimated rotation matrices:

$$\underset{\mathbf{R}^f}{\arg\min} \frac{1}{2}\|\hat{\mathbf{P}} - \mathbf{G}\hat{\mathbf{X}}\|_{\mathcal{F}}^2 + \sum_{f=1}^{F} \|\mathbf{I}_2 - \mathbf{R}^f\mathbf{R}^{f\top}\|_{\mathcal{F}}^2 \; . \tag{8}$$

There exist several approximations to solve Eq. (8), e.g., strategies that enforce smooth trajectories [10], [35], [36], methods based on trace-norm minimization that assume the rank of the subspace a priori [24], [54] or techniques based on Procrustes analysis [40]. Alternatively, when the non-rigid objects do not undergo a rigid motion, the camera motion matrix $\mathbf{G}$ could also be recovered using a few background rigid points [50], and then applying rigid factorization [55]. In this paper, following [35], Eq. (8) is solved by applying an initial SVD factorization over $\hat{\mathbf{P}}$, and then enforcing metric constraints by non-linear optimization.

6.2.3 Joint Clustering and 3D Reconstruction

In order to jointly recover 3D shape and the spatio-temporal clustering, we again resort to the ALM method. Assuming $\hat{\mathbf{P}}$ and $\mathbf{G}$ are already known, the minimization we need to perform is:

$$\begin{aligned}\underset{\mathbf{T},\mathbf{S},\mathbf{X},\mathbf{E}_t,\mathbf{E}_s}{\arg\min} \; & \phi\|\mathbf{T}\|_* + \phi\|\mathbf{S}\|_* + \gamma\|\mathbf{X}\|_* + \lambda_t\|\mathbf{E}_t\|_1 + \lambda_s\|\mathbf{E}_s\|_1 \\ \text{subject to} \; & \hat{\mathbf{P}} = \mathbf{G}\hat{\mathbf{X}} \\ & \mathbf{X} = \mathbf{X}\mathbf{T} + \mathbf{E}_t \\ & \hat{\mathbf{X}} = \hat{\mathbf{X}}\mathbf{S} + \mathbf{E}_s \\ & \mathbf{X}\mathbf{Q} = \mathbf{0}\end{aligned}$$

Since the parameters $\{\phi, \gamma, \lambda_t, \lambda_s\}$ can be scaled with respect to one of them, in the following, without loss of

generality, we fix $\phi = 1$. Finally, the Lagrangian function can be written as:

$$\underset{\boldsymbol{\Theta}_{\text{DUST-TS}}}{\arg\min} \quad \{\text{CostDUST-TS}\} \tag{9}$$

where:

$$\begin{aligned}\text{CostDUST-TS} =& \|\mathbf{J}\|_* + \|\mathbf{K}\|_* + \gamma\|\mathbf{X}\|_* + \lambda_t\|\mathbf{E}_t\|_1 + \lambda_s\|\mathbf{E}_s\|_1 \\ &+ \langle \mathbf{L}_1, \mathbf{X} - \mathbf{X}\mathbf{T} - \mathbf{E}_t \rangle + \frac{\alpha}{2}\|\mathbf{X} - \mathbf{X}\mathbf{T} - \mathbf{E}_t\|_F^2 \\ &+ \langle \mathbf{L}_2, \hat{\mathbf{P}} - \mathbf{G}\hat{\mathbf{X}} \rangle + \frac{\alpha}{2}\|\hat{\mathbf{P}} - \mathbf{G}\hat{\mathbf{X}}\|_F^2 \\ &+ \langle \mathbf{L}_3, \hat{\mathbf{X}} - \hat{\mathbf{X}}\mathbf{S} - \mathbf{E}_s \rangle + \frac{\alpha}{2}\|\hat{\mathbf{X}} - \hat{\mathbf{X}}\mathbf{S} - \mathbf{E}_s\|_F^2 \\ &+ \langle \mathbf{L}_4, q(\mathbf{X}) - \hat{\mathbf{X}} \rangle + \frac{\alpha}{2}\|q(\mathbf{X}) - \hat{\mathbf{X}}\|_F^2 \\ &+ \langle \mathbf{L}_5, \mathbf{X}\mathbf{Q} \rangle + \frac{\alpha}{2}\|\mathbf{X}\mathbf{Q}\|_F^2 \\ &+ \langle \mathbf{L}_6, \mathbf{T} - \mathbf{J} \rangle + \frac{\alpha}{2}\|\mathbf{T} - \mathbf{J}\|_F^2 \\ &+ \langle \mathbf{L}_7, \mathbf{S} - \mathbf{K} \rangle + \frac{\alpha}{2}\|\mathbf{S} - \mathbf{K}\|_F^2 , \end{aligned} \tag{10}$$

and $\boldsymbol{\Theta}_{\text{DUST-TS}} \equiv \{\mathbf{J}, \mathbf{T}, \mathbf{K}, \mathbf{S}, \mathbf{X}, \hat{\mathbf{X}}, \mathbf{E}_s, \mathbf{E}_t\}$ are the spatio-temporal clustering and shape parameters, including three support matrices we have introduced corresponding to $q(\mathbf{X}) \equiv \hat{\mathbf{X}}$, $\mathbf{T} \equiv \mathbf{J}$ and $\mathbf{S} \equiv \mathbf{K}$. Additionally, $\{\mathbf{L}_1, \mathbf{L}_5\} \in \mathbb{R}^{3N \times F}$, $\mathbf{L}_2 \in \mathbb{R}^{2F \times N}$, $\{\mathbf{L}_3, \mathbf{L}_4\} \in \mathbb{R}^{3F \times N}$, $\mathbf{L}_6 \in \mathbb{R}^{F \times F}$ and $\mathbf{L}_7 \in \mathbb{R}^{N \times N}$ are the Lagrange multipliers; and $\alpha > 0$ is a penalty coefficient so as to improve convergence.

The previous minimization problem is carried out efficiently by solving each subproblem separately and in closed form, while keeping fixed the rest of variables. Algorithm 1 explains the details. The expressions for estimating $\mathbf{T}$, $\mathbf{S}$ and $\hat{\mathbf{X}}$ (steps 5, 7 and 10) are obtained by computing the derivatives of Eq. (10) in $\mathbf{T}$, $\mathbf{S}$ and $\hat{\mathbf{X}}$ and equating to zero. For $\mathbf{J}$, $\mathbf{K}$ and $\mathbf{X}$ matrices (steps 4, 6 and 8), we apply a Singular Value Thresholding minimization [18] with a 'shrinkage operator' $S^*_{\frac{\alpha}{\alpha}}(x) = \max(0, x - \frac{*}{\alpha})$ where $* = \{1, \gamma\}$. The optimization of matrices $\mathbf{E}_t$ and $\mathbf{E}_s$ (steps 12 and 13) can be done in closed form by the element-wise shrinkage operator $S^*_{\frac{\alpha}{\alpha}}(x) = \max(0, x - \frac{*}{\alpha})$ where $* = \{\lambda_s, \lambda_t\}$ [44]. After each iteration, the Lagrange multipliers are updated according to standard rules as shown in lines 14-19. The affinity matrices and the rest of model parameters are initialized to null matrices.

Figure 3-left shows the evolution of the cumulative error of the seven constraints in Eq. (10), for one specific dataset (see *Violence* experiment in the Results section). The 3D reconstruction error e_x is plotted in the same figure. Note that after around 60 iterations all constraints are almost perfectly satisfied. Indeed, a few extra iterations guarantee the exact satisfaction. As expected, there still remains a residual 3D reconstruction error, as the constraints are just an approximate model of the true physical behavior of the deformable bodies. In any event, as it will be shown in the results section, the reconstruction error we obtain improves state of the art.

6.3 DUST2-TS: Joint 3D Shape, Clustering, Camera Motion and Missing Tracks

We next present DUST2-TS, which in contrast to DUST-TS, iteratively estimates all model parameters in one single iterative loop, including the full measurement matrix $\hat{\mathbf{P}}$ (recall that in DUST-TS this matrix was estimated as an independent step). To this end, we leverage on the dual union of spatio-temporal subspaces to span also the measurement matrix.

Theoretically, we could define the relations $\hat{\mathbf{P}} = \hat{\mathbf{P}}\mathbf{S} + \mathbf{H}_s$ and $\mathbf{P} = \mathbf{P}\mathbf{T} + \mathbf{H}_t$, where $\mathbf{S}$ and $\mathbf{T}$ are the same spatial and temporal affinity matrices defined in Eqs. (2) and (3), respectively. $\mathbf{H}_s$ and $\mathbf{H}_t$ represent 2D spatial and temporal residual noise. However, since $\hat{\mathbf{P}} = \mathbf{G}\hat{\mathbf{X}} = \mathbf{G}\hat{\mathbf{X}}\mathbf{S} + \mathbf{G}\mathbf{E}_s$ we would have that $\mathbf{H}_s = \mathbf{G}\mathbf{E}_s$, turning the relation $\hat{\mathbf{P}} = \hat{\mathbf{P}}\mathbf{S} + \mathbf{H}_s$ into redundant. A similar conclusion could be reached for the constraint $\mathbf{P} = \mathbf{P}\mathbf{T} + \mathbf{H}_t$. That is, both these constraints on the 2D tracks, do not need to be explicitly formulated in order to ensure that $\hat{\mathbf{P}}$ and $\mathbf{P} = d^{-1}(\hat{\mathbf{P}})$ lie in a dual union of spatio-temporal subspaces.

Based on this observation, we now formulate a new ALM optimization which besides camera motion, shape and spatio-temporal clustering, it also estimates, the missing point tracks. The corresponding Lagrangian be built upon that in Eq. (10) as follows:

$$\underset{\boldsymbol{\Theta}_{\text{DUST2-TS}}}{\arg\min} \quad \{\text{CostDUST2-TS}\} \tag{11}$$

where:

$$\begin{aligned}\text{CostDUST2-TS} =& \text{CostDUST-TS} \\ &+ \| \left(\mathbf{1}_2 \otimes \mathbf{O}^\top\right) \odot d^{-1}\left(\hat{\mathbf{P}} - \bar{\mathbf{P}}\right) \|_F^2 + \|\mathbf{M}\|_* \\ &+ \langle \mathbf{L}_8, d(\mathbf{P}) - \hat{\mathbf{P}} \rangle + \frac{\alpha}{2}\|d(\mathbf{P}) - \hat{\mathbf{P}}\|_F^2 \\ &+ \langle \mathbf{L}_9, \mathbf{P} - \mathbf{M} \rangle + \frac{\alpha}{2}\|\mathbf{P} - \mathbf{M}\|_F^2 \\ &+ \zeta \sum_{f=1}^{F-1} \|\nabla^f \mathbf{R}\|_{\mathcal{F}}^2 , \end{aligned} \tag{12}$$

and $\boldsymbol{\Theta}_{\text{DUST2-TS}} = \boldsymbol{\Theta}_{\text{DUST-TS}} \cup \{\hat{\mathbf{P}}, \mathbf{M}, \mathbf{P}, \mathbf{G}\}$. Two support matrices have been introduced, namely $d(\mathbf{P}) \equiv \hat{\mathbf{P}}$ and $\mathbf{P} \equiv \mathbf{M}$, and $\mathbf{L}_8 \in \mathbb{R}^{2F \times N}$ and $\mathbf{L}_9 \in \mathbb{R}^{2N \times F}$ are the extra Lagrange multipliers. In this context, the function $d(\cdot)$ is equivalent to the function $q(\cdot)$, but re-arranging 2D observations rather than 3D coordinates. ∇^f represents the discrete temporal derivative operator, and $\zeta > 0$ is a weight coefficient.

The cost function of DUST2-TS algorithm in Eq. (12) can be seen as an extended version of the original cost for DUST-TS defined in Eq. (10), with additional update rules for both camera parameters and missing tracks. That is, the full energy defined in Eq. (5) is solved in a unified manner. Since this problem is non-convex, proper initialization is required for fast convergence. In this paper, we uniquely initialize the missing tracks and the camera parameters using Eq. (7) and Eq. (8), respectively.

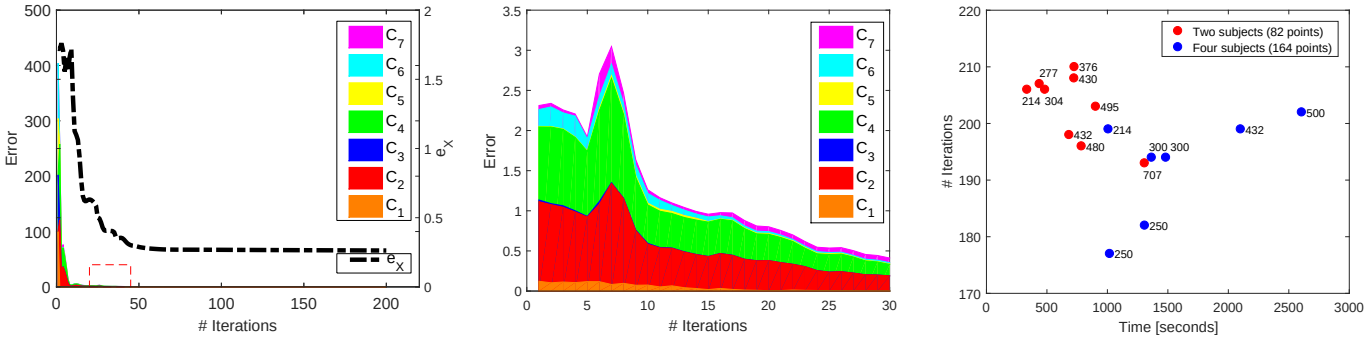

Fig. 3. **Convergence analysis and number of iterations vs. computation time. Left:** Evolution of the error for the seven constraints (denoted as C_c, with $c = \{1, \ldots, 7\}$) in Eq. (10) and the 3D reconstruction error e_X as a function of the number of iterations until convergence (corresponding to the *Violence* sequence described in the results section). Note that two different scales (left and right vertical axes) are used to represent the errors of the constraints and the error e_X **Center:** Zoom of the area within the red dashed rectangle in the left plot. **Right:** Computation time vs. number of iterations until convergence on the Mocap sequences described in the Results section, for two (red dots) and four (blue dots) subjects. Next to each dot are indicated the number of frames of the sequence. In all cases, the number of iterations until convergence always remains within reasonable bounds. The corresponding computation time depends on the number of frames and points.

Then, we iteratively apply the update rules detailed in Algorithm 2 until convergence. Obtaining the camera parameters requires solving a manifold-optimization problem to enforce the $\mathbf{R}^f$ matrices to lie in $SO(3)$. We achieve this using the trust-region solver in the Manopt toolbox [13].

6.4 Spatial and Temporal Clustering

Once the affinity matrices $\mathbf{T}$ and $\mathbf{S}$ are estimated, we run the spectral clustering algorithm proposed in [20] to discover the actual clusters. Figure 1 shows an example of two matrices we obtain, where each entry (i, j) indicates the degree of similarity between the i-th and j-th frame (for the case of $\mathbf{T}$), or between the i-th and j-th data point (for the case of $\mathbf{S}$). The bar right below the affinity matrices represents the clusters discovered after applying [20]. The granularity of the segmentation can be controlled through a threshold on the eigenvalues internally computed by [20].

6.5 Complexity Analysis

The most computationally demanding part of the algorithms 1-2 corresponds to the step 10 in Algorithm #1, which requires computing an inverse matrix of size $3FN \times 3FN$. It is worth pointing out that even though our algorithm requires to compute several SVD operations (see steps 4, 6 and 8; and step 2 for Algorithm #2), their complexities become negligible compared to the previous inverse computation. On balance, our problem can be sorted out in a polynomial time with a computational complexity of at most of $\mathcal{O}(N^3F^3)$ [33]. The computation times (in Matlab) for Motion Capture (Mocap) sequences for two and four people are reported in Fig. 3-right. On average, the median computation time in experiments with sequences between $214 - 707$ and $214 - 500$ image frames for two and four humans was of 724 and 1365 seconds, respectively, on a laptop with an Intel Core i7 processor at 2.4GHz.

input : Possibly incomplete 2D trajectories $\bar{\mathbf{P}}$ and parameters $\{\lambda_t, \lambda_s, \gamma\}$
output: 3D reconstruction $\hat{\mathbf{X}}$ or $\mathbf{X}$, camera rotation $\mathbf{G}$, spatial $\mathbf{S}$ and temporal $\mathbf{T}$ clustering, and full 2D trajectories $\hat{\mathbf{P}}$

/* Initialization: Eqs. (7)–(8) */
/* ALM Optimization of Eq. (12) */
1 **while** *not converged* **do**
  /* Update Model Parameters */
2 $\mathbf{M} = \min \frac{1}{\alpha}\|\mathbf{M}\|_* + \frac{1}{2}\|\mathbf{M} - (\mathbf{P} + \frac{\mathbf{L}_9}{\alpha})\|_F^2$
3 $\mathbf{P} \approx (\mathbf{1}_2 \otimes \mathbf{O}^\top) \odot d^{-1}(\bar{\mathbf{P}}) + (\mathbf{1}_2 \otimes \tilde{\mathbf{O}}^\top) \odot \frac{1}{2}(d^{-1}(\hat{\mathbf{P}} - \frac{\mathbf{L}_8}{\alpha}) + \mathbf{M} - \frac{\mathbf{L}_9}{\alpha})$
4 $\hat{\mathbf{P}} = \frac{1}{2}(d(\mathbf{P}) + \mathbf{G}\hat{\mathbf{X}} + \frac{\mathbf{L}_8 - \mathbf{L}_2}{\alpha})$
  /* Update Rules to solve Eq. (10) */
  /* Update Lagrange Multipliers and Penalty Weights */
  /* Check Convergence */
5 **if** $\|\hat{\mathbf{P}} - \mathbf{G}\hat{\mathbf{X}}\|_\infty < \epsilon_2$ **then**
   $\arg\min \frac{1}{2}\|\hat{\mathbf{P}} - \mathbf{G}\hat{\mathbf{X}}\|_\mathcal{F}^2 + \zeta \sum_{f=1}^{F-1} \|\nabla^f \mathbf{R}\|_\mathcal{F}^2$
6 **else** $\mathbf{G} \equiv \mathbf{G}$
7 **end**
8 Not.: The negative of a binary matrix $\mathbf{O}$ is denoted as $\tilde{\mathbf{O}}$. Hyper-parameters: $\zeta = 6.0$ and $\epsilon_2 = 0.5 \cdot 10^{-1}$.

Algorithm 2: DUST2-TS algorithm for optimizing Eq. (5).

7 Experimental Evaluation

We now present our experimental evaluation for different types of scenarios where several humans perform different everyday activities (see videos in the supplemental material). We provide both qualitative and quantitative results, and compare our approach against state-of-the-art techniques on several Mocap datasets with 3D ground truth.

In the following, we will report the reconstruction error in terms of the normalized mean 3D error e_X used

Method / Data / Metric:	CSF [35] e_X	KSTA [36] e_X	BMM [24] e_X	EM-PND [40] e_X	TUS [62] e_X	GBNR [30] e_X	CNR [41] e_X	DUST [5] e_X	Ours (DUST-TS) clean and complete data e_X	e_S[%]	e_T[%]	sparse/structured/noise e_X	e_X	e_X
Two subjects														
Jump	0.053	0.071	0.078	0.065	0.054	0.070	0.074	0.045	**0.035**	0.0(2)	5.4(3)	0.040	0.058	0.060
Pull	0.123	0.128	0.146	0.113	0.116	0.138	0.183	0.118	**0.093**	0.0(2)	7.7(4)	0.103	0.106	0.115
Soldiers	0.104	0.106	0.080	0.342	0.073	0.076	0.091	0.049	**0.049**	1.2(2)	5.0(2)	0.049	0.052	0.069
Stares Down	0.036	0.022	0.050	0.013	0.032	0.048	0.038	0.016	**0.012**	0.0(2)	0.0(2)	0.014	0.022	0.043
Stumbles	0.094	0.102	0.124	0.099	0.112	0.119	0.119	0.096	**0.086**	0.0(2)	1.3(2)	0.091	0.105	0.103
Squats	0.047	0.041	0.040	0.055	0.016	0.036	0.023	0.015	**0.012**	4.8(2)	0.8(2)	0.015	0.019	0.047
Synchronized	0.141	0.145	0.152	0.145	0.091	0.147	0.112	0.083	**0.062**	0.0(2)	1.2(2)	0.069	0.072	0.086
Violence	0.072	0.073	0.090	0.150	0.081	0.085	0.135	0.060	**0.053**	0.0(2)	1.1(3)	0.056	0.068	0.073
Zombie	0.070	0.067	0.062	0.076	0.056	0.061	0.087	0.043	**0.042**	0.0(2)	9.3(3)	0.043	0.057	0.057
Average error:	0.082	0.084	0.091	0.117	0.070	0.087	0.096	0.058	**0.049**	0.6	3.5	0.053	0.062	0.073
Relative error:	1.66	1.70	1.84	2.37	1.42	1.76	1.94	1.17	**1.00**	-	-	1.08	1.26	1.47
Four subjects														
Blind4	0.047	0.040	0.079	0.079	0.059	0.074	0.137	0.045	**0.038**	0.0(4)	0.3(2)	0.041	0.044	0.048
Chicken4	0.030	0.034	0.027	0.022	0.017	0.021	0.022	0.015	**0.015**	0.0(4)	0.2(3)	0.018	0.021	0.023
Greet4	0.048	0.041	0.078	0.069	0.072	0.077	0.085	0.051	**0.040**	0.0(4)	2.0(3)	0.045	0.047	0.051
Shelters4	0.055	0.053	0.087	0.053	0.037	0.085	0.069	0.034	**0.031**	0.0(3)	3.2(2)	0.032	0.043	0.040
Soda4	0.011	0.011	0.009	0.010	0.009	0.011	0.016	0.007	**0.007**	0.0(4)	1.0(2)	0.007	0.010	0.019
Synchronized4	0.093	0.077	0.056	0.042	0.046	0.049	0.078	0.041	**0.032**	0.0(4)	1.2(2)	0.040	0.042	0.044
Zombie4	0.055	0.067	0.047	0.051	0.043	0.046	0.061	0.033	**0.032**	0.0(4)	8.9(3)	0.033	0.033	0.042
Average error:	0.048	0.046	0.055	0.046	0.040	0.052	0.067	0.032	**0.027**	0.0	2.4	0.031	0.034	0.038
Relative error:	1.72	1.65	1.97	1.65	1.44	1.87	2.40	1.15	**1.00**	-	-	1.10	1.23	1.37

TABLE 2

Evaluation on CMU sequences with two and four subjects, assuming known the camera rotation. The table reports the 3D reconstruction error e_X for the following NRSfM baselines considering full and clean 2D tracks: CSF [35], KSTA [36], SPM [24], EM-PND [40], TUS [62], GBNR [30], CNR [41] and DUST [5]; and ours (DUST-TS). Relative error is computed with respect to DUST-TS reconstruction, using complete and clean data. For our approach, we also show the clustering errors e_S and e_T and the number of spatial and temporal clusters in parentheses. The three right-most columns summarize the reconstruction accuracy under: 40% of random missing data; 15% of structured patterns of missing data; and noisy measurements, respectively.

before in [10], [24], [35]:

$$e_X = \frac{1}{\sigma F N} \sum_{f=1}^{F} \sum_{n=1}^{N} e_n^f, \quad \sigma = \frac{1}{3F} \sum_{f=1}^{F} (\sigma_x^f + \sigma_y^f + \sigma_z^f),$$

where e_n^f is the 3D error for the n-th point at frame f. Triple $(\sigma_x^f, \sigma_y^f, \sigma_z^f)$ are the standard error deviations at f.

For the assessment of the subspace clustering accuracy, we compare our results with a ground truth clustering obtained as follows. First, the 'ground truth' similarity matrices $\mathbf{S}^{GT}$ and $\mathbf{T}^{GT}$ are computed by applying the low-rank representation proposed in [45] over the matrices $\hat{\mathbf{X}}$ and $\mathbf{X}$ with the true 3D point positions. We then perform spectral clustering [20] over $\mathbf{S}^{GT}$ and $\mathbf{T}^{GT}$ to retrieve $\mathcal{S}^{GT}$ and $\mathcal{T}^{GT}$, which are N- and $F-$ dimensional vectors, where each entry is an integer representing the ground truth cluster index. Note that when performing this clustering directly on clean 3D data makes these algorithms very reliable. If we denote by $\mathcal{S}$ and $\mathcal{T}$ the corresponding cluster indexes obtained from the similarity matrices estimated by our approach, we finally define the following clustering errors:

$$e_S = \frac{100}{N} \sum_{i=1}^{N} \mathbb{I}(\mathcal{S}_i \neq \mathcal{S}_i^{GT}), \quad e_T = \frac{100}{F} \sum_{f=1}^{F} \mathbb{I}(\mathcal{T}_f \neq \mathcal{T}_f^{GT}),$$

where $\mathbb{I}(a)$ is the indicator function, i.e., $\mathbb{I}(a) = 1$ if a is true, and 0 otherwise. In practice, for the results we report later, we run [20] for different levels of granularity and keep the result that minimizes e_S and e_T. This, however, does not have any effect on the 3D reconstruction results since the spectral clustering is applied after estimating the affinity matrices.

7.1 Motion Capture Data

We initially evaluate the proposed approaches on the CMU MoCap dataset [1]. Specifically, we consider several scenarios with two or more subjects interacting and performing complex motions. Since 2D projections are not directly available on this dataset, we generate them by synthesizing point tracks acquired by an orthographic camera that follows a circular trajectory around the scene, at an angular speed of $0.66\pi/sec$. On average, the sequences we consider are around 600 frames long, and the number of points per frame is either 82 (when considering two subjects) or 164 (four subjects). For all experiments, we provide two types of validations: the 3D reconstruction accuracy that we compare to other NRSfM methods, and the results of the spatial and temporal subspace clustering, which are compared to a ground truth. For all experiments, we set the coefficients in Eq. (10) to $\lambda_t = \lambda_s = 0.03$, and $\gamma = 14$.

We assess the reconstruction accuracy of our two approaches, DUST-TS and DUST2-TS, and seven other NRSfM baselines: the shape-trajectory methods CSF [35] and KSTA [36]; the block matrix approach BMM [24], the probabilistic-normal-distribution method EM-PND[2] [40], the temporal union of subspaces TUS [62], the grouping-based NRSfM of GBNR [30] and the consensus NRSfM of CNR [41]. We also include our early baseline DUST [5], without assuming smoothness priors. For CSF [35] and KSTA [36], we manually set the rank of the subspace to the value yielding the best results. [39] is not con-

2. EM-PND [40] is not reflection-aware, and requires selecting either the estimated shape or its reflected version. This choice is done based on which of the two configurations mostly resembles the ground truth shape.

Method / Data / Metric:	CSF [35] e_X	KSTA [36] e_X	BMM [24] e_X	EM-PND [40] e_X	TUS [62] e_X	GBNR [30] e_X	CNR [41] e_X	DUST [5] e_X	Ours (DUST-TS) e_X	Ours (DUST-TS) e_S[%]	Ours (DUST-TS) e_T[%]	Ours (DUST2-TS) e_X	Ours (DUST2-TS) e_S[%]	Ours (DUST2-TS) e_T[%]
Two subjects														
Jump	0.067	0.104	1.228	0.311	0.072	1.236	0.172	0.070	**0.059**	0.0(2)	6.2(3)	0.068	0.0(2)	6.2(3)
Pull	0.168	0.129	0.985	0.246	0.103	1.034	0.169	0.099	**0.089**	0.0(2)	6.7(4)	**0.089**	0.0(2)	6.5(4)
Soldiers	0.649	0.754	0.704	0.175	0.095	0.566	0.256	0.072	0.072	0.0(2)	13.0(3)	**0.055**	0.0(2)	11.2(3)
Stares Down	0.046	0.034	0.144	0.033	0.044	0.219	0.047	0.032	**0.023**	0.0(2)	0.4(2)	0.038	0.0(2)	0.6(2)
Stumbles	0.213	0.164	0.189	0.096	0.098	0.238	0.105	0.086	**0.078**	0.0(2)	1.6(2)	0.089	0.0(2)	2.3(2)
Squats	**0.022**	0.095	1.298	0.119	0.442	1.193	0.039	0.028	0.026	4.8(2)	0.8(2)	0.026	4.8(2)	0.8(2)
Synchronized	0.175	0.539	1.188	0.512	0.097	1.178	0.297	0.081	**0.067**	0.0(2)	2.2(2)	0.099	0.0(2)	2.5(2)
Violence	0.147	0.169	0.288	**0.141**	0.297	0.339	0.154	0.267	0.263	0.0(2)	3.2(3)	0.263	0.0(2)	3.2(3)
Zombie	0.208	0.176	0.162	0.184	0.171	0.198	0.120	0.175	0.149	0.0(2)	9.3(3)	**0.075**	0.0(2)	5.0(3)
Average error:	0.188	0.240	0.687	0.202	0.157	0.689	0.151	0.101	0.092	0.6	4.8	**0.089**	0.6	4.2
Relative error:	2.05	2.61	7.48	2.20	1.71	7.51	1.64	1.10	1.00	-	-	**0.97**	-	-
Four subjects														
Blind4	0.145	0.173	0.204	0.094	0.056	0.221	0.198	0.048	**0.045**	0.0(4)	0.3(2)	0.056	0.0(4)	0.3(2)
Chicken4	0.031	0.032	0.054	0.037	2.335	0.102	0.029	0.018	**0.017**	0.0(4)	0.2(3)	0.023	0.0(4)	0.2(3)
Greet4	0.331	0.336	0.163	**0.067**	0.175	0.159	0.121	0.188	0.172	0.0(4)	4.4(3)	0.171	0.0(4)	4.4(3)
Shelters4	0.234	0.309	0.237	0.263	0.199	0.200	0.114	0.205	0.190	0.0(3)	4.8(2)	**0.112**	0.0(3)	4.4(2)
Soda4	0.013	0.016	1.702	0.037	0.692	0.035	0.017	**0.008**	**0.008**	0.0(4)	1.0(2)	0.015	0.0(4)	1.3(2)
Synchronized4	0.164	0.107	0.103	0.059	0.486	0.139	0.153	0.045	**0.042**	0.0(4)	1.2(2)	0.056	0.0(4)	1.6(2)
Zombie4	0.154	0.257	0.136	**0.108**	0.138	0.131	0.113	0.131	0.122	0.0(4)	10.7(3)	0.122	0.0(4)	10.7(3)
Average error:	0.154	0.176	0.372	0.092	0.583	0.141	0.106	0.092	0.085	0.0	3.2	**0.079**	0.0	3.3
Relative error:	1.81	2.06	4.37	1.08	6.85	1.66	1.24	1.08	1.00	-	-	**0.93**	-	-

TABLE 3

Evaluation on CMU sequences with two and four subjects when jointly estimating 3D shape and camera motion. 3D reconstruction error e_X for the following NRSfM baselines: CSF [35], KSTA [36], SPM [24], EM-PND [40], TUS [62], GBNR [30], CNR [41] and DUST [5]; and our DUST-TS and DUST2-TS algorithms. The relative error is computed with respect to the DUST-TS reconstruction. Again, for our approaches, we also report the clustering errors e_S and e_T, indicating the number of estimated spatial and temporal clusters in parentheses.

Method / Data / Metric:	DUST-TS sparse e_X	DUST-TS structured e_X	DUST-TS noise e_X	DUST2-TS sparse e_X	DUST2-TS structured e_X	DUST2-TS noise e_X
Two subjects						
Jump	0.062	0.079	0.075	0.089	0.086	0.095
Pull	0.092	0.109	0.106	0.109	0.117	0.103
Soldiers	0.073	0.084	0.085	0.065	0.071	0.073
Stares Down	0.025	0.030	0.048	0.051	0.055	0.062
Stumbles	0.081	0.099	0.092	0.101	0.116	0.103
Squats	0.028	0.031	0.049	0.030	0.029	0.049
Synchronized	0.069	0.073	0.083	0.121	0.118	0.104
Violence	0.263	0.264	0.266	0.279	0.272	0.268
Zombie	0.149	0.159	0.165	0.146	0.155	0.165
Average error:	0.093	0.103	0.107	0.110	0.113	0.113
Relative error:	1.02	1.13	1.17	1.20	1.24	1.24
Four subjects						
Blind4	0.045	0.049	0.052	0.080	0.078	0.082
Chicken4	0.021	0.024	0.025	0.041	0.059	0.042
Greet4	0.172	0.172	0.174	0.171	0.173	0.174
Shelters4	0.190	0.195	0.192	0.193	0.196	0.146
Soda4	0.008	0.015	0.049	0.020	0.019	0.025
Synchronized4	0.044	0.071	0.048	0.077	0.078	0.070
Zombie4	0.122	0.129	0.125	0.123	0.127	0.135
Average error:	0.086	0.094	0.095	0.100	0.104	0.096
Relative error:	1.02	1.10	1.12	1.18	1.23	1.13

TABLE 4

Evaluation under measurement artifacts. 3D reconstruction error e_X for DUST-TS and DUST2-TS, considering: 70% of sparsely distributed missing data, 20% of structured missing tracks, and noisy observations. The relative error is computed with respect to the DUST-TS solution in Table 3.

sidered as its source code is not publicly available, and for TUS [62] we use our own implementation for the same motive. Our approaches do not require tuning the subspace rank parameter for any domain, neither assigning which data points belong to which subspace.

It is worth noting that all methods, except DUST2-TS, decouple the problems of camera rotation estimation and 3D shape reconstruction. Therefore, in order to focus our analysis solely on the 3D shape reconstruction capacity, we will first provide the same ground truth matrix $\mathbf{G}$ of camera rotations to all methods and then report the 3D reconstruction errors. After that, we will report the results when the camera rotations are estimated. Additionally, we also report experimental results against artifacts in measurements, in three situations: 1) randomly removing 40/70% of the observed points, 2) removing patterns of 15/20% of structured missing entries where consecutive frames include patterns with 50% of missing entries –intended to simulate self-occlusions or structured lack of visibility–, and 3) corrupting the measurements by adding Gaussian noise with standard deviation $\sigma_{noise} = 0.02 \max_{i,j,k} \{|d_{ijk}|\}$, where d_{ijk} represents the maximum distance of an image point to the centroid of all the points.

7.1.1 Sequences with Two Subjects

We select nine sequences of the CMU dataset with two subjects performing different activities and motion patterns, in order to show the ability of our approach on a wide variety of configurations. Namely, we consider *23_16 (Synchronized)*: subjects alternating synchronized jumping jacks; *19_05 (Pull)*: a subject pulls the other by the elbow; *22_20 (Violence)*: a subject picks up high stool and threatens to strike the other; *20_08 (Zombie)*: subjects follow a zombie march; *20_06 (Soldiers)*: subjects follow a soldiers march; *23_19 (Stares Down)*: a subject stares down the other and leans with hands on high stool; *22_12 (Stumbles)*: a subject stumbles into the other; *23_15 (Jump)*: subjects alternating jumping jacks; and *23_14 (Squats)*: subjects alternating squats.

Table 2 summarizes the reconstruction errors for all methods and the subspace clustering accuracy of ours, provided the ground truth camera rotation. Note that DUST-TS consistently outperforms state-of-the-art in

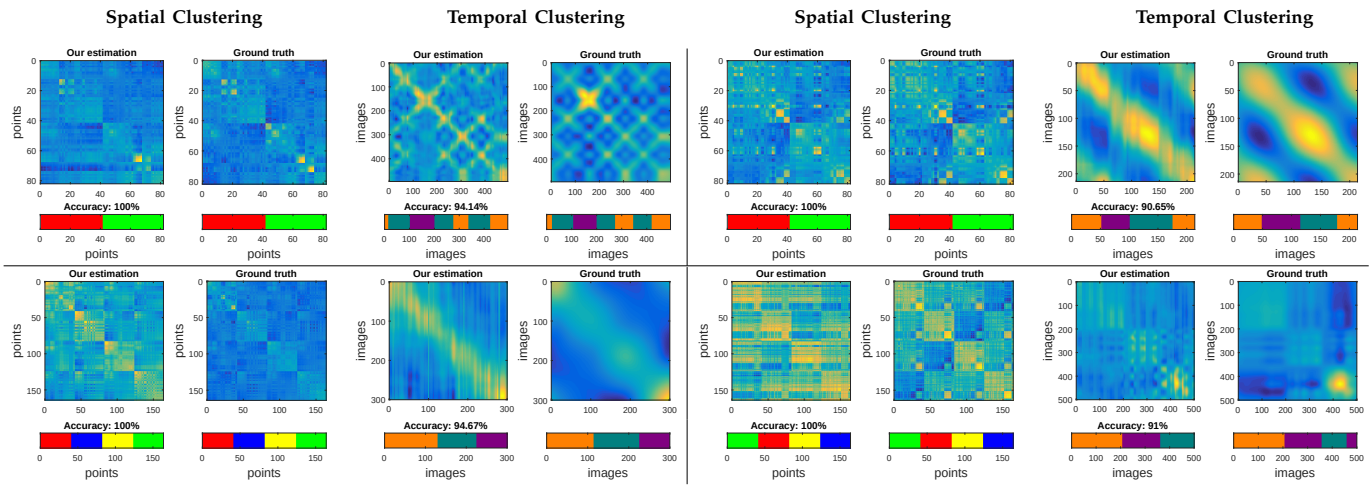

Fig. 4. **Spatial and temporal clustering on CMU sequences.** We compare the spatial $\mathbf{S}$ and temporal $\mathbf{T}$ clustering matrices obtained with our approach with the ground truth ones. Below each matrix we plot a bar with the results of the spectral clustering. **Top:** *Jump* and *Zombie* sequences with two subjects and three temporal primitives. **Bottom:** *Blind4* and *Chicken4* sequences with four subjects and three temporal primitives.

terms of 3D reconstruction, reducing the 3D error of other methods by large margins between the 17% and 237%. Furthermore, DUST-TS also performs shape and temporal clustering. The quality of these segmentations is also very good. In particular, the number of spatial clusters we retrieve in all experiments is two, and all points are correctly assigned to one of the subjects. The number of temporal clusters we estimate is between 2 and 4, and the exact temporal split (i.e., the moment when one sub-action switches to another one) is very close (if not equal) to that of the ground truth. Indeed, most temporal clusters match real motion primitives (see the example in Fig. 1). Note also that when input 2D measurements are corrupted by artifacts (random and structured occlusions, and noise), our method obtains similar performance to other approaches that use complete and clean data. The clustering results in these situations are roughly the same.

Table 3-top summarizes the results when both 3D shape and camera rotation are estimated. Again, our approaches DUST-TS (and now also DUST2-TS) consistently outperform state-of-the-art in terms of reconstruction accuracy, largely reducing the 3D error of other methods (from 10% to 751%). Focusing on our two approaches, we observe that DUST2-TS slightly outperforms DUST-TS for the artifact-free case. Finally, we also evaluate the robustness of our two approaches under artifacts in the measurement matrix $\tilde{\mathbf{P}}$. These results are reported in Table 4-top. In general, our approaches provide accurate solutions with a similar performance to state-of-the-art solutions when they use perfect 2D data (see Table 3-top). The completion algorithms do a pretty good job hypothesizing the missing observations, especially for the random scattered occlusions, and the final reconstruction is nearly unaffected by these artifacts. The accuracy of the spatio-temporal clustering is almost identical to that for the artifacts-free case. Regarding our approaches DUST-TS and DUST2-TS, we observe DUST2-TS outperforms DUST-TS when the level of artifacts is small. This is because DUST2-TS has to estimate more parameters than DUST-TS and becomes more sensitive to perturbations of the input 2D measurements. Recall that in DUST-TS, both $\hat{\mathbf{P}}$ and $\mathbf{G}$ are kept fixed during the optimization.

Figure 4 shows a qualitative comparison of the similarity matrices we estimate and those of the ground truth, which are directly computed from clean 3D data. Despite the matrices provided by our approach are noisier, we can clearly identify the same patterns as in the ground truth. The spectral algorithm we use [20], can easily handle this noise and yields the correct number of clusters in almost all experiments. In Fig. 5, we show several frames of the 3D reconstruction results for the *Violence* and *Pull* sequences.

7.1.2 Sequences with Four Subjects

We also considered a more complex case with four subjects. Since the CMU dataset only includes sequences with one or two subjects, we combined several of them to generate seven new sequences with four subjects, namely: *Synchronized4*: subjects alternating synchronized jumping jacks; *Zombie4*: subjects follow a zombie march; *Chicken4*: subjects perform a non-synchronized chicken dance; *Greet4*: subjects walking and shaking hands; *Blind4*: four subjects playing "blind man's bluff"; *Soda4*: two subjects pass a soda cup to the other two and all of them drink; and *Shelters4*: two subjects individually shelter the other two. Again an orthographic camera moving slowly around the scene is considered. In these examples, the degree of superposition in the image plane is so extreme, that the task of performing the spatial segmentation becomes very difficult (see the 2D projections in Fig. 5-bottom). Indeed, in some of the sequences two of the subjects are so intimately connected, that they can be interpreted as one single object.

Fig. 5. **3D reconstruction and spatio-temporal segmentation on multi-subject sequences**. Results for the *Violence* (first row), *Pull* (second row), *Greet4* (third row) and *Blind4* (fourth row) sequences. For each scene we display several image frames, seen from two perpendicular viewpoints (z-x and y-x). Colored dots represent the 3D position and spatial cluster index estimated by our approach (DUST-TS). Note that the two subjects (first and second row) and the four subjects (third and fourth row) are clearly identified. No single point is assigned to a wrong subject. Empty circles indicate the ground truth 3D position. The color of the contour of these circles encodes to which temporal prior does the frame belong. Observe that in every video we identify several temporal groups. For instance, for the *Violence* sequence, the priors have a clear physical meaning: 'two subjects sitting down', 'one subject standing up an threatening the second one', 'one subject attacks the other that falls down'. The physical interpretation of the temporal priors for the four-subject cases is not that straight-forward, although it seems to encode the types of subject interactions.

The experimental results are summarized in Tables 2, 3, 4-bottom. These results are based on the same type of analysis we did for two subjects. Again, our approaches improve other NRSfM approaches in terms of 3D reconstruction by a large margin (from 15% to 240% when the camera rotation is known, and from 8% to 685% when it is unknown). It is worth pointing out the good performance of KSTA [36] for the sequences in which the subjects perform larger trajectories (*Blind4* and *Greet4*). We also observe our methods demonstrate a great resilience against artifacts in the measurements.

Regarding the segmentation accuracy, note that for the *Shelters4* we obtain a better segmentation accuracy by choosing a spatial granularity of 3 instead of 4 (in this case, we could obtain a 89.63% of accuracy). This is because for this specific sequence, two of the objects are always together. We show some similarity matrices and reconstruction examples in Figs. 4 and 5, respectively.

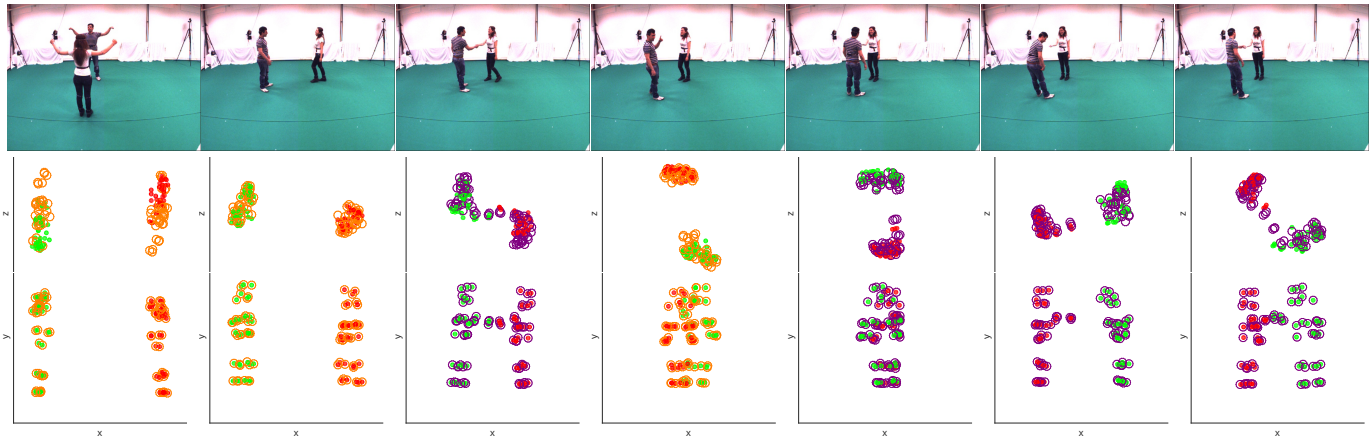

Fig. 6. **Meet sequence. Top:** Sample frames of two interacting humans. **Bottom:** 3D reconstruction and spatio-temporal segmentation. See caption in Fig. 5 for an interpretation of the color coding.

7.2 Video Data

Finally, we evaluate our approach on the *p2_meet_1* sequence from the UMPM [58] dataset, where two subjects are moving and interacting. For evaluation and comparison purposes, we use the 3D ground truth of 74 points, obtaining an e_X of 0.208 and 0.203 for our DUST-TS and DUST2-TS algorithms, respectively. Both these results consistently outperform the competing methods CSF [35], KSTA [36], BMM [24], EM-PND [40], TUS [62] (we use our camera estimation for this method), GBNR [30], CNR [41] and DUST [5], which produce errors of 0.741, 0.466, 0.386, 0.275, 0.223, 1.156, 0.285, and 0.224, respectively. In addition, our formulations provide the spatial segmentation of the two humans, and a temporal clustering into actions. Figure 6 shows a few sample frames of the video, and the spatio-temporal 3D reconstruction we obtain.

8 Conclusion

In this paper we have proposed a novel solution to the NRSfM paradigm that allows exploring a problem which had not been tackled before: given a possibly incomplete monocular sequence of 2D tracks, estimating 3D time-varying shape and camera motion while also providing temporal clustering of the data into deformation-primitives, and spatial segmentation into multiple objects. For this purpose, we have presented two strategies based on a low-rank constraint to represent the time-varying shape as a dual combination of spatial and temporal subspaces. In both cases, we solve the problem by means of augmented Lagrange machinery. We have thoroughly evaluated the approach on challenging sequences involving up to four interacting people performing complex motion patterns. We show that besides providing correct spatio-temporal segmentation, our approach does also reconstruct the 3D human poses more accurately than current state-of-the-art NRSfM methods. In the future, we aim at using this research as a first step to perform complete reconstruction and recognition of human activities.

Acknowledgments

This work is supported in part by a Google Faculty Research Award, by the Spanish Ministry of Science and Innovation under projects HuMoUR TIN2017-90086-R, and María de Maeztu Seal of Excellence MDM-2016-0656. We thank the anonymous reviewers for their insightful comments and suggestions which helped to improve the work.

References

[1] http://mocap.cs.cmu.edu/.
[2] A. Agudo, J. M. M. Montiel, L. Agapito, and B. Calvo. Modal space: A physics-based model for sequential estimation of time-varying shape from monocular video. *JMIV*, 57(1):75–98, 2017.
[3] A. Agudo, J. M. M. Montiel, B. Calvo, and F. Moreno-Noguer. Mode-shape interpretation: Re-thinking modal space for recovering deformable shapes. In *WACV*, 2016.
[4] A. Agudo and F. Moreno-Noguer. Combining local-physical and global-statistical models for sequential deformable shape from motion. *IJCV*, 122(2):371–387, 2017.
[5] A. Agudo and F. Moreno-Noguer. DUST: Dual union of spatio-temporal subspaces for monocular multiple object 3D reconstruction. In *CVPR*, 2017.
[6] A. Agudo and F. Moreno-Noguer. Global model with local interpretation for dynamic shape reconstruction. In *WACV*, 2017.
[7] A. Agudo and F. Moreno-Noguer. Force-based representation for non-rigid shape and elastic model estimation. *TPAMI, to appear*, 2018.
[8] A. Agudo, F. Moreno-Noguer, B. Calvo, and J. M. M. Montiel. Sequential non-rigid structure from motion using physical priors. *TPAMI*, 38(5):979–994, 2016.
[9] I. Akhter, Y. Sheikh, and S. Khan. In defense of orthonormality constraints for nonrigid structure from motion. In *CVPR*, 2009.
[10] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Non-rigid structure from motion in trajectory space. In *NIPS*, 2008.
[11] F. Bach, J. Mairal, and J. Ponce. Convex sparse matrix factorizations. *Technical report HAL-00345747*, 2008.
[12] A. Bartoli, V. Gay-Bellile, U. Castellani, J. Peyras, S. Olsen, and P. Sayd. Coarse-to-fine low-rank structure-from-motion. In *CVPR*, 2008.
[13] N. Boumal, B. Mishra, P. A. Absil, and R. Sepulchre. Manopt, a matlab toolbox for optimization on manifolds. *JMLR*, 15(4):1455–1459, 2014.
[14] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *FTML*, 3(1):1–122, 2011.
[15] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3D shape from image streams. In *CVPR*, 2000.
[16] S. Burer and R. Monteiro. Local minima and convergence in low-rank semidefinite programming. *Math. Prog.*, 103(3):427–444, 2005.

[17] R. Cabral, F. De La Torre, J. P. Costeira, and A. Bernardino. Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition. In *ICCV*, 2013.
[18] J. F. Cai, E. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM JO*, 20(4):1956–1982, 2010.
[19] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *FCM*, 9(6):717, 2008.
[20] W. Y. Chen, Y. Song, H. Bai, C.J. Lin, and E. Chang. Parallel spectral clustering in distributed systems. *TPAMI*, 33(3):568–586, 2010.
[21] Y. Chen, H. Xu, C. Caramanis, and S. Sanghavi. Robust matrix completion with corrupted columns. In *ICML*, 2011.
[22] A. Chhatkuli, D. Pizarro, T. Collins, and A. Bartoli. Inextensible non-rigid shape-from-motion by second order cone programming. In *CVPR*, 2016.
[23] J. Costeira and T. Kanade. A multi-body factorization method for motion analysis. In *ICCV*, 1995.
[24] Y. Dai, H. Li, and M. He. A simple prior-free method for non-rigid structure from motion factorization. In *CVPR*, 2012.
[25] A. Del Bue, X. Llado, and L. Agapito. Non-rigid metric shape and motion recovery from uncalibrated images using priors. In *CVPR*, 2006.
[26] A. Del Bue, J. Xavier, L. Agapito, and M. Paladini. Bilinear modeling via augmented lagrange multipliers (BALM). *TPAMI*, 34(8):1496–1508, 2012.
[27] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *CVPR*, 2009.
[28] E. Elhamifar and R. Vidal. Sparse subspace clustering: algorithm, theory, and applications. *TPAMI*, 35(11):2765–2781, 2013.
[29] J. Fayad, L. Agapito, and A. Del Bue. Piecewise quadratic reconstruction of non-rigid surfaces from monocular sequences. In *ECCV*, 2010.
[30] K. Fragkiadaki, M. Salas, P. Arbeláez, and J. Malik. Grouping-based low-rank trajectory completion and 3D reconstruction. In *NIPS*, 2014.
[31] Y. Gao and A. L. Yuille. Symmetric non-rigid structure from motion for category-specific object structure estimation. In *ECCV*, 2016.
[32] R. Garg, A. Roussos, and L. Agapito. Dense variational reconstruction of non-rigid surfaces from monocular video. In *CVPR*, 2013.
[33] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Univ Pr, 1996.
[34] V. Golyanik and D. Stricker. Dense batch non-rigid structure from motion in a second. In *WACV*, 2017.
[35] P. F. U. Gotardo and A. M. Martinez. Computing smooth time-trajectories for camera and deformable shape in structure from motion with occlusion. *TPAMI*, 33(10):2051–2065, 2011.
[36] P. F. U. Gotardo and A. M. Martinez. Kernel non-rigid structure from motion. In *ICCV*, 2011.
[37] M. Irani. Multi-frame optical flow estimation using subspace constraints. In *ICCV*, 1999.
[38] C. Kong and S. Lucey. Prior-less compressible structure from motion. In *CVPR*, 2016.
[39] S. Kumar, Y. Dai, and H. Li. Spatio-temporal union of subspaces for multi-body non-rigid structure-from-motion. *PR*, 77(11):428–443, 2017.
[40] M. Lee, J. Cho, C. H. Choi, and S. Oh. Procrustean normal distribution for non-rigid structure from motion. In *CVPR*, 2013.
[41] M. Lee, J. Cho, and S. Oh. Consensus of non-rigid reconstructions. In *CVPR*, 2016.
[42] M. Lee, C. H. Choi, and S. Oh. A procrustean markov process for non-rigid structure recovery. In *CVPR*, 2014.
[43] Z. Li, J. Guo, L.F. Cheong, and Z. Zhou. Perspective motion segmentation via collaborative clustering. In *ICCV*, 2013.
[44] Z. Lin, M. Chen, L. Wu, and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *UIUC Technical Report UILU-ENG-09-2215*, 2009.
[45] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *TPAMI*, 35(1):171–184, 2013.
[46] G. Liu, Z. Lin, and Y. Yu. Robust subspace segmentation by low-rank representation. In *ICML*, 2010.
[47] K. Ozden, K. Schindler, and L. van Gool. Multibody structure-from-motion in practice. *TPAMI*, 32(6):1134–1141, 2010.
[48] M. Paladini, A. Del Bue, M. Stosic, M. Dodig, J. Xavier, and L. Agapito. Factorization for non-rigid and articulated structure using metric projections. In *CVPR*, 2009.
[49] S. Parashar, D. Pizarro, and A. Bartoli. Isometric non-rigid shape-from-motion in linear time. In *CVPR*, 2016.
[50] H. S. Park, T. Shiratori, I. Matthews, and Y. Sheikh. 3D reconstruction of a moving point from a series of 2D projections. In *ECCV*, 2010.
[51] S. Rao, R. Tron, R. Vidal, and Y. Ma. Motion segmentation in the presence of outlying, incomplete or corrupted trajectories. *TPAMI*, 32(10):1832–1845, 2010.
[52] B. Recht, M. Fazel, and P. A Parrilo. Guaranteed minimum-rank solu- tions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.
[53] C. Russell, R. Yu, and L. Agapito. Video pop-up: Monocular 3D reconstruction of dynamic scenes. In *ECCV*, 2014.
[54] T. Simon, J. Valmadre, I. Matthews, and Y. Sheikh. Separable spatiotemporal priors for convex reconstruction of time-varying 3D point clouds. In *ECCV*, 2014.
[55] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization approach. *IJCV*, 9(2):137–154, 1992.
[56] L. Torresani, A. Hertzmann, and C. Bregler. Nonrigid structure-from-motion: estimating shape and motion with hierarchical priors. *TPAMI*, 30(5):878–892, 2008.
[57] J. Valmadre and S. Lucey. General trajectory prior for non-rigid reconstruction. In *CVPR*, 2012.
[58] N. P. van der Aa, X. Luo, G. J. Giezeman, R. T. Tan, and R. C. Veltkamp. UMPM benchmark: A multi-person dataset with synchronized video and motion capture data for evaluation of articulated human motion and interaction. In *ICCVW*, 2011.
[59] S. Vicente and L. Agapito. Soft inextensibility constraints for template-free non-rigid reconstruction. In *ECCV*, 2012.
[60] J. Xiao, J.X. Chai, and T. Kanade. A closed-form solution to non-rigid shape and motion recovery. In *ECCV*, 2004.
[61] L. Zappella, A. Del Bue, X. Llado, and J. Salvi. Joint estimation of segmentation and structure from motion. *CVIU*, 117(2):113–129, 2013.
[62] Y. Zhu, D. Huang, F. De La Torre, and S. Lucey. Complex non-rigid motion 3D reconstruction by union of subspaces. In *CVPR*, 2014.
[63] Y. Zhu and S. Lucey. Convolutional sparse coding for trajectory reconstruction. *TPAMI*, 37(3):529–540, 2015.

Antonio Agudo received the M.Sc. degree in industrial engineering and electronics in 2010, M.Sc. degree in computer science in 2011, and the Ph.D. degree in computer vision and robotics in 2015, from University of Zaragoza. He was a visiting student at vision group of Queen Mary University of London in 2013 and with the vision and imaging science group of University College London in 2014. He was also a visiting fellow at Harvard University in 2015. After two years as a postdoctoral fellow at Institut de Robòtica i Informàtica Industrial (CSIC-UPC) in Barcelona, he joined as an associate researcher of the Spanish Scientific Research Council in 2017. His research interests include non-rigid structure from motion, machine learning, and deformation analysis to medical and robotics applications.

Francesc Moreno-Noguer received the MSc degrees in industrial engineering and electronics from the Technical University of Catalonia (UPC) and the Universitat de Barcelona in 2001 and 2002, respectively, and the PhD degree from UPC in 2005. From 2006 to 2008, he was a postdoctoral fellow at the computer vision departments of Columbia University and the École Polytechnique Fédérale de Lausanne. In 2009, he joined the Institut de Robòtica i Informàtica Industrial in Barcelona as an associate researcher of the Spanish Scientific Research Council. His research interests include retrieving rigid and nonrigid shape, motion, and camera pose from single images and video sequences. He received UPC's Doctoral Dissertation Extraordinary Award for his work.