Unsupervised 3D Reconstruction and Grouping of Rigid and Non-Rigid Categories

Antonio Agudo

Abstract—In this paper we present an approach to jointly recover camera pose, 3D shape, and object and deformation type grouping, from incomplete 2D annotations in a multi-instance collection of RGB images. Our approach is able to handle indistinctly both rigid and non-rigid categories. This advances existing work, which only addresses the problem for one single object or, they assume the groups to be known a priori when multiple instances are handled. In order to address this broader version of the problem, we encode object deformation by means of multiple unions of subspaces, that is able to span from small rigid motion to complex deformations. The model parameters are learned via Augmented Lagrange Multipliers, in a completely unsupervised manner that does not require any training data at all. Extensive experimental evaluation is provided in a wide variety of synthetic and real scenarios, including rigid and non-rigid categories with small and large deformations. We obtain state-of-the-art solutions in terms of 3D reconstruction accuracy, while also providing grouping results that allow splitting the input images into object instances and their associated type of deformation.

Index Terms—Category Reconstruction, Multiple Unions of Subspaces, Class Clustering, Augmented Lagrange Multipliers.

1 Introduction

Simultaneously recovering camera motion and 3D object shape from a collection of RGB images either acquired from different viewpoints or by a single moving camera is one of the most active research areas in computer vision. Early works addressed this problem under the assumption of a rigid structure [1], [41], [46]. Later, many efforts were focused on the non-rigid case, to retrieve dynamic 3D shape and camera motion from only 2D measurements in a monocular video [5], [33], [39], [52], [55], [58]. This problem is known to be inherently ambiguous and demanded introducing more sophisticated priors. Probably, the most standard formulations include the use of different modalities of low-rank subspaces to constrain the solution space [5], [9], [12], [39], [50]. Moreover, these algorithms exploit the fact that input images smoothly change viewpoints, introducing temporal smoothness priors on both the shape deformations and the camera poses in order to produce more accurate solutions [3].

However, all these previous methods solve the problem for one single object instance. There exist works addressing scenarios with multiple objects within a category. For example, if the observed category is rigid (e.g., buses or chairs) and all objects in it have the same geometry, the problem can be addressed as a rigid Structure from Motion (SfM) one [44], [54]. When object instances within the same category have distinct geometry, even if they are rigid (e.g., different model buses), the global problem of recovering their shape can be formulated in a non-rigid manner [28]. This can be extended to inherently non-rigid classes (e.g., faces or animal skeletons), in which case, both inter- and intra-object deformations shall be considered [2]. However, all these works are only focused on the 3D reconstruction problem, and assume the object grouping to be known a priori.

In this work we move a step forward and tackle the problem in which the object groups are not known a priori. That is, given an input collection of RGB images of a specific category, we aim at simultaneously grouping them into different object instances and their type of deformation or action, together with retrieving their 3D shape regardless of whether the objects are rigid or non-rigid. As shown in Fig. 1 the outcome of our approach is an object grouping of each image, which is likely to correspond to each of the instances (three object instances are showed for the example in the figure), a deformation-type clustering corresponding to pose primitives (again, three deformation types or actions are displayed in the previous figure), a 3D reconstruction of each individual object and the corresponding camera motion. As we have commented above, we formulate our approach to handle both types of categories, i.e., rigid and non-rigid ones. For example, as shown in Fig. 2-left, given a number of images of chairs (five models seen from different viewpoints) our approach groups them into each of the models and reconstructs their 3D shape. Note that some observations of the chair instances are very similar and difficult to distinguish from only 2D annotations. Simultaneously reasoning about the dual grouping and 3D reconstruction helps improving both tasks. Regarding non-rigid categories, as shown in Fig. 2-right, given a collection of face images of five humans under different viewpoints and facial expressions, our algorithm jointly splits the images into each of the individuals and their deformation types, together with their 3D shape.

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

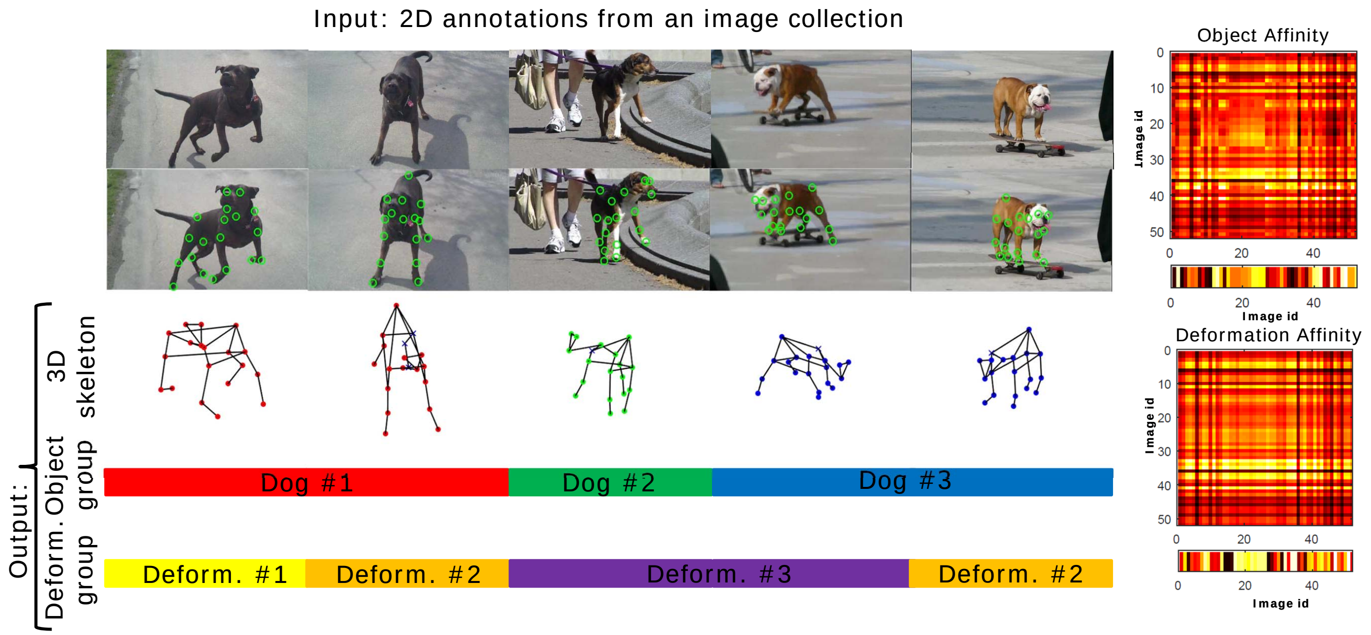

Fig. 1. **Joint 3D reconstruction, camera motion, and dual object and deformation-type grouping from partial 2D annotations. Top-left:** In this part, we represent our input data. Example of real pictures from a dog collection, and some 2D partial annotations in green circles are displayed. We assume these 2D annotations are provided, but the number of object instances and type of deformations are unknown. **Bottom-left:** In this part, we show the output our algorithms can estimate: 3D shape reconstruction together with the object and deformation grouping results. In this example, object segmentation is to split input data into dog instances (three in our case), and deformation grouping to identify pose primitives which have a clear semantic meaning. To make a fair understanding, we display three deformation primitives that we represent by means of action names: 'jump' (yellow), 'stand' (orange), and 'walk' (magenta). Camera motion is not represented in this figure, but it is also an outcome of our algorithm. **Right:** Estimated object and deformation affinity matrices. Each entry in these matrices expresses the object/deformation pairwise affinity between images within the collection. Groups are directly discovered by applying spectral clustering on these matrices.

In order to simultaneously tackle grouping and reconstruction from a collection of unordered images, we propose a novel optimization framework that builds upon recent Non-Rigid Structure from Motion approaches (NRSfM) [6], [64]. More specifically, we model the 3D shape by multiple unions of unknown subspaces, accounting for rigid plus small and large non-rigid deformations. These subspaces, in conjunction with additional matrices encoding the affinities among the samples and among their deformations, are retrieved from incomplete 2D annotations using an efficient Augmented Lagrange Multiplier (ALM) scheme. A subsequent spectral clustering on the affinity matrices yields the results of the partition (an example is shown in Fig. 1). The whole algorithm works in a fully unsupervised manner, without requiring to know a priori the number of object groups nor any other information about the type of deformation (if any) undergone by the objects. We are not aware of any other approach solving the four problems jointly solely from partial 2D point annotations in an image collection. We thoroughly evaluate our algorithm on both synthetic and real images for rigid and non-rigid categories, improving state-of-the-art NRSfM solutions (which do not provide any kind of grouping as we do) by a considerable margin.

An early version of this work was presented in [8], in which we proposed our method to be suitable for simultaneously estimating 3D shape, motion, and object and deformation type grouping, all of them, directly from incomplete 2D annotations in an image collection. In this paper, we extend our contribution incorporating more technical details of our approach and proposing a new algorithm to estimate in the same loop all model parameters we consider. Moreover, we also extend the battery of results to emphasize the advantages of our approach in comparison with state of the art, including more evaluations on both synthetic and real data, and showing the generality and accuracy of our approach even without assuming any training data at all.

2 Related Work

Inferring the 3D shape while retrieving camera location from only 2D point tracks in a collection of RGB images, is a mature problem when the observed object is rigid. In this case, the rigidity constraint is enough to make the problem well-posed, yielding impressively accurate solutions [1], [41], [54]. In contrast, handling non-rigid scenarios becomes an ill-posed problem that requires to exploit the denominated art of priors to constrain the solution space. The most standard prior used in NRSfM consists in constraining the deforming shape to lie in a low-rank subspace. In order to learn such a low-rank model, early approaches rely on factorization [13], [31], [48], [57], or optimization-based strategies [12], [39], [55], [58]. More recently, the low-rank constraint has been enforced by means of PCA-like formulations in which

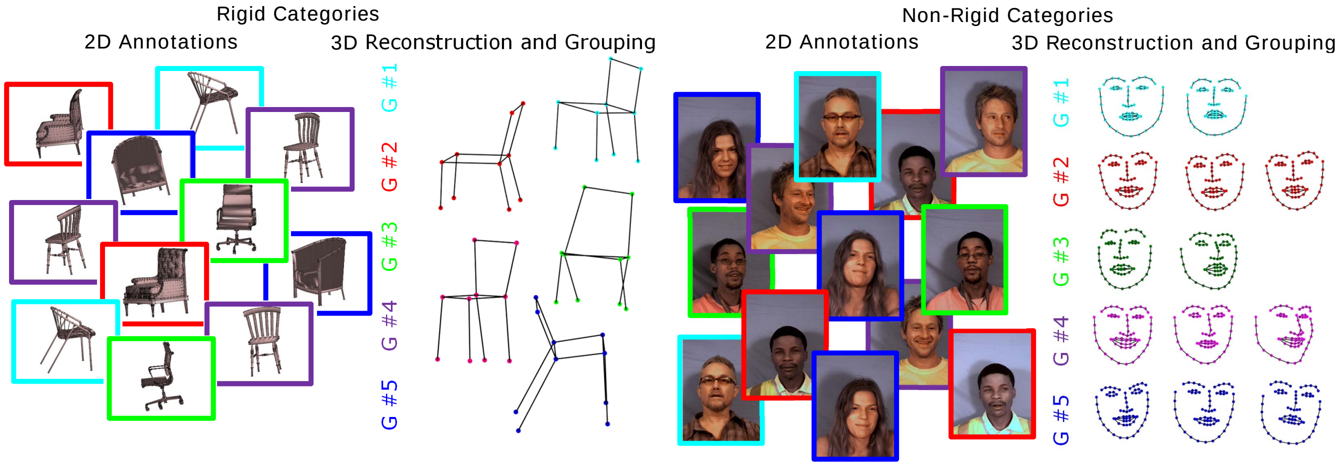

Fig. 2. **3D reconstruction and object grouping from incomplete 2D annotations of rigid and non-rigid categories.** In both cases, input data consists of a collection of RGB images with partial 2D semantic point annotations. The number of objects within the category is unknown. Our goal is to simultaneously retrieve the 3D object reconstruction in every image, the camera pose, and the instance group (a different color per each object instance is used). **Left:** A rigid chair category, in which each instance has a single 3D configuration. **Right:** A non-rigid face category, where every instance may potentially have as many 3D configurations as its number of images. This graph only shows instance grouping, but as we shall see in the results, our approach also permits segmenting every non-rigid instance into several types of deformation (or expressions in the case of the faces).

the rank of the shape matrix is also optimized. These type of methods either assume the data lies in a single low dimensional shape space [22], [27], [29], or in a union of temporal [64] or spatio-temporal subspaces [6], [38]. Low-rank models were also extended to the temporal domain, by exploiting pre-defined trajectory basis [9], [50], the combination of shape-trajectory vectors [32], [33], and the force space that induces the deformations of an object [5].

In addition to low-rank models, there exist also a series of works that enforce other types of constraints. Maybe, the most commonly used prior consists in imposing inextensibility between every pair of neighboring points [20], [49]. While these approaches can produce accurate solutions, the inextensibility prior limits their applicability to only isometric deformations. More general deformations (e.g., articulated motion, discontinuous deformations or elastic warps) can be retrieved through physics-based models [3], [7]. As most of the previous approaches process image sequences, additional temporal smoothness priors on motion and shape deformation have allowed to obtain more robust solutions for rigid [46] and non-rigid scenarios [12], [29], [32], [40].

In any event, while achieving remarkable results, all previous approaches aim at modeling one single object in a category, typically observed from smoothly changing viewpoints. This means they are not directly applicable to a multi-object scenario we contemplate in this paper. However, there have been some attempts along this line. Recent solutions to reconstruct rigid categories from single images [34], [35], resort to large amounts of training data to constrain the solution space. Our approach, instead, aims at learning the solution space on the fly from a collection of images, without requiring any training data at all. There exist very recent works implementing this idea on rigid object categories, either exploiting the concept of symmetry [28], or imposing a sparse shape-space model [37]. In [2], this was extended to non-rigid categories through a dual low-rank shape model which allowed handling small deformations. Nevertheless, these works are still limited by the fact that they assume the grouping of the image collection into objects to be known a priori.

In parallel, some works have relied on neural networks to learn a category deformation model [17], [36], [47] and infer the 3D reconstruction from 2D annotations. In all cases, they propose to exploit several losses to solve the problem in an unsupervised manner as we do in this paper, but require a large amount of training data to learn the deformation model and demand an specific hardware to complete the training step. Unfortunately, this cannot be assumed for generic scenarios, where obtaining training data could be a hard task. In contrast, our formulation can solve the problem in just few seconds in a commodity computer, without requiring sophisticated hardware. Moreover, none of them simultaneously solve for reconstruction and object/deformation grouping as we propose in this paper.

While both SfM and NRSfM approaches exploit feature point correspondences to collect a 2D measurement matrix, in category reconstruction from an image collection this must be extended to semantic point correspondences. Defining the same semantic points in all objects of a category is a difficult task, being their exact position very subjective in some cases. Some recent works [56], [62], [63] to address this problem have been proposed, which though, are beyond the scope of this paper.

We overcome most of the limitations of previous methods with an approach that simultaneously recovers camera pose, 3D shape, object and deformation grouping, and the incomplete 2D annotations, for both rigid and non-rigid categories of object shapes. To this end, we

Feature Method	Automatic rank	Occlusion handing	Object / Type of deformation grouping	Rigid/Non-Rigid categories
[5], [12], [32], [33], [55]	–	✓	–/–	–/–
[22], [29]	✓	–	–/–	–/–
[27], [39], [40]	✓	✓	–/–	–/–
[64]	✓	–	–/✓	–/–
[2]	–	✓	–/–	–/✓
[28], [37]	✓	✓	–/–	✓/–
Ours	✓	✓	✓/✓	✓/✓

TABLE 1

Qualitative comparison of our approach with other competing methods to simultaneously solve reconstruction and dual segmentation of object/deformation categories. Our approach is the only one that jointly retrieves 3D reconstruction of both rigid and non-rigid categories, recovers camera pose, and estimates grouping per object instance and type of deformation. Moreover, it can also handle incomplete 2D observations, and does not need to adjust the rank of the basis, all of them, without assuming any training data at all. Note also that [17], [36], [47], some deep-learning approaches, can handle both rigid and non-rigid categories but require large amounts of training data to learn the deformation model. Additionally, these approaches do not estimate object and deformation type grouping as we do.

encode object deformation by means of multiple unions of subspaces, without assuming any prior knowledge about the dimensionality of the subspaces nor which data points belong to which subspace. As a result, we obtain a unified and unsupervised framework which does not need 3D training data. In table 1, we provide a qualitative comparison of the main characteristics offered by our approach and the most relevant competing techniques. As it can be seen, we are not aware of any other work to jointly offer all characteristics our approach provides.

3 Revisiting Structure from Motion

We next review the SfM formulation that will be later used to describe our approach on rigid and non-rigid category reconstruction and grouping. Let us consider a set of P points detected on I images. Let $\mathbf{x}_p^i = [x_p^i, y_p^i, z_p^i]^\top$ be the 3D coordinates of the p-th point in image i, and $\mathbf{w}_p^i = [u_p^i, v_p^i]^\top$ its 2D position according to an orthographic projection. We can jointly write the 3D-to-2D mapping of all points as the following linear system:

$$\underbrace{\begin{bmatrix} \mathbf{w}_1^1 & \dots & \mathbf{w}_P^1 \\ \vdots & \ddots & \vdots \\ \mathbf{w}_1^I & \dots & \mathbf{w}_P^I \end{bmatrix}}_{\mathbf{W}} = \underbrace{\begin{bmatrix} \mathbf{R}^1 & & \\ & \ddots & \\ & & \mathbf{R}^I \end{bmatrix}}_{\mathbf{G}} \underbrace{\begin{bmatrix} \mathbf{x}_1^1 & \dots & \mathbf{x}_P^1 \\ \vdots & \ddots & \vdots \\ \mathbf{x}_1^I & \dots & \mathbf{x}_P^I \end{bmatrix}}_{\hat{\mathbf{X}}} + \mathbf{T}, \tag{1}$$

where $\mathbf{W}$ is a $2I \times P$ matrix with the 2D measurements arranged in columns, $\mathbf{G}$ is a $2I \times 3I$ block diagonal matrix made of I truncated 2×3 camera rotations $\mathbf{R}^i$, $\hat{\mathbf{X}}$ is a $3I \times P$ matrix with the 3D locations of the points for all the collection, also arranged in columns, and $\mathbf{T}$ is a $2I \times P$ matrix that stacks P copies of the I bi-dimensional translation vectors $\mathbf{t}^i$. The SfM problem consists in recovering the 3D shape $\hat{\mathbf{X}}$, along with the camera motion $\{\mathbf{R}^i, \mathbf{t}^i\}$ with $i = \{1, \dots, I\}$, from 2D point detections $\mathbf{W}$.

When a rigid object is observed, i.e., $\mathbf{x}_p^1 = \mathbf{x}_p^2 = \dots = \mathbf{x}_p^I$, the shape matrix can be simplified. In this case, the shape can be estimated by applying SVD-based factorization strategies, and enforcing a 3-rank constraint on $\mathbf{W}$ [44], [54] together with orthonormality constraints on $\mathbf{G}$. If, by contrast, the observed object was non-rigid, the I locations of every point can be potentially different. Then, shape and motion can be retrieved by enforcing a $3K$-rank decomposition over the measurement matrix $\mathbf{W}$ [13], [57], where K represents the rank of a linear subspace.

For later computations, we will also re-arrange the elements of $\hat{\mathbf{X}}$ into a new $3P \times I$ matrix $\mathbf{X}$ encoding the x, y and z coordinates in different rows. Both matrices can be related through a function $q(\cdot)$ such that $\hat{\mathbf{X}} = q(\mathbf{X})$ [6], [22], [27], [29]. This new interpretation has the advantage of allowing for a K-rank decomposition, rather than $3K$, avoiding the use of unnecessary degrees of freedom.

4 Shape as Multiple Unions of Subspaces

This section describes the deformation model we propose to represent the 3D shape of an unknown number of objects belonging to a specific family and their relation with the 2D measurements in a collection of images. In the following we shall consider three types of scenarios depending on the nature of the deformation: rigid objects, and non-rigid ones with small and large deformations, respectively.

4.1 Type 1: Rigid Objects

Let us consider a collection of I images of a number of rigid objects that belong to the same category (e.g., *bus* in Fig. 3-left). Each object is characterized by P semantic 3D points, which, for the moment, we will assume to be all visible in all images. The number of objects and images per object is not known a priori. Our goal is, given the 2D annotations, to reconstruct the 3D position of the P points in all images, and identify and group the images belonging to the same object. When only considering one single object instance, the problem becomes a standard rigid SfM [44], [54] (see the images #1 and #2 in Fig. 3-left), which we will not tackle in this paper. When more than one type of object is considered, we can consider their P semantic points to be related by a geometric transformation that includes both a rigid and a non-rigid deformation. Reconstructing the P points can then be addressed in a NRSfM context, although without enforcing temporal consistency between consecutive images.

Assuming a single low-rank constraint could be sufficient to span the solution space of the 3D shape in this case, as was shown in [28]. However, this formulation is very sensitive to the chosen rank of the subspace, and its optimal value may be very difficult to discover

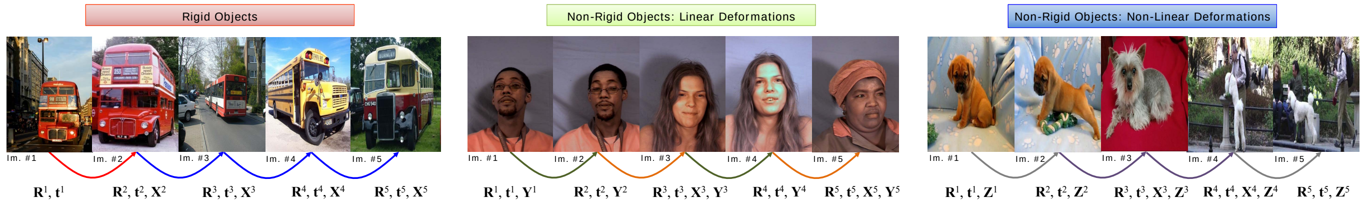
Rigid Objects
Non-Rigid Objects: Linear Deformations
Non-Rigid Objects: Non-Linear Deformations
Im. #1
Im. #2
Im. #3
Im. #4
Im. #5
$\mathbf{R}^1, \mathbf{t}^1$
$\mathbf{R}^2, \mathbf{t}^2, \mathbf{X}^2$
$\mathbf{R}^3, \mathbf{t}^3, \mathbf{X}^3$
$\mathbf{R}^4, \mathbf{t}^4, \mathbf{X}^4$
$\mathbf{R}^5, \mathbf{t}^5, \mathbf{X}^5$
Im. #1
Im. #2
Im. #3
Im. #4
Im. #5
$\mathbf{R}^1, \mathbf{t}^1, \mathbf{Y}^1$
$\mathbf{R}^2, \mathbf{t}^2, \mathbf{Y}^2$
$\mathbf{R}^3, \mathbf{t}^3, \mathbf{X}^3, \mathbf{Y}^3$
$\mathbf{R}^4, \mathbf{t}^4, \mathbf{Y}^4$
$\mathbf{R}^5, \mathbf{t}^5, \mathbf{X}^5, \mathbf{Y}^5$
Im. #1
Im. #2
Im. #3
Im. #4
Im. #5
$\mathbf{R}^1, \mathbf{t}^1, \mathbf{Z}^1$
$\mathbf{R}^2, \mathbf{t}^2, \mathbf{Z}^2$
$\mathbf{R}^3, \mathbf{t}^3, \mathbf{X}^3, \mathbf{Z}^3$
$\mathbf{R}^4, \mathbf{t}^4, \mathbf{X}^4, \mathbf{Z}^4$
$\mathbf{R}^5, \mathbf{t}^5, \mathbf{Z}^5$

Fig. 3. **Rigid and non-rigid transformations to model 3D shape deformations of rigid and non-rigid categories from a RGB image collection.** Our deformation model can code several types of transformations. In all cases, between every pair of images, we define a 6 d.o.f. rigid motion, consisting of a rotation matrix $\mathbf{R}^i$ and a translation vector $\mathbf{t}^i$. **Left:** The geometric relation between pairs of objects in a rigid category (e.g., *bus*) can be defined in the context of a NRSfM problem using a global deformation $\mathbf{X}^i$. For this particular case, whether the objects within the same category are the same (see for instance images #1 and #2), the problem can be addressed in the context of rigid SFM. **Middle:** In some categories (e.g., *face*), everyone of the objects deforms by themselves. In this case, besides the global deformation between objects, we define a linear deformation $\mathbf{Y}^i$ to encode the non-rigid motion that each object may undergo. **Right:** Other categories (e.g., *dog*) follow more complex patterns. In this case we consider a non-linear deformation $\mathbf{Z}^i$. Our deformation model jointly considers all types of deformations and automatically learns the contribution of each term to describe the geometry of the objects in a specific category. Images in this figure are taken from the PASCAL VOC [26], MUCT [45], and TigDog [24] datasets, respectively.

when the number of objects is unknown. Additionally, the maximum rank, and hence the expressiveness of the subspace, is limited by construction by the number of semantic points P, which in most of our scenarios is rather small. To overcome these difficulties, we introduce a formulation that models deformation using a union of subspaces, allowing to automatically represent a wide range of deformations, from simple low-rank solution spaces to highly expressive ones. We mathematically write this model as:

$$\mathbf{X} = \mathbf{X}\mathbf{Q} + \mathbf{E}_1 \,, \tag{2}$$

where $\mathbf{Q}$ is a $I \times I$ affinity matrix which should have higher entries for pairs of images of the same object, and $\mathbf{E}_1$ is a $3P \times I$ residual noise matrix to avoid the trivial solution $\mathbf{Q} = \mathbf{I}_I$. In essence, by doing this, we bring the standard scenario of the rigid SfM problem to the non-rigid domain, with the additional outcome of grouping the input images into different objects, with no a priori knowledge about the dimensionality of the subspaces nor which data points belong to which subspace. As we shall see later, once the affinity matrix $\mathbf{Q}$ is recovered, spectral clustering [18] can be applied on it to discover and match the different objects within the collection.

It is worth noting that the matrix $\mathbf{X}$ should ideally have a rank equal to the number of objects in the image collection (i.e., $\mathbf{X}$ is low-rank) as long as that number is greater than the number of semantic points. This analysis can be also done with the matrix $\hat{\mathbf{X}}$, becoming the rank in this case 3 times the number of objects. In any case, as we assume in our problem that the number of objects is unknown, i.e., the rank value is not provided a priori, we will enforce this constraint by directly minimizing the rank of $\mathbf{X}$.

4.2 Type 2: Non-Rigid Objects with Small Deformations

We next consider the case in which the objects, besides a rigid motion, also undergo small deformations or a partial deformation of some of their points. Figure 3-middle shows an example of such situation for faces, where most of the deformation is concentrated around the mouth and eye areas. Existing solutions address this case by enforcing a single low-rank subspace [12], [22], [23], [48], when only considering one object, or through a dual low-rank shape representation [2] when multiple objects appear in the set of images. Most these approaches, however, still require accurately adjusting a priori the dimensionality of the subspace.

In order to account for such small and sparse deformations we will introduce a matrix $\mathbf{Y} \in \mathbb{R}^{3P \times I}$ in our model. In contrast to the aforementioned approaches, no low-rank constraint will be enforced, but only a sparsity constraint that allows the deformation of just a few points.

4.3 Type 3: Non-Rigid Objects with Large Deformations

We finally consider the case in which the images correspond to a number of non-rigid objects of a given category, that can potentially undergo large deformations. The articulated motion of humans or animals (see Fig. 3-right) are examples of this scenario. In addition to the unknown number of objects in the category, we also assume the number of actions or poses not to be known.

In order to model this situation, we require a model with large expressibility. This is achieved by introducing into the model a matrix $\mathbf{Z} \in \mathbb{R}^{3P \times I}$ which is enforced to be formed by another union of subspaces:

$$\mathbf{Z} = \mathbf{Z}\mathbf{Q}\mathbf{H} + \mathbf{E}_2 \,, \tag{3}$$

where $\mathbf{H}$ is again a $I \times I$ affinity matrix, and $\mathbf{E}_2$ is a residual noise one. Note that in this case we are considering the total affinity to be defined by the product $\mathbf{QH}$, that is, we consider affinities between type of deformations. Like mentioned before for the matrix $\mathbf{Q}$, applying spectral clustering on the affinity $\mathbf{QH}$ will yield groups of objects with similar deformation (e.g., animal #1 or #2 standing, animal #1 or #2 sitting). Again, as the number of deformation groups is not known, we enforce the low-rank constraint by minimizing the rank of $\mathbf{Z}$.

5 3D Shape, Motion and Grouping per Object and Deformation Type

Our goal is to jointly recover 3D shape, camera motion, and object and deformation type from partial 2D annotations. In this section we formulate this problem by integrating the three deformation types discussed above (and hence, we will use the name of Multiple Unions of Subspaces –MUS– to denote our approach) into the 3D-to-2D projection model defined in Eq. (1), along with the orthogonality constraints. We then describe two optimization schemes we propose to solve it.

5.1 Problem Formulation

Let $\bar{\mathbf{W}}$ be a possibly incomplete matrix of 2D annotations (recall that I is the number of images of an object class and P the number of points defining the class), and $\mathbf{O}$ the corresponding $I \times P$ observation matrix with $\{1, 0\}$ entries indicating whether a specific point in an image is observed or not. Given $\bar{\mathbf{W}}$ and $\mathbf{O}$, we aim at recovering: 1) the 3D locations of all points in all images, encoded by the shape matrices $\mathbf{X}$, $\mathbf{Y}$ and $\mathbf{Z}$ defined in Section 4; 2) the object specific $\mathbf{Q}$ and deformation specific $\mathbf{QH}$ affinity matrices which we shall use later for grouping; 3) the camera pose parameters $(\mathbf{G}, \mathbf{T})$ in all images; and 4) the complete 2D detections matrix $\mathbf{W}$. We denote all these unknown parameters, plus the corresponding noise matrices by $\mathbf{\Psi} \equiv \{\mathbf{W}, \mathbf{G}, \mathbf{T}, \mathbf{Q}, \mathbf{H}, \mathbf{X}, \mathbf{Y}, \mathbf{Z}, \mathbf{E}_1, \mathbf{E}_2\}$.

In order to tackle this problem we propose optimizing a cost function that enforces the correct reprojection of the estimated 3D shape onto the image and incorporates the shape constraints we mentioned when describing the model in Section 4. In particular, the matrices $\mathbf{X}$ and $\mathbf{Z}$ are enforced to lie in low-rank subspaces. Since rank minimization is a non-convex NP-hard problem [51], the nuclear norm is used as a convex relaxation [16], [19]. Sparsity on the component $\mathbf{Y}$ is encouraged through l_1-norm minimization. Additionally, we consider the mixed $l_{2,1}$-norm over the matrices of residual noise $\mathbf{E}_1$ and $\mathbf{E}_2$, as this type of norm favors structured sparsity. Note that structured noise patterns may occur on the shape matrices $\mathbf{X}$ and $\mathbf{Z}$ when specific data points are missing or corrupted by noise. Taking all this into consideration we formulate the optimization problem as follows:

$$\begin{aligned}\arg\min_{\mathbf{\Psi}} \; & \| (\mathbf{O} \otimes \mathbf{1}_2) \odot \left(\mathbf{W} - \bar{\mathbf{W}}\right) \|_F^2 + \beta \|\mathbf{W}\|_* + \phi \|\mathbf{Q}\|_* \\ & + \gamma (\|\mathbf{X}\|_* + \|\mathbf{Y}\|_1 + \|\mathbf{Z}\|_*) + \phi \|\mathbf{H}\|_* \\ & + \lambda (\|\mathbf{E}_1\|_{2,1} + \|\mathbf{E}_2\|_{2,1}) \qquad (4) \\ \text{subject to} \quad & \mathbf{W} = \mathbf{G}\, q(\mathbf{X} + \mathbf{Y} + \mathbf{Z}) + \mathbf{T} \\ & \mathbf{G}\mathbf{G}^\top = \mathbf{I}_{2I} \\ & \mathbf{X} = \mathbf{X}\mathbf{Q} + \mathbf{E}_1 \\ & \mathbf{Z} = \mathbf{Z}\mathbf{Q}\mathbf{H} + \mathbf{E}_2\end{aligned}$$

where $\otimes$ and $\odot$ represent the Kronecker and Hadamard products, respectively. $\mathbf{1}$ is a vector of ones, and $\mathbf{I}$ the identity matrix. $\|\cdot\|_F$ indicates the Frobenius norm, $\|\cdot\|_*$ denotes the nuclear norm, and $\|\cdot\|_1$, and $\|\cdot\|_{2,1}$ are the l_1-norm and $l_{2,1}$-norm, respectively. Finally, $\{\beta, \phi, \gamma, \lambda\}$ represent the set of penalty weights. As it can be seen, the projection system in Eq. (1) is coded by the first constraint in our full energy in Eq. (4). In order to obtain symmetric affinity matrices, we could impose directly the constraint $\mathbf{Q} = \mathbf{Q}^\top$ in our optimization. However, this results in increasing the computational complexity of our algorithm and, in practice, the performance is not better than applying a post-symmetrization of $\mathbf{Q}$. Therefore, and following other approaches in the literature [61], we will not enforce this constraint in the optimization.

As it can be seen in Eq. (4), we do not include in our formulation temporal smoothness priors neither in the shape deformation nor in the affinities [4]. While this type of priors could provide more accurate solutions for some collections, in general terms, we cannot assume these priors for an arbitrary image collection, where the order of the images within the collection is normally random. As we learn the deformation model directly from data, without assuming any training data, it is worth noting that once learned, we could apply it in order to infer unseen instances in the same category, by enforcing the multiple unions of subspaces we recover.

In this paper, we propose two algorithms to minimize the cost function in Eq. (4). First, we present a 3-step factorization approach in which: 1) complete missing entries $\mathbf{W}$; 2) estimate camera pose parameters $\{\mathbf{G}, \mathbf{T}\}$, and 3) recover the 3D object reconstruction $q(\mathbf{X}+\mathbf{Y}+\mathbf{Z})$, and perform grouping per object $\mathbf{Q}$ and type of deformation $\mathbf{QH}$. This algorithm is denoted as MUS, and will be described in section 5.2. Our second algorithm solves the problem jointly estimating the model parameters in the same loop, instead of fixing the camera parameters in an early step. This algorithm is denoted as MUS2, and it will be described in section 5.3.

5.2 MUS: A 3-Step Factorization Strategy

We next present the three main steps of the MUS algorithm.

5.2.1 *Recovering 2D Missing Annotations*

To complete the unobserved 2D annotations of $\bar{\mathbf{W}}$ (zeros in the observation matrix $\mathbf{O}$), we independently optimize $\mathbf{W}$ in the first two terms of Eq. (4) while enforcing this matrix to be low rank. As shown in [6], [11], [14], this optimization can be done by means of bilinear factorization, defining $\mathbf{W} = \mathbf{U}\mathbf{V}^\top$. We write the equivalent problem as:

$$\begin{aligned}\arg\min_{\mathbf{W},\mathbf{U},\mathbf{V}} \; & \| (\mathbf{O} \otimes \mathbf{1}_2) \odot \left(\mathbf{W} - \bar{\mathbf{W}}\right) \|_F^2 + \frac{\beta}{2} \left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right) \\ \text{subject to} \quad & \mathbf{W} = \mathbf{U}\mathbf{V}^\top \qquad (5)\end{aligned}$$

This can be efficiently solved via ALM. To improve convergence, the missing entries of $\bar{\mathbf{W}}$ are initialized in every image as the mean value of the observed data points.

5.2.2 Retrieving Camera Motion

Once the missing observations are estimated, the camera translation $\mathbf{t}^i$ and rotation $\mathbf{R}^i$ in every image can be inferred from the rest of model parameters. For this purpose, we first estimate the translations in $\mathbf{T}$ as $\mathbf{t}^i = \frac{1}{P}\sum_{p=1}^{P}\mathbf{w}_p^i$. The rotations matrices in $\mathbf{G}$ can then be jointly estimated by solving the following non-convex problem:

$$\begin{aligned}&\arg\min_{\mathbf{G}} \frac{1}{2}\|\mathbf{W}-\mathbf{T}-\mathbf{G}\hat{\mathbf{X}}\|_F^2 \qquad (6)\\ &\text{subject to} \quad \mathbf{G}\mathbf{G}^\top = \mathbf{I}_{2I}\end{aligned}$$

where the constraint enforces the camera rotation matrices to be orthonormal. This optimization is solved by factorization, using different values of rank and stopping automatically when there is no additional improvement in the average camera orthonormality.

5.2.3 Simultaneous 3D Reconstruction and Grouping

We finally formulate the problem of simultaneously retrieving 3D shape in all images as well as the type of object and deformation grouping. Assuming the matrices $\mathbf{W}$, $\mathbf{G}$ and $\mathbf{T}$ to be known, the optimization problem that needs to be solved becomes:

$$\begin{aligned}\arg\min_{\mathbf{\Psi}'} \;& \gamma(\|\mathbf{X}\|_*+\|\mathbf{Y}\|_1+\|\mathbf{Z}\|_*)+\|\mathbf{Q}\|_*+\|\mathbf{H}\|_*\\ &+\lambda(\|\mathbf{E}_1\|_{2,1}+\|\mathbf{E}_2\|_{2,1}) \qquad (7)\\ \text{subject to} \quad & \mathbf{W}=\mathbf{G}\left(\mathbf{A}+\mathbf{B}+\mathbf{C}\right)+\mathbf{T}\\ & \mathbf{X}=\mathbf{X}\mathbf{Q}+\mathbf{E}_1\\ & \mathbf{Z}=\mathbf{Z}\mathbf{F}+\mathbf{E}_2\\ & q(\mathbf{X})=\mathbf{A}\\ & q(\mathbf{Y})=\mathbf{B}\\ & q(\mathbf{Z})=\mathbf{C}\\ & \mathbf{F}=\mathbf{Q}\mathbf{H}\end{aligned}$$

where $\mathbf{\Psi}' \equiv \{\mathbf{Q},\mathbf{H},\mathbf{F},\mathbf{X},\mathbf{A},\mathbf{Y},\mathbf{B},\mathbf{Z},\mathbf{C},\mathbf{E}_1,\mathbf{E}_2\}$. Note that compared to the original Eq. (4) we have included three additional constraints, namely $q(\mathbf{X})=\mathbf{A}$, $q(\mathbf{Y})=\mathbf{B}$ and $q(\mathbf{Z})=\mathbf{C}$, where $q(\cdot)$ simply rearranges the elements of a matrix as discussed in Section 3. Furthermore, to reduce the computational burden, we have included the constraint $\mathbf{F}=\mathbf{QH}$. Without loss of generality we have also reduced the number of weight parameters originally appearing in Eq. (4), by setting $\phi = 1$ and re-scaling the rest. In order to solve Eq. (7), we again resort to the ALM framework, and writing the equivalent Lagrangian function as:

$$\arg\min_{\mathbf{\Psi}_{\text{MUS}}} \;\{\text{CostMUS}\} \qquad (8)$$

where:

$$\begin{aligned}\text{CostMUS} =&\gamma(\|\mathbf{X}\|_*+\|\mathbf{Y}\|_1+\|\mathbf{Z}\|_*)+\|\mathbf{J}\|_*+\|\mathbf{K}\|_*\\ &+\lambda(\|\mathbf{E}_1\|_{2,1}+\|\mathbf{E}_2\|_{2,1})\\ &+\langle\mathbf{L}_1,\mathbf{P}-\mathbf{G}(\mathbf{A}+\mathbf{B}+\mathbf{C})-\mathbf{T}\rangle+\langle\mathbf{L}_4,q(\mathbf{X})-\mathbf{A}\rangle\\ &+\langle\mathbf{L}_2,\mathbf{X}-\mathbf{X}\mathbf{Q}-\mathbf{E}_1\rangle+\langle\mathbf{L}_3,\mathbf{Z}-\mathbf{Z}\mathbf{F}-\mathbf{E}_2\rangle\\ &+\langle\mathbf{L}_5,q(\mathbf{Y})-\mathbf{B}\rangle+\langle\mathbf{L}_6,q(\mathbf{Z})-\mathbf{C}\rangle\\ &+\langle\mathbf{L}_7,\mathbf{F}-\mathbf{Q}\mathbf{H}\rangle+\langle\mathbf{L}_8,\mathbf{Q}-\mathbf{J}\rangle+\langle\mathbf{L}_9,\mathbf{H}-\mathbf{K}\rangle\\ &+\frac{\alpha}{2}(\|\mathbf{P}-\mathbf{G}(\mathbf{A}+\mathbf{B}+\mathbf{C})-\mathbf{T}\|_F^2+\|q(\mathbf{X})-\mathbf{A}\|_F^2)\\ &+\frac{\alpha}{2}(\|\mathbf{X}-\mathbf{X}\mathbf{Q}-\mathbf{E}_1\|_F^2+\|\mathbf{Z}-\mathbf{Z}\mathbf{F}-\mathbf{E}_2\|_F^2)\\ &+\frac{\alpha}{2}(\|q(\mathbf{Y})-\mathbf{B}\|_F^2+\|q(\mathbf{Z})-\mathbf{C}\|_F^2+\|\mathbf{F}-\mathbf{Q}\mathbf{H}\|_F^2)\\ &+\frac{\alpha}{2}(\|\mathbf{Q}-\mathbf{J}\|_F^2+\|\mathbf{H}-\mathbf{K}\|_F^2) \qquad (9)\end{aligned}$$

with $\mathbf{\Psi}_{MUS} \equiv \{\mathbf{J},\mathbf{Q},\mathbf{K},\mathbf{H},\mathbf{F},\mathbf{X},\mathbf{A},\mathbf{Y},\mathbf{B},\mathbf{Z},\mathbf{C},\mathbf{E}_1,\mathbf{E}_2\}$. For later computations, we also include the supporting matrices $\mathbf{Q} \equiv \mathbf{J}$ and $\mathbf{H} \equiv \mathbf{K}$. The Lagrange multipliers are defined as: $\mathbf{L}_1 \in \mathbb{R}^{2I\times P}$, $\{\mathbf{L}_2,\mathbf{L}_3\} \in \mathbb{R}^{3P\times I}$, $\{\mathbf{L}_4,\mathbf{L}_5,\mathbf{L}_6\} \in \mathbb{R}^{3I\times P}$, and $\{\mathbf{L}_7,\mathbf{L}_8,\mathbf{L}_9\} \in \mathbb{R}^{I\times I}$; and $\alpha > 0$ is a penalty coefficient to improve convergence.

The previous optimization problem can be efficiently tackled by means of partial subproblems independently resolved in closed form. The outline of the algorithm is shown in Algorithm 1. The closed-form solutions for computing $\mathbf{Q}$, $\mathbf{H}$, $\mathbf{F}$, $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ are obtained solving the derivatives of Eq. (8) with respect to $\mathbf{Q}$, $\mathbf{H}$, $\mathbf{F}$, $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$, respectively, and equating to zero. For $\mathbf{J}$, $\mathbf{K}$, $\mathbf{X}$ and $\mathbf{Z}$ matrices, it is required to solve a singular-value-thresholding minimization problem [15]. The optimization of $\mathbf{Y}$ can be done in closed form by the element-wise shrinkage operator [42]. Finally, to optimize the noise terms $\mathbf{E}_1$ and $\mathbf{E}_2$, we apply the Lemma 4.1 in [59]. It is worth noting that after updating, the Lagrange multipliers are also modified.

5.3 MUS2: Joint 3D Shape, Motion and Grouping in the Same Loop

We now describe MUS2, which in contrast to MUS, iteratively updates camera-motion parameters in one single iterative loop. To this end, we have to solve the following problem:

$$\arg\min_{\mathbf{\Psi}_{\text{MUS2}}} \;\{\text{CostMUS2}\} \qquad (10)$$

where:

$$\text{CostMUS2} = \text{CostMUS}+\|\mathbf{G}\mathbf{G}^\top-\mathbf{I}_{2I}\|_F^2\,,$$

and $\mathbf{\Psi}_{\text{MUS2}} = \mathbf{\Psi}_{\text{MUS}} \cup \{\mathbf{G}\}$. The previous cost function of MUS2 is a generalization of that for MUS in Eq. (8), with additional update rules for camera motion. That is, the full energy function in Eq. (4) is minimized in a unified fashion. To perform initialization on this new version, we propose to solve Eqs. (5)-(6).

In order to solve for camera rotation, we re-write our problem as:

$$\underset{\mathbf{M}^i \in \text{SO}(3)}{\arg\min} \sum_{i=1}^{I}\sum_{p=1}^{P} \|\mathbf{w}_p^i - \mathcal{M}\mathbf{M}^i(\mathbf{a}_p^i + \mathbf{b}_p^i + \mathbf{c}_p^i) - \mathbf{t}_p^i\|_F^2 \,, \quad (11)$$

where $\mathbf{M}^i \in \text{SO}(3)$ is a 3×3 full camera rotation matrix, with $\mathbf{R}^i = \mathcal{M}\mathbf{M}^i$ and $\mathcal{M} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$. For simplicity, in this section we will use $\mathbf{x}_p^i$ to denote $\mathbf{x}_p^i = \mathbf{a}_p^i + \mathbf{b}_p^i + \mathbf{c}_p^i$, where $\mathbf{a}_p^i$, $\mathbf{b}_p^i$ and $\mathbf{c}_p^i$ represent the p-th point onto the i-th image in $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ shape matrices, respectively.

As every $\mathbf{M}^i$ is subject to an orthogonality constraint (recall that every rotation matrix lies in a smooth manifold based on the orthogonal group SO(3)), obtaining a closed-form solution is not possible. To solve this, we present a Riemannian-Newton algorithm [25], [53] to enforce every rotation matrix to be a Stiefel matrix. Let $\Delta_{\mathbf{M}^i} \in T_{\mathbf{M}}(SO(3))$ be the tangent of $\mathbf{M}^i$, that it can be expressed as $\Delta_{\mathbf{M}^i} = \mathbf{M}^i\,[\boldsymbol{\delta}]_\times$, with $[\boldsymbol{\delta}]_\times$ being a skew-symmetric matrix. As $\mathbf{M}^i$ lies in SO(3), a geodesic at $\mathbf{M}^i$ in the tangent direction can be coded by means of the Rodrigues' formula as:

$$\mathbf{M}^i(\hat{\boldsymbol{\delta}}, \omega) = \mathbf{M}^i\left(\mathbf{I}_3 + \left[\hat{\boldsymbol{\delta}}\right]_\times \sin(\omega) + \left[\hat{\boldsymbol{\delta}}\right]_\times^2 (1 - \cos(\omega))\right),$$

where $[\boldsymbol{\delta}]_\times \in \mathfrak{so}(3)$ is the Lie algebra of the SO(3) group, and $[\boldsymbol{\delta}]_\times = \omega\left[\hat{\boldsymbol{\delta}}\right]_\times$. Thanks to the previous definition, we can compute both gradient $\nabla f(\cdot)$ and Hessian Hess$f(\cdot)$ expressions (for $\omega = 0$) in Eq. (11) and in a tangent direction $\Delta_{\mathbf{M}^i}$ as:

$$\nabla f(\Delta_{\mathbf{M}^i}) = \Big(\mathbf{R}^i \sum_{p=1}^{P}((\mathbf{x}_p^i)(\mathbf{x}_p^i)^\top) - \sum_{p=1}^{P}((\mathbf{w}_p^i - \mathbf{t}_p^i)(\mathbf{x}_p^i)^\top)\Big)\Delta_{\mathbf{R}^i}^\top \,,$$

$$\text{Hess} f(\Delta_{\mathbf{M}^i}, \Delta_{\mathbf{M}^i}) =$$

$$\Big(\mathbf{R}^i \sum_{p=1}^{P}((\mathbf{x}_p^i)(\mathbf{x}_p^i)^\top) - \sum_{p=1}^{P}((\mathbf{w}_p^i - \mathbf{t}_p^i)(\mathbf{x}_p^i)^\top)\Big)\Delta_{\mathbf{M}^i}^\top \mathbf{M}^i \Delta_{\mathbf{R}^i}^\top$$

$$+ \Delta_{\mathbf{R}^i} \sum_{p=1}^{P}((\mathbf{x}_p^i)(\mathbf{x}_p^i)^\top)\Delta_{\mathbf{R}^i}^\top \,,$$

where $\Delta_{\mathbf{R}^i} = \mathcal{M}\Delta_{\mathbf{M}^i}$ represents the first two rows of a full tangent vector $\Delta_{\mathbf{M}^i}$. With these ingredients, we can now propose a Riemannian-Newton algorithm for updating the camera rotation. The outline of our MUS2 algorithm is displayed in Alg. 2.

5.4 Object and Deformation Grouping

Once the similarity matrices $\mathbf{Q}$ and $\mathbf{F}$ are recovered from an image collection, we apply a post-symmetrization step, using $(|\mathbf{Q}| + |\mathbf{Q}^\top|)$ and $(|\mathbf{F}| + |\mathbf{F}^\top|)$ as the affinity matrices, respectively. After that, we run the spectral clustering algorithm proposed in [18] to find subspace segmentation. Figure 1 shows an instance of two matrices that we obtain, where each entry (a, b) indicates the degree of affinity between the a-th and b-th image object for $\mathbf{Q}$, or between the a-th and b-th deformation type for $\mathbf{F}$. To improve qualitative evaluation, we also

```
Input : Incomplete 2D annotations $\bar{\mathbf{W}}$, observation
        matrix $\mathbf{O}$, and penalty weights $\gamma$ and $\lambda$
Output: Full observations $\mathbf{W}$, 3D reconstruction
        $\{\mathbf{A}+\mathbf{B}+\mathbf{C}\}$, camera pose $\{\mathbf{G},\mathbf{T}\}$, and
        object $\mathbf{Q}$ and deformation $\mathbf{F}$ grouping
/* Complete Missing Entries, Eq. (5)                */
/* Estimate Camera Pose $\{\mathbf{G},\mathbf{T}\}$, Eq. (6)   */
/* 3D Shape $\{\mathbf{A}+\mathbf{B}+\mathbf{C}\}$ and grouping
   $\{\mathbf{Q},\mathbf{F}\}$, Eq. (7), by solving the ALM
   problem in Eq. (8)                                */
while not converged do
    /* Update Model Parameters                        */
    $\mathbf{J} = \min \frac{1}{\alpha}\|\mathbf{J}\|_* + \frac{1}{2}\|\mathbf{J} - (\mathbf{Q} + \frac{\mathbf{L}_8}{\alpha})\|_F^2$
    $\mathbf{D} = \mathbf{X}^\top(\mathbf{X} - \mathbf{E}_1 + \frac{\mathbf{L}_2}{\alpha}) + \mathbf{F}\mathbf{H}^\top + \mathbf{J} + \frac{\mathbf{L}_7}{\alpha}\mathbf{H}^\top - \frac{\mathbf{L}_8}{\alpha}$
    $\text{vec}(\mathbf{Q}) = \big(\mathbf{I}_I \otimes (\mathbf{X}^\top\mathbf{X} + \mathbf{I}_I) + \mathbf{H}\mathbf{H}^\top \otimes \mathbf{I}_I\big)^{-1}\text{vec}(\mathbf{D})$
    $\mathbf{Q} = \text{mat}(\text{vec}(\mathbf{Q}))$
    $\mathbf{K} = \min \frac{1}{\alpha}\|\mathbf{J}\|_* + \frac{1}{2}\|\mathbf{J} - (\mathbf{H} + \frac{\mathbf{L}_9}{\alpha})\|_F^2$
    $\mathbf{H} = (\mathbf{Q}^\top\mathbf{Q} + \mathbf{I}_I)^{-1}\big(\mathbf{Q}^\top\mathbf{F} + \mathbf{K} + (\frac{\mathbf{Q}^\top\mathbf{L}_7 - \mathbf{L}_9}{\alpha})\big)$
    $\mathbf{F} = (\mathbf{Z}^\top\mathbf{Z} + \mathbf{I}_I)^{-1}\big(\mathbf{Z}^\top(\mathbf{Z} - \mathbf{E}_2) + \mathbf{Q}\mathbf{H} + \frac{\mathbf{Z}^\top\mathbf{L}_3 - \mathbf{L}_7}{\alpha}\big)$
    $\mathbf{X} = \min \frac{\gamma}{\alpha}\|\mathbf{X}\|_* + \frac{1}{2}\|\mathbf{X} - \big((\mathbf{E}_1 - \frac{\mathbf{L}_2}{\alpha})(\mathbf{I}_I - \mathbf{Q})^\top +$
      $q^{-1}(\mathbf{A} - \frac{\mathbf{L}_4}{\alpha})\big)\big((\mathbf{I}_I - \mathbf{Q})(\mathbf{I}_I - \mathbf{Q})^\top + \mathbf{I}_I\big)^{-1}\|_F^2$
    $\mathbf{A} = \mathbf{N}^{-1}\big(\mathbf{G}^\top(\mathbf{W} + \frac{\mathbf{L}_1}{\alpha} - \mathbf{G}(\mathbf{B} + \mathbf{C})) + \frac{\mathbf{L}_4}{\alpha} + q(\mathbf{X})\big)$
    $\mathbf{Y} = \min \frac{\gamma}{\alpha}\|\mathbf{Y}\|_1 + \frac{1}{2}\|\mathbf{Y} - q^{-1}(\mathbf{B} - \frac{\mathbf{L}_5}{\alpha})\|_F^2$
    $\mathbf{B} = \mathbf{N}^{-1}\big(\mathbf{G}^\top(\mathbf{W} + \frac{\mathbf{L}_1}{\alpha} - \mathbf{G}(\mathbf{A} + \mathbf{C})) + \frac{\mathbf{L}_5}{\alpha} + q(\mathbf{Y})\big)$
    $\mathbf{Z} = \min \frac{\gamma}{\alpha}\|\mathbf{Z}\|_* + \frac{1}{2}\|\mathbf{Z} - \big((\mathbf{E}_2 - \frac{\mathbf{L}_3}{\alpha})(\mathbf{I}_I - \mathbf{F})^\top +$
      $q^{-1}(\mathbf{C} - \frac{\mathbf{L}_6}{\alpha})\big)\big((\mathbf{I}_I - \mathbf{F})(\mathbf{I}_I - \mathbf{F})^\top + \mathbf{I}_I\big)^{-1}\|_F^2$
    $\mathbf{C} = \mathbf{N}^{-1}\big(\mathbf{G}^\top(\mathbf{W} + \frac{\mathbf{L}_1}{\alpha} - \mathbf{G}(\mathbf{B} + \mathbf{A})) + \frac{\mathbf{L}_6}{\alpha} + q(\mathbf{Z})\big)$
    $\mathbf{E}_1 = \min \frac{\lambda}{\alpha}\|\mathbf{E}_1\|_{2,1} + \frac{1}{2}\|\mathbf{E}_1 - (\mathbf{X} - \mathbf{X}\mathbf{Q} + \frac{\mathbf{L}_2}{\alpha})\|_F^2$
    $\mathbf{E}_2 = \min \frac{\lambda}{\alpha}\|\mathbf{E}_2\|_{2,1} + \frac{1}{2}\|\mathbf{E}_2 - (\mathbf{Z} - \mathbf{Z}\mathbf{F} + \frac{\mathbf{L}_3}{\alpha})\|_F^2$
    /* Update Lagrange Multipliers                    */
    $\mathbf{L}_1 = \mathbf{L}_1 + \alpha(\mathbf{W} - \mathbf{G}(\mathbf{A} + \mathbf{B} + \mathbf{C}) - \mathbf{T})$
    $\mathbf{L}_2 = \mathbf{L}_2 + \alpha(\mathbf{X} - \mathbf{X}\mathbf{Q} - \mathbf{E}_1)$
    $\mathbf{L}_3 = \mathbf{L}_3 + \alpha(\mathbf{Z} - \mathbf{Z}\mathbf{F} - \mathbf{E}_2)$
    $\mathbf{L}_4 = \mathbf{L}_4 + \alpha(q(\mathbf{X}) - \mathbf{A})$
    $\mathbf{L}_5 = \mathbf{L}_5 + \alpha(q(\mathbf{Y}) - \mathbf{B})$
    $\mathbf{L}_6 = \mathbf{L}_6 + \alpha(q(\mathbf{Z}) - \mathbf{C})$
    $\mathbf{L}_7 = \mathbf{L}_7 + \alpha(\mathbf{F} - \mathbf{Q}\mathbf{H})$
    $\mathbf{L}_8 = \mathbf{L}_8 + \alpha(\mathbf{Q} - \mathbf{J})$
    $\mathbf{L}_9 = \mathbf{L}_9 + \alpha(\mathbf{H} - \mathbf{K})$
    /* Update Penalty Weights                         */
    $\alpha = \min(\eta\alpha, 10^{12})$
    /* Check Convergence                              */
    $\|\mathbf{W} - \mathbf{G}(\mathbf{A} + \mathbf{B} + \mathbf{C}) - \mathbf{T}\|_\infty < \epsilon$
    $\|\mathbf{X} - \mathbf{X}\mathbf{Q} - \mathbf{E}_1\|_\infty < \epsilon$
    $\|\mathbf{Z} - \mathbf{Z}\mathbf{F} - \mathbf{E}_2\|_\infty < \epsilon$
    $\|q(\mathbf{X}) - \mathbf{A}\|_\infty < \epsilon$
    $\|q(\mathbf{Y}) - \mathbf{B}\|_\infty < \epsilon$
    $\|q(\mathbf{Z}) - \mathbf{C}\|_\infty < \epsilon$
    $\|\mathbf{F} - \mathbf{Q}\mathbf{H}\|_\infty < \epsilon$
    $\|\mathbf{Q} - \mathbf{J}\|_\infty < \epsilon$
    $\|\mathbf{H} - \mathbf{K}\|_\infty < \epsilon$
end
Not.: $\mathbf{N} = \mathbf{G}^\top\mathbf{G} + \mathbf{I}_{3I}$, $\eta = 1.1$, $\alpha = 10^{-2}$ and $\epsilon = 10^{-7}$.
Matrices $\mathbf{L}_c$, $c = \{1, \ldots, 9\}$, are initially set to zero.
```

Algorithm 1: MUS algorithm for optimizing Eq. (4). vec$(\cdot)$ and mat$(\cdot)$ are vectorization and matrization operators, respectively.

```
input : Incomplete 2D annotations $\bar{\mathbf{W}}$, observation
        matrix $\mathbf{O}$, and penalty weights $\gamma$ and $\lambda$
output: Full observations $\mathbf{W}$, 3D reconstruction
        $\{\mathbf{A}+\mathbf{B}+\mathbf{C}\}$, camera pose $\{\mathbf{G},\mathbf{T}\}$, and object
        $\mathbf{Q}$ and deformation $\mathbf{F}$ grouping
/* Initialization: Eqs. (5)–(6) */
/* ALM Optimization of Eq. (10) */
1  while not converged do
       /* Update Rules in Alg. 1 */
2      if $\|\mathbf{W}-\mathbf{G}(\mathbf{A}+\mathbf{B}+\mathbf{C})-\mathbf{T}\|_\infty < \epsilon_2$ then
3      for i=1,...,I do
           /* Optimal Updating Vector */
4          ${}^{a}\mathbf{E} = \mathbf{M}^i \left[{}^{a}\mathbf{e}\right]_\times \; 1 \leq a \leq 3$
5          ${}^{a}\mathbf{g} = \nabla f({}^{a}\mathbf{E})$
6          ${}^{aa}\mathbf{P} = \mathrm{Hess} f({}^{a}\mathbf{E}, {}^{a}\mathbf{E})$
7          $\boldsymbol{\delta} = -\mathbf{P}^{-1}({}^{aa}\mathbf{P})\mathbf{g}({}^{a}\mathbf{g})$
8          $\Delta_{\mathbf{M}^i} = \mathbf{M}^i \left[\boldsymbol{\delta}\right]_\times$
           /* Update the Rotation Matrix */
9          $\mathbf{M}^i = \mathbf{M}^i \exp\left(\omega \left[\hat{\boldsymbol{\delta}}\right]_\times\right)$
10     end
11     $\mathbf{G} = \mathrm{blkdiag}(\mathbf{R}^1(\mathbf{M}^1), \ldots, \mathbf{R}^I(\mathbf{M}^I))$
12     else $\mathbf{G} \equiv \mathbf{G}$
13 end
14 Not.: ${}^{a}\mathbf{E}$ represents an orthonormal basis of the
   tangent space on SO(3), with ${}^{a}\mathbf{e}$ a standard basis in
   $\mathbb{R}^3$. The parameter $\omega = \sqrt{\frac{1}{2}\mathrm{tr}\left(\Delta_{\mathbf{M}^i}^\top \Delta_{\mathbf{M}^i}\right)}$, and
   $\epsilon_2 = 25 \cdot 10^{-2}$.
```

Algorithm 2: MUS2 algorithm for optimizing Eq. (4). blkdiag$(\cdot)$ and tr$(\cdot)$ denote a block matrix operator and the trace of a matrix, respectively.

include a grouping bar for every affinity matrix we represent, where every color represents a group discovered after applying spectral clustering. The granularity of the grouping can be controlled through a threshold on the eigenvalues internally computed by [18].

5.5 Complexity Analysis

One of the main virtues of the formulation we propose is that it has a small computational load. The most computationally demanding part of the Algs. 1-2 corresponds to the steps 10, 12 and 14 in Alg. 1, which requires computing an inverse matrix of size $3I \times 3I$. However, as the matrix to be inverted is the same in the three cases, the computational burden can be reduced. Also, we must consider in this analysis the step 4, which requires to solve a very sparse linear system of order I^2. It is worth noting that even if our algorithm needs to compute several SVD operations (see steps 2, 6, 9, 11 and 13), 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 $\mathcal{O}(I^3)$ [30]. On average, the median computation time in rigid-category experiments with image collections between $105-150$ images was of $7.6-12.0$ seconds, respectively, on a commodity laptop with an Intel Core i7 processor at 2.4GHz. In order to handle larger datasets, we could use the results in [60] or extend our formulation to be employed in a sequential manner, being this a part of our future work.

6 Experimental Evaluation

We now present our experimental results for different types of scenarios, including synthetic and real image collections of rigid and non-rigid categories. We provide quantitative and qualitative evaluation and compare our approach against state-of-the-art solutions on several synthetic datasets with 3D ground truth. For quantitative evaluation, we provide a normalized mean 3D reconstruction error e_X used before in [9], [22], [32].

To evaluate the object grouping accuracy, we apply spectral clustering [18] over the estimated matrices as it was said in section 5.4, and retrieve the $I-$dimensional vector $\mathcal{G}$, where each entry is an integer representing the group index. The grouping accuracy is defined as $a_G = 1 - \frac{1}{I}\sum_{i=1}^{I}\mathbb{I}(\mathcal{G}_i \neq \mathcal{G}_i^{GT})$, where $\mathbb{I}(v)$ is the indicator function, i.e., $\mathbb{I}(v) = 1$ if v is true, and 0 otherwise, and $\mathcal{G}_i^{GT}$ is the ground truth group index of the i-th image.

6.1 Synthetic Images

We first evaluate our approach on synthetic collections of images of rigid object categories, where the 3D ground truth is obtained from the CAD models of the PASCAL VOC dataset [26]. We choose the categories which are defined by at least eight points, indicating by (I/P) the number of images and semantic points, respectively. Particularly, we consider the following image collections: Aeroplane (105/16), Bicycle (150/11), Bus (150/8), Car (150/14), Chair (150/10), Diningtable(150/8), Motorbike (150/10), and Sofa (135/12). Based on this, we evaluate our approach on those categories which contain between seven and ten objects each (see Table 2).

We compare the 3D reconstruction error of our approaches, denoted as MUS and MUS2, with two SfM baselines: TK [54] and MC [44]; as well as with nine NRSfM solutions: the shape-trajectory methods CSF [32] and KSTA [33]; the block matrix approach BMM [22], the probabilistic-normal-distribution method EM-PND [39], the temporal union of subspaces TUS [64], the grouping-based NRSfM of GBNR [27], the consensus NRSfM of CNR [40], and the deep-learning approaches DNRSM [36] and C3DPO [47]. We also include the baseline LRR [43] to obtain the object grouping from 2D annotations. The parameters of these methods were set in accordance to their original papers. We manually set the rank of the subspace for the methods CSF [32] and KSTA [33]; and did a fine hyperparameter search for [36] and [47], using the values that gave the best results. As the source code for TUS [64] is not publicly available, we used our own implementation. In this particular case, we also used our annotation completion and camera motion estimation, as the method did not address any strategy

Algorithm / Data	TK [54]	MC [44]	CSF [32]	KSTA [33]	BMM [22]	EM-PND [39]	TUS [64]	GBNR [27]	CNR [40]	DNRSM [36]	C3DPO [47]	LRR [43]	MUS		Ours (MUS2)	
Metric:	e_X	e_X	e_X	e_X	e_X	e_X	e_X	e_X	e_X	e_X	e_X	a_G	e_X	a_G	e_X	a_G
Noise-free Annotations																
Aeroplane	0.679	0.584	0.363	**0.145**	0.843	0.578	0.294	–	0.263	0.551	0.725	0.39(7)	0.261	0.95(7)	0.253	0.97(7)
Bicycle	0.309	0.440	0.424	0.442	0.308	0.763	0.182	0.221	–	1.187	1.220	0.39(10)	0.178	0.95(10)	**0.143**	0.96(10)
Bus	0.202	0.238	0.217	0.214	0.300	1.048	0.129	0.214	–	0.263	0.941	0.44(10)	0.113	0.75(10)	**0.107**	0.82(10)
Car	0.239	0.256	0.195	0.159	0.266	0.496	0.084	0.217	0.099	0.364	0.415	0.36(10)	0.078	0.87(10)	**0.069**	0.91(10)
Chair	0.356	0.447	0.398	0.399	0.357	0.687	0.211	–	–	0.501	0.535	0.39(10)	0.210	0.87(10)	**0.184**	0.89(10)
Diningtable	0.386	0.512	0.406	0.372	0.422	0.670	0.265	0.351	–	–	0.481	0.41(10)	0.264	0.86(10)	**0.246**	0.91(10)
Motorbike	0.339	0.346	0.278	0.270	0.336	0.740	0.228	0.268	–	0.448	0.507	0.41(10)	0.222	0.91(10)	**0.202**	0.93(10)
Sofa	0.381	0.390	0.409	0.298	0.279	0.692	0.179	0.264	0.214	0.281	0.366	0.44(9)	0.167	0.85(9)	**0.141**	0.89(9)
Average error:	0.361	0.402	0.336	0.287	0.388	0.709	0.196	0.256*	0.192*	0.513*	0.649	0.40	0.186	0.88	**0.168**	0.91
Relative error:	2.15	2.39	2.00	1.71	2.31	4.22	1.17	1.52*	1.14*	3.05*	3.86	–	1.11	–	**1.00**	–
Noisy Annotations																
Aeroplane	0.677	0.583	0.233	**0.183**	0.566	0.760	0.297	–	0.294	0.566	1.305	0.41(7)	0.271	0.87(7)	0.265	0.89(7)
Bicycle	0.308	0.442	0.455	0.457	0.307	0.808	0.195	0.231	–	1.252	1.212	0.38(10)	0.188	0.93(10)	**0.156**	0.94(10)
Bus	0.204	0.241	0.227	0.218	0.255	1.197	0.139	0.223	–	0.275	0.519	0.44(10)	0.122	0.80(10)	**0.117**	0.83(10)
Car	0.241	0.259	0.169	0.164	0.161	0.624	0.100	0.222	0.122	0.366	0.362	0.36(10)	0.093	0.92(10)	**0.086**	0.93(10)
Chair	0.358	0.447	0.398	0.396	0.258	0.818	0.221	–	–	0.502	0.597	0.41(10)	0.220	0.91(10)	**0.192**	0.91(10)
Diningtable	0.392	0.522	0.414	0.383	0.358	0.807	0.268	0.370	–	0.401	0.476	0.38(10)	0.267	0.89(10)	**0.241**	0.92(10)
Motorbike	0.342	0.348	0.295	0.290	0.299	0.748	0.237	0.277	–	0.492	0.508	0.41(10)	0.233	0.89(10)	**0.215**	0.93(10)
Sofa	0.384	0.392	0.303	0.294	0.240	0.726	0.188	0.271	0.228	0.285	0.311	0.42(9)	0.174	0.91(9)	**0.150**	0.92(9)
Average error:	0.363	0.404	0.312	0.298	0.305	0.811	0.206	0.266*	0.215*	0.517	0.661	0.40	0.196	0.89	**0.177**	0.91
Relative error:	2.16	2.40	1.85	1.78	1.82	4.82	1.22	1.58*	1.28*	3.08	3.93	–	1.16	–	**1.05**	–

TABLE 2

Evaluation on synthetic collections for several rigid object categories under noise-free and noisy annotations. The table reports the 3D reconstruction error e_X for the following SfM baselines: TK [54] and MC [44]; and the NRSfM baselines: CSF [32], KSTA [33], SPM [22], EM-PND [39], TUS [64], GBNR [27], CNR [40], DNRSM [36] and C3DPO [47]; and our MUS and MUS2 algorithms. In all cases, we consider full 2D annotations. The symbol "$-$" indicates the algorithm did not manage to process the sequence, and *, that the summary is obtained considering only the successful cases. Relative error is always computed with respect to MUS2 reconstruction with clean annotations, on average, the most accurate solution. In addition, for LRR [43] and our approaches we also show the grouping accuracies a_G, and the number of object groups in parentheses.

Algorithm / Data	TK [54]	MC [44]	CSF [32]	KSTA [33]	BMM [22]	EM-PND [39]	TUS [64]	GBNR [27]	CNR [40]	DNRSM [36]	C3DPO [47]	LRR [43]	MUS		Ours (MUS2)	
Metric:	e_X	e_X	e_X	e_X	e_X	e_X	e_X	e_X	e_X	e_X	e_X	a_G	e_X	a_G	e_X	a_G
Face ($\sigma_{noise} = 0$)	0.055	0.056	0.037	0.033	0.032	0.068	0.031	0.041	0.054	0.023	0.037	0.38(3)	0.023	0.65(3)	**0.022**	0.65(3)
Face ($\sigma_{noise} \neq 0$)	0.056	0.057	0.044	0.040	0.035	0.076	0.050	0.048	0.059	0.081	0.056	0.35(3)	**0.034**	0.63(3)	**0.034**	0.63(3)

TABLE 3

Evaluation on a non-rigid face synthetic collection under noise-free and noisy annotations. The table reports the 3D reconstruction error e_X for the following SfM baselines: TK [54] and MC [44]; and the NRSfM baselines: CSF [32], KSTA [33], SPM [22], EM-PND [39], TUS [64], GBNR [27], CNR [40], DNRSM [36] and C3DPO [47]; and our MUS and MUS2 algorithms. In all cases, we consider full 2D annotations. Again, for our approaches and LRR [43], we also report the grouping accuracies a_G, and the number of object groups we estimate in parentheses.

to solve these problems. We would like to recall that our approach does not need manually tuning any subspace rank parameter, neither assigning which images belong to which object class. For all experiments, we set the coefficients in Eq. (4) to $\lambda = 0.03$ and $\gamma = 10$.

Table 2 summarizes the reconstruction errors for all methods and the object grouping accuracy of ours and LRR [43], considering both noise-free and noisy annotations. For the noisy case, we corrupt 2D detections with a zero mean Gaussian perturbation with standard deviation $\sigma_{noise} = 0.01 \max_{i,j,k} \{|d_{ijk}|\}$, where d_{ijk} represents the maximum distance of an image point to the centroid of all the points. Note that our approaches MUS and MUS2 consistently outperform the rest of competing techniques in terms of 3D reconstruction accuracy for both cases, reducing, for instance, the 3D error of other methods by large margins between the 14% and 422% for the noise-free case, or from 22% to 482% for noisy annotations. Focusing on our two algorithms, we observe that MUS2 produces more accurate solutions than MUS in both noise-free and noisy annotations. Note also that GBNR [27], CNR [40] and DNRSM [36] do not provide solutions for all collections, as the number of points is not sufficient for their formulation or no convergence is achieved. As it can be seen, deep-learning approaches do not provide the best solutions on this dataset, since requiring large amounts of training data, i.e., images, to learn the deformation model. In addition, our approach also estimates the object grouping, as seen in the right-most column for both algorithms MUS and MUS2, resulting in very accurate segmentations compared to the LRR [43] solution. As it can be seen, our algorithms provide accurate segmentations even for noisy annotations, outperforming the results with clean data provided by LRR [43]. Figure 4 shows a few sample images for the *Bicycle* and *Chair* categories, and the 3D reconstructions we obtain by using our approaches.

We now evaluate our approaches on a synthetic collection of images of non-rigid *Faces* with 3D ground truth. This collection is provided by [10], and consists of 300 images and 63 2D feature annotations. As the deformation in this dataset is relatively small, in general terms, all methods provide accurate 3D reconstructions. In any case, our approaches provide again the most accurate solutions. Regarding object segmentation, in spite of using the faces of three different subjects, the shape configuration of every sub-collection is quite similar (note that it could be the same subject performing different expressions) and computing the class affinities only from 2D semantic points is a complex problem. Fortunately,

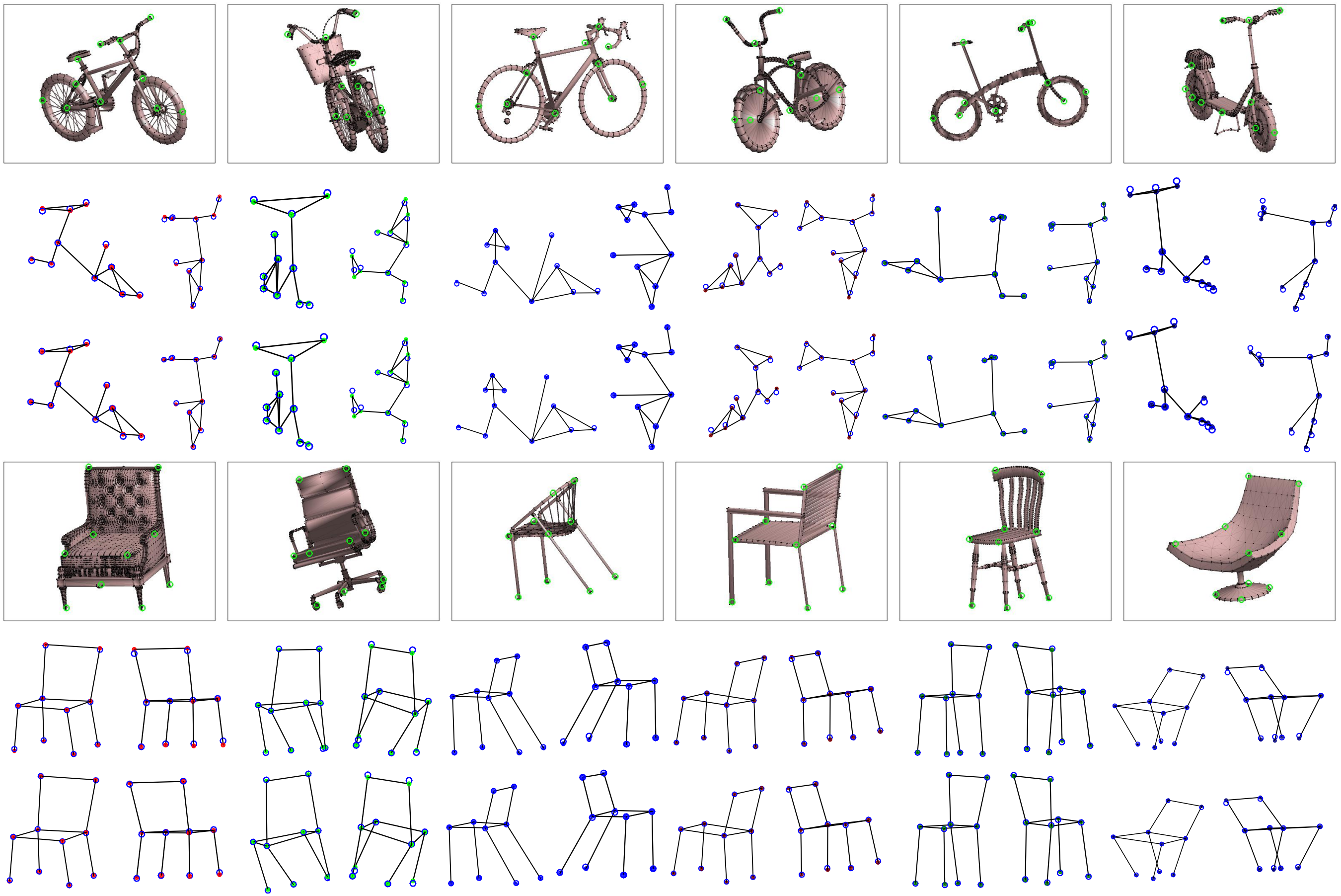

Fig. 4. **Bicycle and Chair collections.** The same information is shown for the two experiments. **Top:** Images $\{\#2, \#31, \#53, \#70, \#83, \#148\}$ and $\{\#21, \#37, \#49, \#63, \#93, \#139\}$ for the bicycle and chair collections, respectively. The semantic 2D point measurements fed to our model are represented by green circles. **Bottom:** Color-coded dots correspond to our 3D estimation (MUS and MUS2 solutions are displayed in two views, respectively) where every color represents a different object, and empty circles represent the 3D ground truth. To improve visualization, some links are also drawn.

our approach is the only that jointly provides both class and deformation grouping, obtaining a segmentation accuracy of $a_G = 0.65(3)$ for both algorithms. A summary of these results are reported in Table 3.

Finally, we show some failure cases of our algorithms, that are common to the rest of the literature. To this end, we use the *Bottle* (150/8) image collection of the PASCAL VOC dataset [26]. It is worth pointing out that our approaches only use 2D semantic points from an image collection. Considering that, for some revolution objects, if the distribution of the annotations is symmetric some rotations can become ambiguous (at least, that degree of freedom in the revolution axis), producing poor 3D reconstructions. Unfortunately, symmetric annotations are normally employed in these cases to maximize the volume of the object to be recovered. An example of this scenario is displayed in Fig. 5. Despite producing more accurate solutions than state-of-the-art approaches in terms of 3D reconstruction with $e_X = 0.51$, the solution is bad due to the motion ambiguities, that, though, it could be solved by re-annotating the dataset and enforcing non-symmetric semantic points.

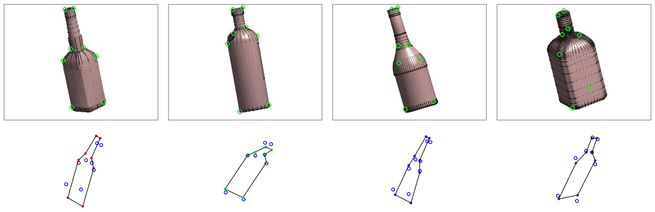

Fig. 5. **Bottle collection: a failure case. Top:** Images $\{\#5, \#28, \#71, \#86\}$ for the bottle collection together with the semantic 2D point annotations in green circles. **Bottom:** Color-coded dots correspond to our 3D estimation where every color represents a different object, and empty circles represent the 3D ground truth. As it can be seen, our algorithm fails in this case due to the motion ambiguities.

Ablation study. Each component is crucial for the proper performance of the full model, especially for non-rigid categories where all components $\{\mathbf{X}, \mathbf{Y}, \mathbf{Z}\}$ are considered (see section 4). However, for rigid categories not all components are strictly needed, and, as expected, the contribution of both $\mathbf{Y}$ and $\mathbf{Z}$ is reduced (ideally, both matrices should be null in this case). As we assume no prior information about the data to be processed, in real scenarios our full model is automatically adapted even for rigid categories, providing on average better solutions. The results for all collections with ground truth we consider in the paper are reported in table 4.

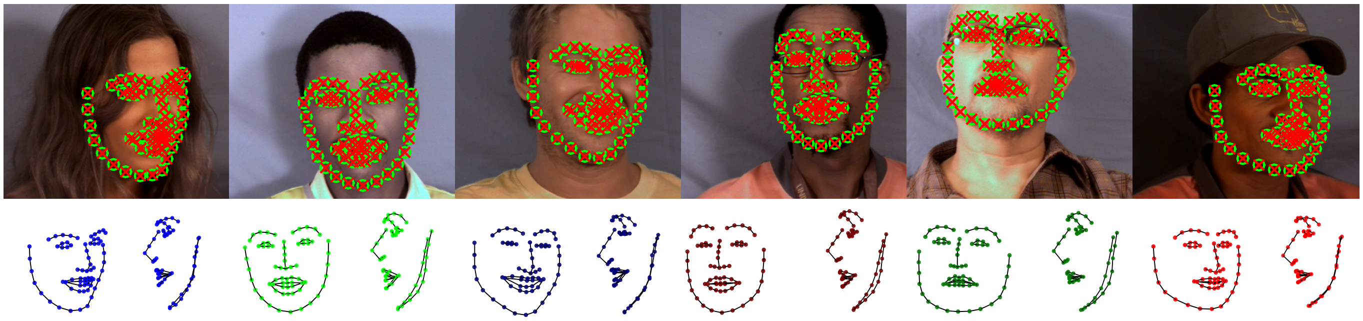

Fig. 6. **MUCT collection**. **Top:** Images #3, #26, #32, #46, #65 and #70 of the dataset. Input 2D detections and reprojected 3D shape are shown as green circles and red squares, respectively. **Bottom:** Camera viewpoint and side views of the estimated 3D shape. The colored dots indicate the object group index estimated by our approach MUS, i.e., a different person in the manifold of faces. Best viewed in color.

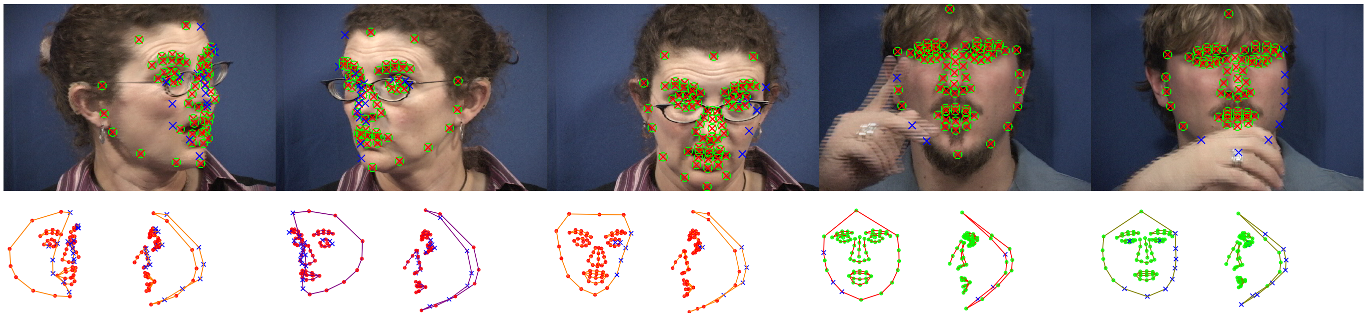

Fig. 7. **ASL collection. Top:** Images #29, #47, #100, #142 and #228 of the dataset. Input 2D detections and reprojected 3D shape are shown as green circles and red crosses, respectively. Blue crosses correspond to reconstructed (hallucinated) missing points. **Bottom:** Camera viewpoint and side views of the 3D reconstruction estimated by our algorithm MUS, where colored dots (red and green) indicate every human in the collection. The colored lines indicate a specific deformation group that was recovered. These estimated groups have a clear physical meaning and correspond to open/close mouth (shown in orange/magenta for the woman, and red/dark green for the man). 3D reconstructed missing points are represented by blue crosses. Best viewed in color.

Combination / Data Strategy:	**X**		**X + Y**		**X + Z**		Full	
	MUS	MUS2	MUS	MUS2	MUS	MUS2	MUS	MUS2
Aeroplane ($\sigma_{noise} = 0$)	0.259	**0.249**	0.260	0.260	0.260	0.253	0.261	0.253
Aeroplane ($\sigma_{noise} \neq 0$)	0.268	**0.255**	0.269	0.268	0.269	0.257	0.271	0.265
Bicycle ($\sigma_{noise} = 0$)	0.179	0.149	0.179	**0.143**	0.179	0.145	0.178	**0.143**
Bicycle ($\sigma_{noise} \neq 0$)	0.188	0.164	0.188	0.157	0.187	0.157	0.188	**0.156**
Bus ($\sigma_{noise} = 0$)	0.114	0.108	0.113	**0.106**	0.113	**0.106**	0.113	0.107
Bus ($\sigma_{noise} \neq 0$)	0.124	**0.115**	0.122	0.116	0.122	0.116	0.122	0.117
Car ($\sigma_{noise} = 0$)	0.077	**0.066**	0.078	0.068	0.078	0.068	0.078	0.069
Car ($\sigma_{noise} \neq 0$)	0.093	**0.084**	0.093	0.086	0.094	0.085	0.093	0.086
Chair ($\sigma_{noise} = 0$)	0.211	0.186	0.211	0.187	0.210	0.185	0.210	**0.184**
Chair ($\sigma_{noise} \neq 0$)	0.220	0.193	0.220	0.200	0.221	0.198	0.220	**0.192**
Diningtable ($\sigma_{noise} = 0$)	0.264	0.249	0.264	0.247	0.264	0.249	0.264	**0.246**
Diningtable ($\sigma_{noise} \neq 0$)	0.268	0.252	0.267	0.250	0.267	0.252	0.267	**0.241**
Motorbike ($\sigma_{noise} = 0$)	0.224	**0.200**	0.223	0.201	0.223	**0.200**	0.222	0.202
Motorbike ($\sigma_{noise} \neq 0$)	0.234	**0.215**	0.234	**0.215**	0.234	**0.215**	0.233	**0.215**
Sofa ($\sigma_{noise} = 0$)	0.168	0.142	0.167	**0.140**	0.167	**0.140**	0.167	0.141
Sofa ($\sigma_{noise} \neq 0$)	0.176	0.151	0.174	**0.150**	0.174	**0.150**	0.174	**0.150**
Face ($\sigma_{noise} = 0$)	0.028	0.026	0.024	0.023	0.024	0.024	0.023	**0.022**
Face ($\sigma_{noise} \neq 0$)	0.047	0.047	0.035	0.035	0.035	0.035	**0.034**	**0.034**

TABLE 4
Ablation study. The table reports the 3D reconstruction error e_X as a function of the effect of different components in the algorithms MUS and MUS2. Both noise-free $\sigma_{noise} = 0$ and noisy $\sigma_{noise} \neq 0$ observations are considered, as well as rigid and non-rigid categories.

6.2 Real Images

We next provide results on several real image collections either deforming linearly (faces) or highly non-linearly (animal motion). Since no ground truth is available for these datasets we only show qualitative evaluation.

The MUCT collection [45] is made of 72 images of faces of seven people, both men and women, of different ages and races, and under varying poses and expressions. The 2D annotations are obtained by using an off-the-shelf 2D active appearance model [21]. This model consists of 68 2D points, which are all visible in all frames. The results we provide in this dataset are shown in Fig. 6. Despite no quantitative estimates are available, the 3D reconstruction we obtain seems very realistic. We can, however, manually annotate the results of the object segmentation. Even though the 2D shapes are very similar (recall that object segmentation is computed based just on the 2D location of points) we obtain a segmentation accuracy $a_G = 0.68(7)$ for both algorithms.

In order to validate our approach against missing annotations, we process the ASL collection [33], consisting of 229 images of a man and a woman. The number of 2D feature points is 77, but some of them are not visible due to structured occlusions (by the hands or face self-rotation). In total, 14.43% of the points are missing. The 3D reconstruction results are shown in Fig. 7. Note that the inferred shapes seem to be very accurate, even when hallucinating the occluded points. In this case, the object segmentation is computed with no error, i.e., $a_G = 1.0(2)$. For this experiment, we also display the grouping in terms of type of deformation (colored lines in the 3D reconstruction of Fig. 7). These groups seem to have a clear physical meaning indicating face deformations with closed or open mouth.

We finally evaluate our approaches on two challenging

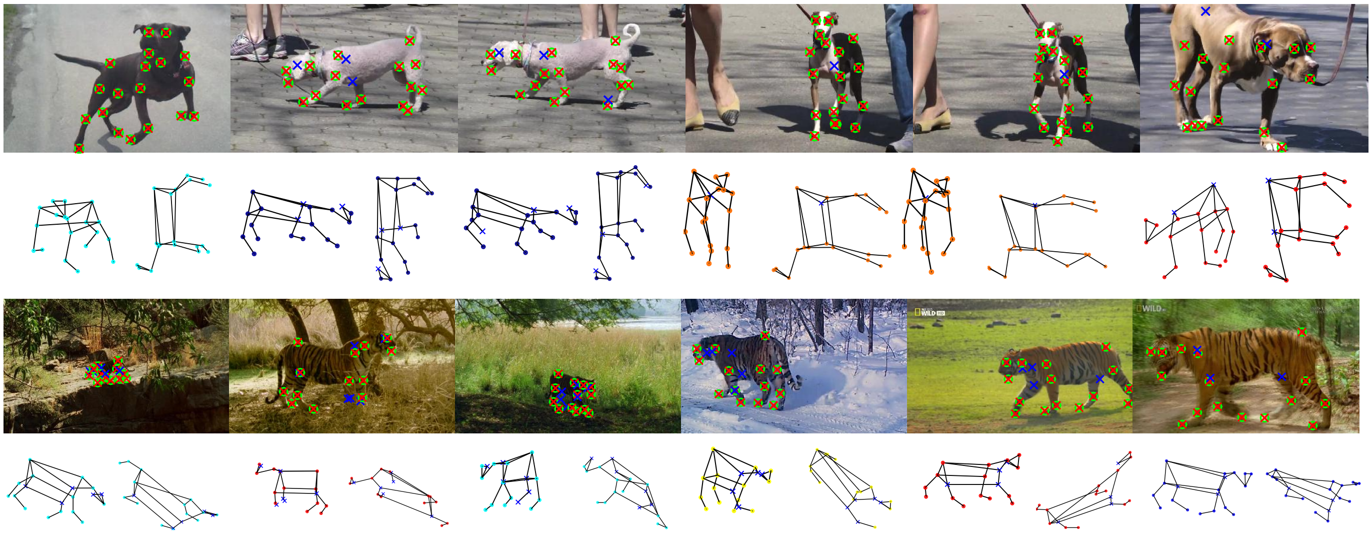

Fig. 8. **Dog and tiger collections**. The same information is shown for the two experiments. **Top:** Images #4, #14, #15, #24, #25 and #35 of the dog dataset, and #6, #9, #13, #18, #23 and #31 of the tiger one. Input 2D detections and reprojected 3D shape are shown as green circles and red crosses, respectively. **Bottom:** 3D reconstruction from a couple of novel point of views, where colored dots indicate the object group index estimated by our approach. In both cases, missing points are shown as blue crosses.

collections of animal images [24] composed each one of them of 19 semantic points, which were partially and manually annotated, i.e., not all points were visible or annotated in all images. In the first collection, we use 52 *Dog* images where 33 instances appear, and the 11.34% of the 2D input points are missing. In second one, we use 32 *Tiger* images where 4 instances appear[1], being the 25.00% of the 2D semantic points annotated as missing. The 3D reconstruction and grouping results are shown in Fig. 8 for dogs and tigers, using our algorithms MUS and MUS2, respectively. In both cases, we can observe as the 3D skeletons we obtain seem physically very plausible, even for the points that are not observed in the picture. Regarding grouping, despite being a hard task even for a human being, our method can obtain a realistic estimation, obtaining $a_G = 0.69(26)/0.68(4)$ for dog/tiger collections, respectively, and for both algorithms.

7 Conclusion

In this paper we have extended both SfM and NRSfM to a new scenario in which we can estimate 3D shape of either rigid or non-rigid categories from collections of RGB images. Considering only incomplete 2D point annotations per image, we present an approach that besides reconstructing 3D shape, it also recovers camera pose per image, as well as splits the collection of images into different objects and deformation primitives. For this purpose, we have introduced two algorithms that model object shape using multiple unions of subspaces, being able to render from rigid motion to highly non-rigid deformations. The model parameters are learned via an ALM scheme in a completely unsupervised and unified manner. We have experimentally evaluated our method on synthetic and real collections of images, of both rigid and non-rigid categories, outperforming state-of-the-art solutions in terms of 3D reconstruction and grouping by large margins. An interesting avenue for future research is to extend our formulation for sequential processing, allowing us the use of bigger datasets.

Acknowledgment: This work has been partially supported by the CSIC project R3OBJ 201850I099, the Spanish Ministry of Science and Innovation under project HuMoUR TIN2017-90086-R as well as the Salvador de Madariaga grant PRX19/00626.

References

[1] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski. Building Rome in a day. In *ICCV*, 2009.

[2] A. Agudo and F. Moreno-Noguer. Recovering pose and 3D deformable shape from multi-instance image ensembles. In *ACCV*, 2016.

[3] 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.

[4] A. Agudo and F. Moreno-Noguer. Deformable motion 3D reconstruction by union of regularized subspaces. In *ICIP*, 2018.

[5] A. Agudo and F. Moreno-Noguer. Force-based representation for non-rigid shape and elastic model estimation. *TPAMI*, 40(9):2137–2150, 2018.

[6] A. Agudo and F. Moreno-Noguer. Robust spatio-temporal clustering and reconstruction of multiple deformable bodies. *TPAMI*, 41(4):971–984, 2019.

[7] 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.

[8] A. Agudo, M. Pijoan, and F. Moreno-Noguer. Image collection pop-up: 3D reconstruction and clustering of rigid and non-rigid categories. In *CVPR*, 2018.

[9] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Trajectory space: A dual representation for nonrigid structure from motion. *TPAMI*, 33(7):1442–1456, 2011.

[10] I. Akhter, T. Simon, S. Khan, I. Matthews, and Y. Sheikh. Bilinear spatiotemporal basis models. *TOG*, 31(2):17:1–17:12, 2012.

[11] F. Bach, J. Mairal, and J. Ponce. Convex sparse matrix factorizations. *Technical report HAL-00345747*, 2008.

[1] Note that defining the exact number of tiger instances only from images is a complex task. As the dataset did not include ground truth in the number of instances, in this paper it was manually estimated and hence it is fully subjective.

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

[14] 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.

[15] J.F. Cai, E. Candes, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM JO*, 20(4):1956–1982, 2010.

[16] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *FCM*, 9(6):717, 2008.

[17] G. Cha, M. Lee, and S. Oh. Unsupervised 3D reconstruction networks. In *ICCV*, 2019.

[18] 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.

[19] Y. Chen, H. Xu, C. Caramanis, and S. Sanghavi. Robust matrix completion with corrupted columns. In *ICML*, 2011.

[20] A. Chhatkuli, D. Pizarro, T. Collins, and A. Bartoli. Inextensible non-rigid structure-from-motion by second-order cone programming. *TPAMI*, 40(10):2428–2441, 2018.

[21] T. Cootes, G. Edwards, and C. Taylor. Active appearance models. In *ECCV*, 1998.

[22] Y. Dai, H. Li, and M. He. A simple prior-free method for non-rigid structure from motion factorization. In *CVPR*, 2012.

[23] A. Del Bue, J. Xavier, L. Agapito, and M. Paladini. Bilinear modeling via augmented lagrange multipliers (BALM). *TPAMI*, 34(8):1496–1508, 2012.

[24] L. Del Pero, S. Ricco, R. Sukthankar, and V. Ferrari. Articulated motion discovery using pairs of trajectories. In *CVPR*, 2015.

[25] A. Edelman, T. Arias, and S. Smith. The geometry of algorithms with orthogonality constraints. *SIAM JMAA*, 20(20):303–353, 1998.

[26] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (VOC) challenge. *IJCV*, 88(2):303–338, 2010.

[27] K. Fragkiadaki, M. Salas, P. Arbeláez, and J. Malik. Grouping-based low-rank trajectory completion and 3D reconstruction. In *NIPS*, 2014.

[28] Y. Gao and A. L. Yuille. Symmetric non-rigid structure from motion for category-specific object structure estimation. In *ECCV*, 2016.

[29] R. Garg, A. Roussos, and L. Agapito. Dense variational reconstruction of non-rigid surfaces from monocular video. In *CVPR*, 2013.

[30] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Univ Pr, 1996.

[31] V. Golyanik and D. Stricker. Dense batch non-rigid structure from motion in a second. In *WACV*, 2017.

[32] 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.

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

[34] A. Kanazawa, S. Tulsiani, A. A. Efros, and J. Malik. Learning category-specific mesh reconstruction from image collections. In *ECCV*, 2018.

[35] A. Kar, S. Tulsiani, L. Carreira, and J. Malik. Category-specific object reconstruction from a single image. In *CVPR*, 2015.

[36] C. Kong and S. Lucey. Deep non-rigid structure from motion. In *ICCV*, 2019.

[37] C. Kong, R. Zhu, H. Kiani, and S. Lucey. Structure from category: A generic and prior-less approach. In *3DV*, 2016.

[38] 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.

[39] M. Lee, J. Cho, C. H. Choi, and S. Oh. Procrustean normal distribution for non-rigid structure from motion. In *CVPR*, 2013.

[40] M. Lee, J. Cho, and S. Oh. Consensus of non-rigid reconstructions. In *CVPR*, 2016.

[41] J. Lim, J. M. Frahm, and M. Pollefeys. Online environment mapping. In *CVPR*, 2011.

[42] 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.

[43] 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.

[44] M. Marques and J. Costeira. Optimal shape from estimation with missing and degenerate data. In *WMVC*, 2008.

[45] S. Milborrow, J. Morkel, and F. Nicolls. The MUCT Landmarked Face Database. *Pattern Recognition Association of South Africa*, 2010.

[46] R. Newcome and A. J. Davison. Live dense reconstruction with a single moving camera. In *CVPR*, 2010.

[47] D. Novotny, N. Ravi, B. Graham, N. Neverova, and A. Vedaldi. C3DPO: Canonical 3D pose networks for non-rigid structure from motion. In *ICCV*, 2019.

[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 with riemannian geometry solved in linear time. *TPAMI*, 40(10):2442–2454, 2018.

[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] B. Recht, M. Fazel, and P. A Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[52] C. Russell, R. Yu, and L. Agapito. Video pop-up: Monocular 3D reconstruction of dynamic scenes. In *ECCV*, 2014.

[53] A. Shaji and S. Chandran. Riemannian manifold optimisation for non-rigid structure from motion. In *CVPRW*, 2008.

[54] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization approach. *IJCV*, 9(2):137–154, 1992.

[55] L. Torresani, A. Hertzmann, and C. Bregler. Nonrigid structure-from-motion: estimating shape and motion with hierarchical priors. *TPAMI*, 30(5):878–892, 2008.

[56] Q. Wang, X. Zhou, and K. Daniilidis. Multi-image semantic matching by mining consistent features. In *CVPR*, 2018.

[57] J. Xiao, J. Chai, and T. Kanade. A closed-form solution to non-rigid shape and motion. *IJCV*, 67(2):233–246, 2006.

[58] X. Xu and E. Dunn. Discrete Laplace operator estimation for dynamic 3D reconstruction. In *ICCV*, 2019.

[59] J. Yang, W. Yin, Y. Zhang, and Y. Wang. A fast algorithm for edge-preserving variational multichannel image restoration. *SIAM JIS*, 2(2):569–592, 2009.

[60] Q. Yao, J. T. Kwok, T. Wang, and T.Y. Liu. Large-scale low-rank matrix learning with nonconvex regularizers. *TPAMI*, 41(11):2628–2643, 2019.

[61] R. Zass and A. Shashua. Doubly stochastic normalization for spectral clustering. In *NIPS*, 2006.

[62] Z. Zhang and W. S. Lee. Deep graphical feature learning for the feature matching problem. In *ICCV*, 2019.

[63] X. Zhou, M. Zhu, and K. Daniilidis. Multi-image matching via fast alternating minimization. In *ICCV*, 2015.

[64] Y. Zhu, D. Huang, F. De La Torre, and S. Lucey. Complex non-rigid motion 3D reconstruction by union of subspaces. In *CVPR*, 2014.

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 with the 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, and at Université de Bordeaux in 2018 and 2019. 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 Consejo Superior de Investigaciones Científicas (CSIC) in 2017. His research interests include non-rigid structure from motion, machine learning, and deformation analysis to medical and robotics applications. He is a recipient of the 2018 Best Paper Award Honorable Mention from the European Conference on Computer Vision.