Symmetric Non-rigid Structure from Motion for Category-Specific Object Structure Estimation

Yuan Gao[1(✉)] and Alan L. Yuille[2,3]

[1] City University of Hong Kong, Kowloon Tong, Hong Kong
Ethan.Y.Gao@gmail.com
[2] UCLA, Los Angeles, USA
Alan.L.Yuille@gmail.com
[3] John Hopkins University, Baltimore, USA

Abstract. Many objects, especially these made by humans, are symmetric, *e.g.* cars and aeroplanes. This paper addresses the estimation of 3D structures of symmetric objects from multiple images of the same object category, *e.g.* different cars, seen from various viewpoints. We assume that the deformation between different instances from the same object category is non-rigid and symmetric. In this paper, we extend two leading non-rigid structure from motion (SfM) algorithms to exploit symmetry constraints. We model the both methods as energy minimization, in which we also recover the missing observations caused by occlusions. In particularly, we show that by rotating the coordinate system, the energy can be decoupled into two independent terms, which still exploit symmetry, to apply matrix factorization separately on each of them for initialization. The results on the Pascal3D+ dataset show that our methods significantly improve performance over baseline methods.

Keywords: Symmetry · Non-rigid structure from motion

1 Introduction

3D structure reconstruction is a major task in computer vision. Structure from motion (SfM) method, which aims at estimating the 3D structure by the 2D annotated keypoints from 2D image sequences, has been proposed for rigid objects [1], and was later extended to non-rigidity [2–14]. Many man-made objects have symmetric structures [15,16]. Motivated by this, symmetry has been studied extensively in the past decades [16–22]. However, this information

This work was been done when Yuan Gao was a visiting student in UCLA.

Electronic supplementary material The online version of this chapter (doi:10.1007/978-3-319-46475-6_26) contains supplementary material, which is available to authorized users.

B. Leibe et al. (Eds.): ECCV 2016, Part II, LNCS 9906, pp. 408–424, 2016.
DOI: 10.1007/978-3-319-46475-6_26

has not been exploited in recent works on 3D object reconstruction [23,24], nor used in standard non-rigid structure from motion (NRSfM) algorithms [3–10,14].

The goal of this paper is to investigate how symmetry can improve NRSfM. Inspired by recent works [23,24], we are interested in estimating the 3D structure of objects, such as cars, airplanes, *etc.* This differs from the classic SfM problem because our input are images of several different object instances from the same category (*e.g.* different cars), instead of sequential images of the same object undergoing motion. In other words, our goal is to estimate the 3D structures of objects from the same class, given intra-class instances from various viewpoints. Specifically, the Pascal3D+ keypoint annotations on different objects from the same category are used as input to our method, where the symmetric keypoint pairs can also be easily inferred. In this paper, non-rigidity means the deformation between the objects from same category can be non-rigid, *e.g.* between sedan and SUV cars, but the objects themselves are rigid and symmetric.

By exploiting symmetry, we propose two symmetric NRSfM methods. By assuming that the 3D structure can be represented by a linear combination of basis functions (the coefficients vary for different objects): one method is an extension of [5] which is based on an EM approach with a Gaussian prior on the coefficients of the deformation bases, named Sym-EM-PPCA; the other method, *i.e.* Sym-PriorFree, is an extension of [9,10], which is a direct matrix factorization method without prior knowledge. For fair comparison, we use the same projection models and other assumptions used in [5] and [9,10].

More specifically, our Sym-EM-PPCA method, following [5], assumes weak perspective projection (*i.e.* the orthographic projection plus scale). We group the keypoints into symmetric keypoint pairs. We assume that the 3D structure is also symmetric and consists of a mean shape (of that category) and a linear combination of the symmetric deformation bases. As in [5], we put a Gaussian prior on the coefficient of the deformation bases. This is intended partly to regularize the problem and partly to deal an apparent ambiguity in non-rigid structure from motion. But recent work [25] showed that this is a "gauge freedom" which does not affect the estimation of 3D structure, so the prior is not really needed.

Our Sym-PriorFree method is based on prior free non-rigid SfM algorithms [9,10], which build on the insights in [25]. We formulate the problem of estimating 3D structure and camera parameters in terms of minimizing an energy function, which exploits symmetry, and at the same time can be re-expressed as the sum of two independent energy functions. Each of these energy functions can be minimized separately by matrix factorization, similar to the methods in [9,10], and the ambiguities are resolved using orthonormality constraints on the viewpoint parameters. This extends work in a companion paper [26], which shows how symmetry can be used to improve rigid structure from motion methods [1].

Our main contributions are: (I) Sym-EM-PPCA, which imposes symmetric constraints on both 3D structure and deformation bases. Sym-Rigid-SfM (see our companion paper [26]) is used to initialize Sym-EM-PPCA with hard symmetric constraints on the 3D structure. (II) Sym-PriorFree, which extends the matrix factorization methods of [9,10], to initialize a coordinate descent algorithm.

In this paper, we group keypoints into symmetric keypoint pairs, and use a superscript † to denote symmetry, *i.e.* Y and $Y^\dagger$ are the 2D symmetric keypoint pairs. The paper is organized as follows: firstly, we review related works in Sect. 2. In Sect. 3, the ambiguities in non-rigid SfM are discussed. Then we present the Sym-EM-PPCA algorithm and Sym-PriorFree algorithm in Sect. 4. After that, following the experimental settings in [24], we evaluated our methods on the Pascal3D+ dataset [27] in Sect. 5. Section 5 also includes diagnostic results on the noisy 2D annotations to show that our methods are robust to imperfect symmetric annotations. Finally, we give our conclusions in Sect. 6.

2 Related Works

There is a long history of using symmetry as a cue for computer vision tasks. For example, symmetry has been used in depth recovery [17,18,20] as well as recognizing symmetric objects [19]. Several geometric clues, including symmetry, planarity, orthogonality and parallelism have been taken into account for 3D scene reconstruction [28,29], in which the author used pre-computed camera rotation matrix by vanishing point [30]. Recently, symmetry has been applied in more areas such as 3D mesh reconstruction with occlusion [21], and scene reconstruction [16]. For 3D keypoints reconstruction, symmetry, incorporated with planarity and compactness prior, has also been studied in [22].

SfM has also been studied extensively in the past decades, ever since the seminal work on rigid SfM [1,31]. Bregler *et al.* extended this to the non-rigid case [32]. A Column Space Fitting (CSF) method was proposed for rank-r matrix factorization (MF) for SfM with smooth time-trajectories assumption [7], which was later unified in a more general MF framework [33][1]. Early analysis of NRSfM showed that there were ambiguities in 3D structure reconstruction [4]. This lead to studies which assumed priors on the NR deformations [4–7,34,35]. But it was then shown that these ambiguities did not affect the final estimate of 3D structure, *i.e.* all legitimate solutions lying in the same subspace (despite under-constrained) give the same solutions for the 3D structure [25]. This facilitated the invention of prior free matrix factorization method for NRSfM [9,10]. Recently SfM methods have also been used for category-specific object reconstruction, *e.g.* estimating the shape of different *cars* under various viewing conditions [23, 24], but the symmetry cues was not exploited. Note that repetition patterns have recently been incorporated into SfM for urban facades reconstruction [36], but this mainly focused on repetition detection and registration. Finally, in a companion paper [26], we exploited symmetry for rigid SfM.

3 The Ambiguities in Non-rigid SfM

This section reviews the intrinsic ambiguities in non-rigid SfM, *i.e.* (i) the ambiguities between the camera projection and the 3D structure, and (ii) the ambiguities between the deformation bases and their coefficients [25]. In the following

[1] However, the general framework in [33] cannot be used to SfM directly, because they did not constrain that all the keypoints have the same translation.

sections (*i.e.* in Remark 5), we will show the ambiguity between camera projection and 3D structure (*i.e.* originally the 3×3 matrix ambiguity as discussed below) can be decomposed into two types of ambiguities under the symmetric constraints, *i.e.* a scale ambiguity along the symmetry axis, and a 2×2 matrix ambiguity on the other two axes.

The key idea of non-rigid SfM is to represent the non-rigid deformations of objects in terms of a linear combination of bases:

$$\mathbf{Y} = \mathbf{RS} \quad \text{and} \quad \mathbf{S} = \mathbf{Vz}, \quad \mathbf{RR}^T = I, \tag{1}$$

where $\mathbf{Y}$ is the stacked 2D keypoints, $\mathbf{R}$ is the camera projection for the N images. $\mathbf{S}$ is the 3D structure which is modeled by the linear combination of the stacked deformation bases $\mathbf{V}$, and $\mathbf{z}$ is the coefficient.

Firstly, as is well known, there are ambiguities between the projection $\mathbf{R}$ and the 3D structure $\mathbf{S}$ in the matrix factorization, *i.e.* let $\mathbf{A}_1$ be an invertible matrix, then $\mathbf{R} \leftarrow \mathbf{RA}_1$ and $\mathbf{S} \leftarrow \mathbf{A}_1^{-1}\mathbf{S}$ will not change the value of $\mathbf{RS}$. These ambiguities can be solved by imposing orthogonality constraints on the camera parameters $\mathbf{RR}^T = I$ up to a fixed rotation, which is a "gauge freedom" [37] corresponding to a choice of coordinate system.

In addition, there are other ambiguities between the coefficients $\mathbf{z}$ and the deformation bases $\mathbf{V}$ [4]. Specifically, let $\mathbf{A}_2$ be another invertible matrix, and let $\mathbf{w}$ lie in the null space of the projected deformation bases $\mathbf{RV}$, then $\mathbf{z} \leftarrow \mathbf{A}_2\mathbf{z}$ and $\mathbf{V} \leftarrow \mathbf{VA}_2^{-1}$, or $\mathbf{z} \leftarrow \mathbf{z} + \alpha\mathbf{w}$ will not change the value of $\mathbf{RVz}$. This motivated Bregler *et al.* to impose a Gaussian prior on the coefficient $\mathbf{z}$ in order to eliminate the ambiguities. Recently, it was proved in [25] that these ambiguities are also "fake", *i.e.* they do not affect the estimate of the 3D structure. This proof facilitated prior-free matrix factorization methods for non-rigid SfM [9,10].

4 Symmetric Non-rigid Structure from Motion

In this paper we extend non-rigid SfM methods by requiring that the 3D structure is symmetric. We assume the deformations are non-rigid and also symmetric[2]. We propose two symmetric non-rigid SfM models. One is the extension of the iterative EM-PPCA model with a prior on the deformation coefficients [5], and the other extends the prior-free matrix factorization model [9,10].

For simplicity of derivation, we focus on estimating the 3D structure and camera parameters. In practice, there are occluded keypoints in almost all images in the Pascal3D+ dataset. But we use standard ways to deal with them, such as initializing them ignoring symmetry by rank 3 recovery using the first 3 largest singular value, then treating them as missing data to be estimated by EM or coordinate descent algorithms. In our companion paper [26]), we gave details of these methods for the simpler case of rigid structure from motion.

[2] We assume symmetric deformations because our problem involves deformations from one symmetric object to another. But it also can be extended to non-symmetric deformations straightforwardly.

Note that we use slightly different camera models for Sym-EM-PPCA (weak perspective projection) and Sym-PriorFree (orthographic projection). This is to stay consistent with the non-symmetric methods which we compare with, namely [5] and [9,10]. Similarly, we treat translation differently by either centeralizing the data or treating it as a variable to be estimated, as appropriate. We will discuss this further when presenting the Sym-PriorFree method.

4.1 The Symmetric EM-PPCA Model

In EM-PPCA [5], Bregler *et al.* assume that the 3D structure is represented by a mean structure $\bar{S}$ plus a non-rigid deformation. Suppose there are P keypoints on the structure, the non-rigid model of EM-PPCA is:

$$\mathbb{Y}_n = G_n(\bar{\mathbb{S}} + \mathbf{V}z_n) + \mathbb{T}_n + N_n, \tag{2}$$

where $\mathbb{Y}_n \in \mathbb{R}^{2P\times 1}, \bar{\mathbb{S}} \in \mathbb{R}^{3P\times 1}$, and $\mathbb{T}_n \in \mathbb{R}^{2P\times 1}$ are the stacked vectors of 2D keypoints, 3D mean structure and translations. $G_n = I_P \otimes c_n R_n$, in which c_n is the scale parameter for weak perspective projection, $\mathbf{V} = [\mathbb{V}_1, ..., \mathbb{V}_K] \in \mathbb{R}^{3P\times K}$ is the grouped K deformation bases, $z_n \in \mathbb{R}^{K\times 1}$ is the coefficient of the K bases, and N_n is the Gaussian noise $N_n \sim \mathcal{N}(0, \sigma^2 I)$.

Extending Eq. (2) to our symmetry problem in which there are P keypoint pairs $\mathbb{Y}_n$ and $\mathbb{Y}_n^\dagger$, we have:

$$\mathbb{Y}_n = G_n(\bar{\mathbb{S}} + \mathbf{V}z_n) + \mathbb{T}_n + N_n, \quad \mathbb{Y}_n^\dagger = G_n(\bar{\mathbb{S}}^\dagger + \mathbf{V}^\dagger z_n) + \mathbb{T}_n + N_n. \tag{3}$$

Assuming that the object is symmetric along the x-axis, the relationship between $\bar{\mathbb{S}}$ and $\bar{\mathbb{S}}^\dagger$, $\mathbf{V}$ and $\mathbf{V}^\dagger$ are:

$$\bar{\mathbb{S}}^\dagger = \mathcal{A}_P\bar{\mathbb{S}}, \quad \mathbf{V}^\dagger = \mathcal{A}_P\mathbf{V}, \tag{4}$$

where $\mathcal{A}_P = I_P \otimes \mathcal{A}$, $\mathcal{A} = \text{diag}([-1, 1, 1])$ is a matrix operator which negates the first row, and $I_P \in \mathbb{R}^{P\times P}$ is an identity matrix. Thus, we have[3]:

$$\begin{aligned} P(\mathbb{Y}_n|z_n, G_n, \bar{\mathbb{S}}, \mathbf{V}, \mathbb{T}) &= \mathcal{N}(G_n(\bar{\mathbb{S}} + \mathbf{V}z_n) + \mathbb{T}_n, \sigma^2 I) \\ P(\mathbb{Y}_n^\dagger|z_n, G_n, \bar{\mathbb{S}}, \mathbf{V}^\dagger, \mathbb{T}) &= \mathcal{N}(G_n(\mathcal{A}_P\bar{\mathbb{S}} + \mathbf{V}^\dagger z_n) + \mathbb{T}_n, \sigma^2 I) \end{aligned} \tag{5}$$

Following Bregler *et al.* [5], we introduce a prior $P(z_n)$ on the coefficient variable z_n. This prior is a zero mean unit variance Gaussian. It is used for (partly) regularizing the inference task but also for dealing with the ambiguities between basis coefficients z_n and bases $\mathbf{V}$, as mentioned above (when [5] was published it was not realized that these are "gauge freedom"). This enables us to treat z_n as the hidden variable and use EM algorithm to estimate the structure and camera viewpoint parameters. The formulation of the problem, in terms of Gaussian distributions (or, more technically, the use of conjugate priors) means that both steps of the EM algorithm are straightforward to implement.

[3] We set hard constraints on $\bar{\mathbb{S}}$ and $\bar{\mathbb{S}}^\dagger$, *i.e.* replace $\bar{\mathbb{S}}^\dagger$ by $\mathcal{A}_P\bar{\mathbb{S}}$ in Eq. (5), because it can be guaranteed by the Sym-RSfM initialization in our companion paper [26]. While the initialization on $\mathbf{V}$ and $\mathbf{V}^\dagger$ by PCA cannot guarantee such a desirable property, thus a Language multiplier term is used for the constraint on $\mathbf{V}$ and $\mathbf{V}^\dagger$ in Eq. (9).

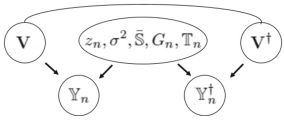

Fig. 1. The graphical model of the variables and parameters.

Remark 1. Our Sym-EM-PPCA method is a natural extension of the method in [5] to maximize the marginal probability $P(\mathbb{Y}_n, \mathbb{Y}_n^\dagger | G_n, \bar{\mathbb{S}}, \mathbf{V}, \mathbf{V}^\dagger, \mathbb{T})$ with a Gaussian prior on z_n and a Language multiplier term (*i.e.* a regularization term) on $\mathbf{V}, \mathbf{V}^\dagger$. This can be solved by *general EM* algorithm [38], where both the **E** and **M** steps take simple forms because the underlying probability distributions are Gaussians (due to conjugate Gaussian prior).

E-Step: This step is to get the statistics of z_n from its posterior. Let the prior on z_n be $P(z_n) = \mathcal{N}(0, I)$ as in [5]. Then, we have $P(z_n)$, $P(\mathbb{Y}_n | z_n; \sigma^2, \bar{\mathbb{S}}, \mathbf{V}, G_n, \mathbb{T}_n)$ and $P(\mathbb{Y}_n^\dagger | z_n; \sigma^2, \bar{\mathbb{S}}, \mathbf{V}^\dagger, G_n, \mathbb{T}_n)$, which do not provide the complete posterior distribution directly. Fortunately, the conditional dependence of the variables shown in Fig. 1 (graphical model) implies that the posterior of z_n can be calculated by:

$$\begin{aligned}
&P(z_n | \mathbb{Y}_n, \mathbb{Y}_n^\dagger; \sigma^2, \bar{\mathbb{S}}, \mathbf{V}, \mathbf{V}^\dagger, G_n, \mathbb{T}_n) \\
\sim &P(z_n, \mathbb{Y}_n, \mathbb{Y}_n^\dagger | \sigma^2, \bar{\mathbb{S}}, \mathbf{V}, \mathbf{V}^\dagger, G_n, \mathbb{T}_n) \\
= &P(\mathbb{Y}_n | z_n; \sigma^2, \bar{\mathbb{S}}, \mathbf{V}, G_n, \mathbb{T}_n) P(\mathbb{Y}_n^\dagger | z_n; \sigma^2, \bar{\mathbb{S}}, \mathbf{V}^\dagger, G_n, \mathbb{T}_n) P(z_n) \\
= &\mathcal{N}(z_n | \mu_n, \Sigma_n)
\end{aligned} \tag{6}$$

The last equation of Eq. (6) is obtained by the fact that the prior and the conditional distributions of z_n are all Gaussians (conjugate prior). Then the first and second order statistics of z_n can be obtained as:

$$\mu_n = \gamma \left\{ \mathbf{V}^T G_n^T (\mathbb{Y}_n - G_n \bar{\mathbb{S}} - \mathbb{T}_n) + \mathbf{V}^{\dagger T} G_n^T (\mathbb{Y}_n^\dagger - G_n \mathcal{A}_P \bar{\mathbb{S}} - \mathbb{T}_n) \right\} \tag{7}$$

$$\phi_n = \sigma^2 \gamma^{-1} + \mu_n \mu_n^T \tag{8}$$

where $\gamma = (\mathbf{V}^T G_n^T G_n \mathbf{V} + \mathbf{V}^{\dagger T} G_n^T G_n \mathbf{V}^\dagger + \sigma^2 I)^{-1}$.

M-Step: This is to maximize the joint likelihood which is similar to the coordinate descent in Sym-RSfM (in a companion paper [26]) and that in Sym-PriorFree method in the later sections. The complete log-likelihood $Q(\theta)$ is:

$$\begin{aligned}
\mathcal{Q}(\theta) &= -\sum_n \ln P(\mathbb{Y}_n, \mathbb{Y}_n^\dagger | z_n; G_n, \bar{\mathbb{S}}, \mathbf{V}, \mathbf{V}^\dagger, \mathbb{T}) + \lambda ||\mathbf{V}^\dagger - \mathcal{A}_P \mathbf{V}||^2 \\
&= -\sum_n \left(\ln P(\mathbb{Y}_n | z_n; G_n, \bar{\mathbb{S}}, \mathbf{V}, \mathbb{T}) + \ln P(\mathbb{Y}_n^\dagger | z_n; G_n, \bar{\mathbb{S}}, \mathbf{V}^\dagger, \mathbb{T}) \right) + \lambda ||\mathbf{V}^\dagger - \mathcal{A}_P \mathbf{V}||^2 \\
&\text{s. t.} \qquad R_n R_n^T = I, \qquad \text{where } \theta = \{G_n, \bar{\mathbb{S}}, \mathbf{V}, \mathbf{V}^\dagger, \mathbb{T}_n, \sigma^2\}.
\end{aligned} \tag{9}$$

The maximization of Eq. (9) is straightforward, *i.e.* taking the derivative of each unknown parameter in θ and equating it to 0. The update rule of each

parameter is very similar to the original EM-PPCA [5] (except $\bar{\mathbb{S}}, \mathbf{V}, \mathbf{V}^\dagger$ should be updated jointly), which we put in supplementary material.

Initialization. $\mathbf{V}$ and $\mathbf{V}^\dagger$ are initialized by the PCA on the residual of the 2D keypoints minus their rigid projections iteratively. Other variables (including the rigid projections) are initialized by Sym-RSfM [26]. Specifically, R_n, $\bar{S}$ and the occluded points $Y_{n,p}, Y^\dagger_{n,p}$ can be initialized directly from Sym-RSfM, c_n is initialized as 1, t_n is initialized by $t_n = \sum_p (Y_{n,p} - R_n \bar{S}_p + Y^\dagger_{n,p} - R_n \mathcal{A} \bar{S}_p)$.

4.2 The Symmetric Prior-Free Matrix Factorization Model

In the Prior-Free NRSfM [9,10], Dai *et al.* also used the linear combination of several deformations bases to represent the non-rigid deformation. But, unlike EM-PPCA [5], Dai *et al.* estimated the non-rigid structure directly without using the mean structure and the prior on the coefficients. We make the same assumptions so that we can directly compare with them.

Assume that $Y_n \in \mathbb{R}^{2\times P}$ are the P keypoints for image n, then we have:

$$\begin{aligned} Y_n &= R_n S_n = [z_{n1} R_n, ..., z_{nK} R_n][\mathbf{V}_1, ..., \mathbf{V}_K]^T = \mathit{\Pi}_n \mathbf{V}, \\ Y^\dagger_n &= R_n S^\dagger_n = [z_{n1} R_n, ..., z_{nK} R_n][\mathbf{V}^\dagger_1, ..., \mathbf{V}^\dagger_K]^T = \mathit{\Pi}_n \mathbf{V}^\dagger, \end{aligned} \tag{10}$$

where $\mathbf{z}_n = [z_{n1}, ..., z_{nK}] \in \mathbb{R}^{1\times K}$, $\mathit{\Pi}_n = R_n(\mathbf{z}_n \otimes I_3) \in \mathbb{R}^{2\times 3K}$, and $\mathbf{V} = [\mathbf{V}_1^T, ..., \mathbf{V}_K^T]^T \in \mathbb{R}^{3K\times P}$.

Without loss of generality, we assume that the symmetry is across the x-axis: $S_n = \mathcal{A} S^\dagger_n$, where $\mathcal{A} = \text{diag}[-1, 1, 1]^T$ is a matrix operator negating the first row of S_n. Then the first two terms in Eq. (10) give us the energy function (or the likelihood) to estimate the unknown R_n, S_n and recover the missing data by *coordinate descent* on:

$$\begin{aligned} &\mathcal{Q}(R_n, S_n, \{Y_{n,p}, (n,p) \in IVS\}, \{Y^\dagger_{n,p}, (n,p) \in IVS^\dagger\}) \\ &= \sum_{(n,p)\in VS} ||Y_{n,p} - R_n S_{n,p}||_2^2 + \sum_{(n,p)\in VS^\dagger} ||Y^\dagger_{n,p} - R_n \mathcal{A} S_{n,p}||_2^2 + \\ &\quad \sum_{(n,p)\in IVS} ||Y_{n,p} - R_n S_{n,p}||_2^2 + \sum_{(n,p)\in IVS^\dagger} ||Y^\dagger_{n,p} - R_n \mathcal{A} S_{n,p}||_2^2, \end{aligned} \tag{11}$$

where VS and IVS are the index sets of the *visible* and *invisible* keypoints, respectively. $Y_{n,p}$ and $S_{n,p}$ are the 2D and 3D p'th keypoints of the n'th image. We treat the $\{Y_{n,p}, (n,p) \in IVS\}, \{Y^\dagger_{n,p}, (n,p) \in IVS^\dagger\}$ as missing/hidden variables to be estimated.

Remark 2. It is straightforward to minimize Eq. (11) by *coordinate descent*. The missing points can be initialized simply by rank 3 recovery (*i.e.* by the reconstruction using the first 3 largest singular value) ignoring the symmetry property and non-rigidity. But it is much harder to get good initializations for the R_n and S_n. In the following, we will describe how we get good initializations for each R_n and S_n exploiting symmetry after the missing points have been initialized.

Let $\mathbf{Y}$ is the stacked keypoints of N images, $\mathbf{Y} = [Y_1^T, ..., Y_N^T]^T \in \mathbb{R}^{2N\times P}$, the model is represented by:

$$\mathbf{Y} = \mathbf{RS} = \begin{bmatrix} R_1 S_1 \\ \vdots \\ R_N S_N \end{bmatrix} = \begin{bmatrix} z_{11}R_1, & ..., & z_{1K}R_1 \\ \vdots & \ddots & \vdots \\ z_{N1}R_N, & ..., & z_{NK}R_N \end{bmatrix} \begin{bmatrix} \mathbf{V}_1 \\ \vdots \\ \mathbf{V}_K \end{bmatrix} = \mathbf{\Pi V}, \tag{12}$$

where $\mathbf{R} = \text{blkdiag}([R_1, ..., R_N]) \in \mathbb{R}^{2N\times 3N}$ are the stacked camera projection matrices, in which blkdiag denotes block diagonal. $\mathbf{S} = [S_1^T, ..., S_N^T]^T \in \mathbb{R}^{3N\times P}$ are the stacked 3D structures. $\mathbf{\Pi} = \mathbf{R}(\mathbf{z} \otimes I_3) \in \mathbb{R}^{2N\times 3K}$, where $\mathbf{z} \in \mathbb{R}^{N\times K}$ are the stacked coefficients. Similar equations apply to $\mathbf{Y}^\dagger$.

Note that $\mathbf{R} \in \mathbb{R}^{2N\times 3N}$, $\mathbf{V} \in \mathbb{R}^{3K\times P}$ are stacked differently than how they were stacked for the Sym-EM-PPCA method (*i.e.* $\mathbf{R} \in \mathbb{R}^{2N\times 3}$, $\mathbf{V} \in \mathbb{R}^{3P\times K}$). It is because now we have N different S_n's (*i.e.* $\mathbf{S} \in \mathbb{R}^{3N\times P}$), while there is only one $\bar{S}$ in the Sym-EM-PPCA method.

In the following, we assume the deformation bases are symmetric, which ensures that the non-rigid structures are symmetric (*e.g.* the deformation from *sedan* to *truck* is non-rigid and symmetric since *sedan* and *truck* are both symmetric). This yields an energy function:

$$\begin{aligned} \mathcal{Q}(\mathbf{R}, \mathbf{S}) =& ||\mathbf{Y} - \mathbf{RS}||_2^2 + ||\mathbf{Y}^\dagger - \mathbf{R}\mathcal{A}_N\mathbf{S}^\dagger||_2^2 \\ =& ||\mathbf{Y} - \mathbf{\Pi V}||_2^2 + ||\mathbf{Y}^\dagger - \mathbf{\Pi}\mathcal{A}_K\mathbf{V}^\dagger||_2^2, \end{aligned} \tag{13}$$

where $\mathcal{A}_N = I_N \otimes \mathcal{A}, \mathcal{A}_K = I_K \otimes \mathcal{A}$, and $\mathcal{A} = \text{diag}([-1, 1, 1])$.

Remark 3. Note that we cannot use the first equation of Eq. (13) to solve $\mathbf{R}, \mathbf{S}$ directly (even if not exploiting symmetry), because $\mathbf{Y}$ and $\mathbf{Y}^\dagger$ are of rank $min\{2N, 3K, P\}$ but estimating $\mathbf{R}, \mathbf{S}$ directly by SVD on $\mathbf{Y}$ and/or $\mathbf{Y}^\dagger$ requires rank $3N$ matrix factorization. Hence we focus on the last equation of Eq. (13) to get the initialization of $\mathbf{\Pi}, \mathbf{V}$ firstly. Then, $\mathbf{R}, \mathbf{S}$ can be updated by coordinate descent on the first equation of Eq. (13) under *orthogonality constraints* on $\mathbf{R}$ and *low-rank* constraint on $\mathbf{S}$.

Observe that the last equation of Eq. (13) cannot be optimized directly by SVD either, because they consist of two terms which are not independent. In other words, the matrix factorizations of $\mathbf{Y}$ and $\mathbf{Y}^\dagger$ do not give consistent estimations of $\mathbf{\Pi}$ and $\mathbf{V}$. Instead, we now discuss how to estimate $\mathbf{\Pi}$ and $\mathbf{V}$ by rotating the coordinate axes (to decouple the depended energy terms), performing matrix factorization, and using subspace intersection (to eliminate the ambiguities), which is an extension of the original prior-free method [9,10] and our companion Sym-RSfM [26].

We first rotate coordinate systems (of $\mathbf{Y}, \mathbf{Y}^\dagger$) to obtain decoupled equations:

$$\mathbf{L} = \frac{\mathbf{Y} - \mathbf{Y}^\dagger}{2} = \hat{\mathbf{\Pi}}^1\hat{\mathbf{V}}_x \qquad \mathbf{M} = \frac{\mathbf{Y} + \mathbf{Y}^\dagger}{2} = \hat{\mathbf{\Pi}}^2\hat{\mathbf{V}}_{yz}, \tag{14}$$

where the two righthand sides of the equation depend on different components of $\hat{\mathbf{\Pi}}, \hat{\mathbf{V}}$. More specifically, by discarding the all 0 rows of the bases, $\hat{\mathbf{\Pi}}^1 \in \mathbf{R}^{2N\times K}$, $\hat{\mathbf{\Pi}}^2 \in \mathbf{R}^{2N\times 2K}$, $\hat{\mathbf{V}}_x \in \mathbf{R}^{K\times P}$, $\hat{\mathbf{V}}_{yz} \in \mathbf{R}^{2K\times P}$.

This yield two independent energies to be minimized separately by SVD:

$$\mathcal{Q}(\mathbf{\Pi}, \mathbf{V}) = ||\mathbf{L} - \hat{\mathbf{\Pi}}^1 \hat{\mathbf{V}}_x||_2^2 + ||\mathbf{M} - \hat{\mathbf{\Pi}}^2 \hat{\mathbf{V}}_{yz}||_2^2 \tag{15}$$

Remark 4. We have formulated Sym-PriorFree as minimizing two energy terms in Eq. (15), which consists of independent variables. This implies that we can solve them by matrix factorization on each energy term separately, which gives solutions for $\mathbf{\Pi} = \mathbf{R}(\mathbf{z} \otimes I_3)$ and for the basis vectors $\mathbf{V}$ up to an ambiguity H. It will be discussed more explicitly in the following and we will show how to use orthogonality of the camera parameters to partially solve for H.

Solving Eq. (15) by matrix factorization gives us solutions up to a matrix ambiguity H. More precisely, there are ambiguity matrices H^1, H^2 between the true solutions $\mathbf{\Pi}^1, \mathbf{V}_x, \mathbf{\Pi}^2, \mathbf{V}_{yz}$ and the initial estimation output by matrix factorization $\hat{\mathbf{\Pi}}^1, \hat{\mathbf{V}}_x, \hat{\mathbf{\Pi}}^2, \hat{\mathbf{V}}_{yz}$:

$$\mathbf{L} = \mathbf{\Pi}^1 \mathbf{V}_x = \hat{\mathbf{\Pi}}^1 H^1 (H^1)^{-1} \hat{\mathbf{V}}_x \qquad \mathbf{M} = \mathbf{\Pi}^2 \mathbf{V}_{yz} = \hat{\mathbf{\Pi}}^2 H^2 (H^2)^{-1} \hat{\mathbf{V}}_{yz} \tag{16}$$

where $H^1 \in \mathbb{R}^{K \times K}$ and $H^2 \in \mathbb{R}^{2K \times 2K}$.

Now, the problem becomes to find H^1, H^2. Note that we have orthonormality constraints on each camera projection matrix R_n, which further impose constraints on $\mathit{\Pi}_n$. Thus, it can be used to partially estimate the ambiguity matrices H^1, H^2. Since the factorized matrix, *i.e.* $\mathbf{L}$ and $\mathbf{M}$, are the stacked 2D keypoints for all the images, thus H^1 and H^2 obtained from one image must satisfy the orthonormality constraints on other images, hence we use $\mathit{\Pi}_n \in \mathbb{R}^{2 \times 3K}$ (*i.e.* from image n) for our derivation.

Let $\hat{\mathit{\Pi}}_n = [\hat{\mathit{\Pi}}_n^1, \hat{\mathit{\Pi}}_n^2] = \begin{bmatrix} \hat{\pi}_n^{1,1:K}, \hat{\pi}_n^{1,K+1:3K} \\ \hat{\pi}_n^{2,1:K}, \hat{\pi}_n^{2,K+1:3K} \end{bmatrix}$, where $\hat{\pi}_n^{1,1:K}, \hat{\pi}_n^{2,1:K} \in \mathbb{R}^{1 \times K}$ are the first K columns of the first and second rows of $\hat{\mathit{\Pi}}_n$, and $\hat{\pi}_n^{1,K+1:3K}, \hat{\pi}_n^{2,K+1:3K} \in \mathbb{R}^{1 \times 2K}$ are the last $2K$ columns of the first and second rows of $\hat{\mathit{\Pi}}_n$, respectively. Thus, Eq. (16) implies:

$$L_n = \hat{\mathit{\Pi}}_n^1 H^1 (H^1)^{-1} \hat{\mathbf{V}}_x = \begin{bmatrix} r_n^{11} \\ r_n^{21} \end{bmatrix} \mathbf{z}_n \mathbf{V}_x, \tag{17}$$

$$M_n = \hat{\mathit{\Pi}}_n^2 H^2 (H^2)^{-1} \hat{\mathbf{V}}_{yz} = \begin{bmatrix} r_n^{1,2:3} \\ r_n^{2,2:3} \end{bmatrix} (\mathbf{z}_n \otimes I_2) \mathbf{V}_{yz}, \tag{18}$$

where $L_n, M_n \in \mathbb{R}^{2 \times P}$ are the n'th double-row of $\mathbf{L}, \mathbf{M}$. $[r_n^{11}, r_n^{12}]^T$ is the first column of the camera projection matrix of the n'th image R_n, and $[(r_n^{1,2:3})^T, (r_n^{2,2:3})^T]^T$ is the second and third columns of R_n.

Let $h_k^1 \in \mathbb{R}^{K \times 1}, h_k^2 \in \mathbb{R}^{2K \times 2}$ be the kth column and double-column of H^1, H^2, respectively. Then, from Eqs. (17) and (18), we get:

$$\hat{\mathit{\Pi}}_n^1 h_k^1 = \begin{bmatrix} \hat{\pi}_n^{1,1K} \\ \hat{\pi}_n^{2,1K} \end{bmatrix} h_k^1 = z_{nk} \begin{bmatrix} r_n^{11} \\ r_n^{21} \end{bmatrix} \qquad \hat{\mathit{\Pi}}_n^2 h_k^2 = \begin{bmatrix} \hat{\pi}_n^{1,K+1:3K} \\ \hat{\pi}_n^{2,K+1:3K} \end{bmatrix} h_k^2 = z_{nk} \begin{bmatrix} r_n^{1,2:3} \\ r_n^{2,2:3} \end{bmatrix} \tag{19}$$

By merging the equations of Eq. (19) together, R_n can be represented by:

$$[\hat{\Pi}_n^1 h_k^1, \hat{\Pi}_n^2 h_k^2] = \begin{bmatrix} \hat{\pi}_n^{1,1:K}, \hat{\pi}_n^{1,K+1:3K} \\ \hat{\pi}_n^{2,1:K}, \hat{\pi}_n^{2,K+1:3K} \end{bmatrix} \begin{bmatrix} h_k^1, & \mathbf{0}_{K\times 2K} \\ \mathbf{0}_{2K\times K}, & h_k^2 \end{bmatrix} = z_{nk} R_n. \tag{20}$$

Remark 5. Similar to the rigid symmetry case in [26], Eq. (20) indicates that there is no rotation ambiguities on the symmetric direction. The rotation ambiguities only exist in the yz-plane (*i.e.* the non-symmetric plane).

The orthonormality constraints $R_n R_n^T = I$ can be imposed to estimate h_k^1, h_k^2:

$$\begin{aligned} &[\hat{\Pi}_n^1 h_k^1, \hat{\Pi}_n^2 h_k^2][\hat{\Pi}_n^1 h_k^1, \hat{\Pi}_n^2 h_k^2]^T = z_{nk}^2 I \\ &= \begin{bmatrix} \hat{\pi}_n^{1,1:K}, \hat{\pi}_n^{1,K+1:3K} \\ \hat{\pi}_n^{2,1:K}, \hat{\pi}_n^{2,K+1:3K} \end{bmatrix} \begin{bmatrix} h_k^1 h_k^{1T}, & \mathbf{0}_{K\times 2} \\ \mathbf{0}_{2K\times 1}, & h_k^2 h_k^{2T} \end{bmatrix} \begin{bmatrix} \hat{\pi}_n^{1,1:K}, \hat{\pi}_n^{1,K+1:3K} \\ \hat{\pi}_n^{2,1:K}, \hat{\pi}_n^{2,K+1:3K} \end{bmatrix}^T \end{aligned} \tag{21}$$

Thus, we have:

$$\hat{\pi}_n^{1,1:K} h_k^1 h_k^{1T} (\hat{\pi}_n^{1,1:K})^T + \hat{\pi}_n^{1,K+1:3K} h_k^2 h_k^{2T} (\hat{\pi}_n^{1,K+1:3K})^T = z_{nk}^2 \tag{22}$$

$$\hat{\pi}_n^{2,1:K} h_k^1 h_k^{1T} (\hat{\pi}_n^{2,1:K})^T + \hat{\pi}_n^{2,K+1:3K} h_k^2 h_k^{2T} (\hat{\pi}_n^{2,K+1:3K})^T = z_{nk}^2 \tag{23}$$

$$\hat{\pi}_n^{1,1:K} h_k^1 h_k^{1T} (\hat{\pi}_n^{2,1:K})^T + \hat{\pi}_n^{1,K+1:3K} h_k^2 h_k^{2T} (\hat{\pi}_n^{2,K+1:3K})^T = 0 \tag{24}$$

Remark 6. The main difference of the derivations from the orthonormality constraints between the rigid and non-rigid cases is that, for the rigid case, the dot product of each row of $\mathbf{R}$ is equal to 1, while for non-rigid the dot product on each row of $\mathbf{\Pi}$ gives us a unknown value z_{nk}^2. But note that z_{nk}^2 is the same for the both rows, *i.e.* Eqs. (22) and (23), corresponding to the same projection.

Eliminating the unknown value z_{nk}^2 in Eqs. (22) and (23) (by subtraction) and rewriting in vectorized form gives:

$$\begin{aligned} &\begin{bmatrix} \hat{\pi}_n^{1,1:K} \otimes \hat{\pi}_n^{1,1:K} - \hat{\pi}_n^{2,1:K} \otimes \hat{\pi}_n^{2,1:K}, & \hat{\pi}_n^{1,K+1:3K} \otimes \hat{\pi}_n^{1,K+1:3K} - \hat{\pi}_n^{2,K+1:3K} \otimes \hat{\pi}_n^{1,K+1:3K} \\ \hat{\pi}_n^{1,1:K} \otimes \hat{\pi}_n^{2,1:K}, & \hat{\pi}_n^{1,K+1:3K} \otimes \hat{\pi}_n^{2,K+1:3K} \end{bmatrix} \\ &\cdot \begin{bmatrix} \mathrm{vec}(h_k^1 h_k^{1T}) \\ \mathrm{vec}(h_k^2 h_k^{2T}) \end{bmatrix} = A_n \begin{bmatrix} \mathrm{vec}(h_k^1 h_k^{1T}) \\ \mathrm{vec}(h_k^2 h_k^{2T}) \end{bmatrix} = 0, \end{aligned} \tag{25}$$

Letting $\mathbf{A} = [A_1^T, ..., A_N^T]^T$, yield the constraints:

$$\mathbf{A}[\mathrm{vec}(h_k^1 h_k^{1T})^T, \mathrm{vec}(h_k^2 h_k^{2T})^T]^T = 0. \tag{26}$$

Remark 7. As shown in Xiao *et al.* [4], the orthonormality constraints, *i.e.* Eq. (26), are not sufficient to solve for the ambiguity matrix H. But Xiao *et al.* showed that the solution of $[\mathrm{vec}(h_k^1 h_k^{1T})^T, \mathrm{vec}(h_k^2 h_k^{2T})^T]^T$ lies in the null space of $\mathbf{A}$ of dimensionality $(2K^2 - K)$ [4]. Akhter *et al.* [6] proved that this was a "gauge freedom" because all legitimate solutions lying in this subspace (despite under-constrained) gave the same solutions for the 3D structure. More technically, the ambiguity of H corresponds only to a linear combination of H's column-triplet and a rotation on H [25]. This observation was exploited by Dai *et al.* in [9,10], where they showed that, up to the ambiguities aforementioned, $h_k h_k^T$ can be solved by the intersection of 3 subspaces as we will describe in the following.

Following the strategy in [9,10], we have intersection of subspaces conditions:

$$\left\{\mathbf{A}\begin{bmatrix}\text{vec}(h_k^1 h_k^{1T})\\ \text{vec}(h_k^2 h_k^{2T})\end{bmatrix}=0\right\}\cap\left\{\begin{matrix}h_k^1 h_k^{1T}=0\\ h_k^2 h_k^{2T}\succeq 0\end{matrix}\right\}\cap\left\{\begin{matrix}\text{rank}(h_k^1 h_k^{1T})=1\\ \text{rank}(h_k^2 h_k^{2T})=2\end{matrix}\right\} \tag{27}$$

The first subspace comes from Eq. (26), *i.e.* the solutions of the Eq. (26) lie in the null space of $\mathbf{A}$ of dimensionality $(2K^2-K)$ [4]. The second subspace requires that $h_k^1 h_k^{1T}$ and $h_k^2 h_k^{2T}$ are positive semi-definite. The third subspace comes from the fact that h_k^1 is of rank 1 and h_k^2 is of rank 2.

Note that as stated in [9,10], Eq. (27) imposes all the necessary constraints on $[\text{vec}(h_k^1\ \ h_k^{1T})^T, \text{vec}(h_k^2 h_k^{2T})^T]^T$. There is no difference in the recovered 3D structures using the different solutions that satisfy Eq. (27).

We can obtain a solution of $[\text{vec}(h_k^1 h_k^{1T})^T, \text{vec}(h_k^2 h_k^{2T})^T]^T$, under the condition of Eq. (27), by standard semi-definite programming (SDP):

$$\begin{aligned}&\min ||h_k^1 h_k^{1T}||_* + ||h_k^2 h_k^{2T}||_*\\ \text{s. t. } h_k^1 h_k^{1T}\succeq 0,\quad h_k^2 h_k^{2T}\succeq 0\quad &\mathbf{A}[\text{vec}(h_k^1 h_k^{1T})^T, \text{vec}(h_k^2 h_k^{2T})^T]^T=0,\end{aligned} \tag{28}$$

where $||\cdot||_*$ indicates the trace norm.

Remark 8. After recovering h_k^1 and h_k^2, we can estimate the camera parameters R as follows. Note that it does not need to the whole ambiguity matrix H [9,10].

After h_k^1, h_k^2 has been solved, Eq. (20) (*i.e.* $[\hat{\Pi}_n^1 h_k^1, \hat{\Pi}_n^2 h_k^2] = z_{nk} R_n$) implies that the camera projection matrix R_n can be obtained by normalizing the two rows of $[\hat{\Pi}_n^1 h_k^1, \hat{\Pi}_n^2 h_k^2]$ to have unit ℓ_2 norm. Then, $\mathbf{R}$ can be constructed by $\mathbf{R} = \text{blkdiag}([R_1, ..., R_N])$.

Remark 9. After estimated the camera parameters, we can solve for the 3D structure adopting the methods in [9,10], *i.e.* by minimizing a *low-rank* constraint on rearranged (*i.e.* more compact) $\mathbf{S}^\sharp$ under the orthographic projection model.

Similar to [9,10], the structure $\mathbf{S}$ can be estimated by:

$$\begin{aligned}&\min ||\mathbf{S}^\sharp||_*\\ \text{s. t. } [\mathbf{Y}, \mathbf{Y}^\dagger] = \mathbf{R}[\mathbf{S}, \mathcal{A}_N\mathbf{S}]\quad &\mathbf{S}^\sharp = [\mathcal{P}_x, \mathcal{P}_y, \mathcal{P}_z](I_3\otimes\mathbf{S}),\end{aligned} \tag{29}$$

where $\mathcal{A}_N = I_N \otimes \text{diag}([-1, 1, 1])$, $\mathbf{S} = [S_1^T, ..., S_N^T]^T \in \mathbb{R}^{3N\times P}$ and $\mathbf{S}^\sharp \in \mathbb{R}^{N\times 3P}$ is rearranged and more compact $\mathbf{S}$, *i.e.*

$$\mathbf{S}^\sharp = \begin{bmatrix} x_{11}, & ..., & x_{1P}, & y_{11}, & ..., & y_{1P}, & z_{11}, & ..., & z_{1P}\\ \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots\\ x_{N1}, & ..., & x_{NP}, & y_{N1}, & ..., & y_{NP}, & z_{N1}, & ..., & z_{NP}\end{bmatrix},$$

and $\mathcal{P}_x, \mathcal{P}_y, \mathcal{P}_z \in \mathbb{R}^{N\times 3N}$ are the row-permutation matrices of 0 and 1 that select $(I_3 \otimes \mathbf{S})$ to form $\mathbf{S}^\sharp$, *i.e.* $\mathcal{P}_x(i, 3i-2) = 1, \mathcal{P}_y(i, 3i-1) = 1, \mathcal{P}_z(i, 3i) = 1$ for $i = 1, ..., N$.

Remark 10. After obtaining the initial estimates of R_n, S_n (from matrix factorization as described above) and the occluded keypoints, we can minimize the full energy (likelihood) in Eq. (11) d by *coordinate descent* to obtain better estimates of R_n, S_n and the occluded keypoints.

Energy Minimization. After obtained initial $\mathbf{R}$, $\mathbf{S}$ and missing points, Eq. (11) can be minimized by coordinate descent. The energy about $\mathbf{R}$, $\mathbf{S}$ is:

$$\mathcal{Q}(\mathbf{R}, \mathbf{S}) = ||\mathbf{Y} - \mathbf{R}\mathbf{S}||_2^2 + ||\mathbf{Y}^\dagger - \mathbf{R}\mathcal{A}_K\mathbf{S}||_2^2, \tag{30}$$

Note that $\mathbf{S}$ can be updated exactly as the same as its initialization in Eq. (29) by the low-rank constraint. While each R_n of $\mathbf{R}$ should be updated under the nonlinear orthonormality constraints $R_n R_n^T = I$ similar to the idea in EM-PPCA [5]: we first parameterize R_n to a full 3×3 rotation matrix Q and update Q by its rotation increment. Please refer to the supplementary material for the details.

The occluded points $Y_{n,p}$ and $Y_{n,p}^\dagger$ with $(n, p) \in IVS$ are updated by minimizing the full energy in Eq. (11) directly:

$$Y_{n,p} = R_n S_p, \quad Y_{n,p}^\dagger = R_n \mathcal{A} S_{n,p} \tag{31}$$

Similar to Sym-RSfM [26], after updating the occluded points, we also re-estimate the translation for each image by $t_n = \sum_p (Y_{n,p} - R_n S_p + Y_{n,p}^\dagger - R_n \mathcal{A} S_p)$, then centralize the data again by $Y_n \leftarrow Y_n - \mathbf{1}_P^T \otimes t_n$ and $Y_n^\dagger \leftarrow Y_n^\dagger - \mathbf{1}_P^T \otimes t_n$.

5 Experiments

5.1 Experimental Settings

We follow the experimental settings in [24], using the 2D annotations in [39] and 3D keypoints in Pascal3D+ [27]. Although Pascal3D+ is the best 3D dataset available, it still has some limitations for our task. Specifically, it does not have the complete 3D models for each object; instead it provides the 3D shapes of object subtypes. For example, it provides 10 subtypes for *car* category, such as *sedan, truck*, which ignores the shape variance within each subtype.

Similar to [7,9,10,35], the rotation error e_R and shape error e_S are used for evaluation. We normalize 3D groundtruth and our 3D estimates to eliminate different scales they may have. For each shape S_n we use its standard deviations in X, Y, Z coordinates $\sigma_n^x, \sigma_n^y, \sigma_n^z$ for the normalization: $S_n^{\text{norm}} = 3S_n/(\sigma_n^x + \sigma_n^y + \sigma_n^z)$. To deal with the rotation ambiguity between the 3D groundtruth and our reconstruction, we use the Procrustes method [40] to align them. Then, the rotation error e_R and shape error e_S can be calculated as:

$$e_R = \frac{1}{N}\sum_{n=1}^{N} ||R_n^{\text{aligned}} - R_n^*||_F, e_S = \frac{1}{2NP}\sum_{n=1}^{N}\sum_{p=1}^{2P} ||S_{n,p}^{\text{norm aligned}} - S_{n,p}^{\text{norm}*}||_F, \tag{32}$$

where R_n^{aligned} and R_n^* are the recovered and the groundtruth camera projection matrix for image n. $S_{n,p}^{\text{norm aligned}}$ and $S_{n,p}^{\text{norm}*}$ are the normalized estimated and the normalized groundtruth structure for the p'th point of image n. R_n^{aligned} and R_n^*, $S_{n,p}^{\text{norm aligned}}$ and $S_{n,p}^{\text{norm}*}$ are aligned by Procrustes method [40].

5.2 Experimental Results on the Annotated Dataset

In this section, we construct the 3D keypoints for each image using the non-rigid model. Firstly, we follow the experimental setting in [24] and collect all images with more than 5 visible keypoints. Then, we do 10 iterations with rank 3 recovery to initialize the occluded/missing data. In this experiments, we use 3 deformation bases and set λ in Sym-EM-PPCA, *i.e.* in Eqs. (9), as 1.

We tested our algorithm on Pascal *aeroplane, bus, car, sofa, train, tv* based on the mean shape and rotation errors as in Eq. (32). For the shape error, since Pascal3D+ [27] only provides one 3D model for each subtype, we compare the reconstructed 3D keypoints for each image with their subtype groundtruth. More specifically, the reconstructed 3D keypoints for each image are grouped into subtypes, and we count the mean shape error (we also have median errors in the supplementary material) by comparing all the images within that subtype to the subtype groundtruth from Pascal3D+. While such problem does not exist for the rotation errors, *i.e.* the groundtruth projection is available for each image in Pascal3D+ [27], thus the rotation errors are reported by each category.

The results are reported in Table 1. The results show that our method outperforms the baselines in general. But we note that our method is not as good as the baselines in some cases, especially for *tv*. The reasons might be: (i) the orthographic projection is inaccurate when the object is close to the camera. Although all the methods used the same suboptimal orthographic projection for these cases, it may deteriorate more on our model sometimes, since we model more constraints. (ii) It might be because the 3D groundtruth in Pascal3D+,

Table 1. The mean *shape* and *rotation* errors for *aeroplane, bus, car, sofa, train, tv*. The Roman numerals indicates the index of subtypes for the mean shape error, and mRE is short for the mean rotation error. EP, PF, Sym-EP, Sym-PF are short for EM-PPCA [5], PriorFree [9,10], Sym-EM-PPCA, Sym-PriorFree, respectively.

	aeroplane								bus						
	I	II	III	IV	V	VI	VII	mRE	I	II	III	IV	V	VI	mRE
EP	0.36	0.59	0.50	0.49	0.57	0.57	**0.45**	0.34	0.42	0.34	0.56	0.54	0.98	0.86	0.26
PF	0.99	1.08	1.13	1.15	1.22	1.10	1.11	0.52	1.62	1.56	1.75	1.59	2.09	1.70	0.47
Sym-EP	**0.33**	**0.53**	**0.46**	**0.43**	**0.51**	**0.53**	0.46	**0.31**	**0.28**	**0.25**	**0.33**	**0.33**	**0.65**	**0.46**	**0.21**
Sym-PF	0.57	0.76	0.84	0.76	0.73	0.61	0.79	0.46	1.92	1.95	1.77	1.54	1.70	1.42	1.23
	car											sofa			
	I	II	III	IV	V	VI	VII	VIII	IX	X	mRE	I	II	III	
EP	1.10	1.01	1.09	1.05	1.03	1.07	0.99	1.46	1.00	0.85	0.39	2.00	1.87	2.03	
PF	1.76	1.67	1.76	1.77	1.65	1.79	1.67	1.57	1.70	1.42	0.86	1.71	1.41	1.46	
Sym-EP	**0.99**	**0.89**	**1.05**	**1.02**	**0.92**	**1.00**	**0.89**	**1.39**	**0.95**	**0.68**	**0.34**	**1.18**	**0.81**	**1.08**	
Sym-PF	1.74	1.41	1.70	1.48	1.69	1.58	1.43	1.69	1.52	1.30	0.79	1.33	1.15	1.36	
	sofa				train					tv					
	IV	V	VI	mRE	I	II	III	IV	mRE	I	II	III	IV	mRE	
EP	1.99	2.37	1.81	0.79	1.18	0.53	0.49	0.42	0.85	**0.44**	**0.51**	**0.44**	**0.36**	**0.41**	
PF	2.02	2.66	1.64	1.36	1.97	**0.27**	0.47	0.34	0.98	0.56	1.01	0.97	0.65	0.80	
Sym-EP	1.12	1.80	**0.88**	**0.34**	**0.95**	0.46	**0.42**	**0.31**	**0.73**	0.51	0.60	0.53	0.64	0.52	
Sym-PF	**1.02**	**1.17**	0.95	0.85	1.52	0.40	0.49	0.47	0.99	0.60	1.01	1.15	0.51	0.86	

which neglects the shape variations in the same subtype, may not be accurate enough (*e.g.* it has only one 3D model for all the *sedan* cars).

5.3 Experimental Results on the Noisy Annotations

We also investigate what happens if the keypoints are not perfectly annotated. This is important to check because our method depends on keypoint pairs therefore may be sensitive to errors in keypoint location, which will inevitably arise when we use features detectors, *e.g.* deep nets [41], to detect the keypoints.

To simulate this, we add Gaussian noise $\mathcal{N}(0, \sigma^2)$ to the 2D annotations and re-do the experiments. The standard deviation is set to $\sigma = sd_{max}$, where d_{max} is the longest distance between all the keypoints (*e.g.* for *aeroplane*, it is the distance between the nose tip to the tail tip). We have tested for different s by: 0.03, 0.05, 0.07. The other parameters are the same as the previous section.

The results for *aeroplane* with $s = 0.03, 0.05, 0.07$ are shown in Table 2. Each result value is obtained by averaging 10 repetitions. The results in Table 2 show that the performances of all the methods decrease in general with the increase in the noise level. Nonetheless, our methods still outperform our counterparts with the noisy annotations (*i.e.* the imperfectly labeled annotations).

Table 2. The mean *shape* and *rotation* errors for *aeroplane* with imperfect annotations. The noise is Gaussian $\mathcal{N}(0, \sigma^2)$ with $\sigma = sd_{max}$, where we choose $s = 0.03, 0.05, 0.07$ and d_{max} is the longest distance between all the keypoints (*i.e.* the tip of the nose to the tip of the tail for aeroplane). Other parameters are the same as those in Table 1. Each result value is obtained by averaging 10 repetitions.

	$\sigma = \mathbf{0.03}\ d_{max}$								$\sigma = \mathbf{0.05}\ d_{max}$			
	I	II	III	IV	V	VI	VII	mRE	I	II	III	IV
EP	**0.34**	0.59	0.49	0.45	0.54	0.55	**0.45**	0.33	0.37	0.58	0.51	0.47
PF	0.92	1.01	1.05	1.06	1.13	1.03	1.06	0.52	0.93	1.04	1.05	1.08
Sym-EP	0.34	**0.54**	**0.47**	**0.44**	**0.52**	**0.55**	0.46	**0.32**	**0.35**	**0.54**	**0.47**	**0.43**
Sym-PF	0.79	0.93	1.01	0.93	0.91	0.79	0.94	0.60	0.83	0.99	1.09	0.98
	$\sigma = \mathbf{0.05}\ d_{max}$				$\sigma = \mathbf{0.07}\ d_{max}$							
	V	VI	VII	mRE	I	II	III	IV	V	VI	VII	mRE
EP	0.57	0.57	0.46	0.35	0.38	0.61	0.50	0.45	0.61	**0.56**	**0.46**	0.36
PF	1.15	1.02	1.07	0.54	0.94	1.04	1.08	1.07	1.15	1.03	1.08	0.65
Sym-EP	**0.52**	**0.57**	**0.46**	**0.33**	**0.37**	**0.58**	**0.49**	**0.44**	**0.58**	0.57	0.46	**0.35**
Sym-PF	0.94	0.84	1.04	0.63	0.94	1.06	1.15	1.04	1.05	0.89	1.08	0.70

6 Conclusion

This paper shows that non-rigid SfM can be extended to the important special case where the objects are symmetric, which is frequently possessed by man-made objects [15,16]. We derive and implement this extension to two popular non-rigid structure from motion algorithms [5,9,10], which perform well on the Pascal3D+ dataset when compared to the baseline methods.

In this paper, we have focused on constructing the non-rigid SfM model(s) that can exploit the symmetry property. In future work, we will extend to perspective projection, apply a better initialization of the occluded keypoints such as low-rank recovery, use additional object features (instead of just key-points), and detect these features from images automatically such as [41].

Acknowledgment. We would like to thank Ehsan Jahangiri, Cihang Xie, Weichao Qiu, Xuan Dong, Siyuan Qiao for giving feedbacks on the manuscript. This work was supported by ARO 62250-CS and ONR N00014-15-1-2356.

References

1. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: a factorization method. Int. J. Comput. Vis. **9**(2), 137–154 (1992)
2. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press, Cambridge (2004)
3. Torresani, L., Hertzmann, A., Bregler, C.: Learning non-rigid 3D shape from 2D motion. In: NIPS (2003)
4. Xiao, J., Chai, J., Kanade, T.: A closed-form solution to non-rigid shape and motion recovery. In: Pajdla, T., Matas, J.G. (eds.) ECCV 2004. LNCS, vol. 3024, pp. 573–587. Springer, Heidelberg (2004)
5. Torresani, L., Hertzmann, A., Bregler, C.: Nonrigid structure-from-motion: estimating shape and motion with hierarchical priors. IEEE Trans. Pattern Anal. Mach. Intell. **30**, 878–892 (2008)
6. Akhter, I., Sheikh, Y., Khan, S., Kanade, T.: Trajectory space: a dual representation for nonrigid structure from motion. IEEE Trans. Pattern Anal. Mach. Intell. **33**(7), 1442–1456 (2011)
7. Gotardo, P., Martinez, A.: Computing smooth timetrajectories for camera and deformable shape in structure from motion with occlusion. IEEE Trans. Pattern Anal. Mach. Intell. **33**, 2051–2065 (2011)
8. Hamsici, O.C., Gotardo, P.F.U., Martinez, A.M.: Learning spatially-smooth mappings in non-rigid structure from motion. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part IV. LNCS, vol. 7575, pp. 260–273. Springer, Heidelberg (2012)
9. Dai, Y., Li, H., He, M.: A simple prior-free method for non-rigid structure-from-motion factorization. In: CVPR (2012)
10. Dai, Y., Li, H., He, M.: A simple prior-free method for non-rigid structure-from-motion factorization. Int. J. Comput. Vis. **107**, 101–122 (2014)
11. Ma, J., Zhao, J., Ma, Y., Tian, J.: Non-rigid visible and infrared face registration via regularized gaussian fields criterion. Pattern Recogn. **48**(3), 772–784 (2015)
12. Ma, J., Zhao, J., Tian, J., Tu, Z., Yuille, A.L.: Robust estimation of nonrigid transformation for point set registration. In: CVPR (2013)
13. Ma, J., Zhao, J., Tian, J., Bai, X., Tu, Z.: Regularized vector field learning with sparse approximation for mismatch removal. Pattern Recogn. **46**(12), 3519–3532 (2013)
14. Agudo, A., Agapito, L., Calvo, B., Montiel, J.: Good vibrations: a modal analysis approach for sequential non-rigid structure from motion. In: CVPR, pp. 1558–1565 (2014)

15. Rosen, J.: Symmetry Discovered: Concepts and Applications in Nature and Science. Dover Publications, Mineola (2011)
16. Hong, W., Yang, A.Y., Huang, K., Ma, Y.: On symmetry and multiple-view geometry: structure, pose, and calibration from a single image. Int. J. Comput. Vis. **60**, 241–265 (2004)
17. Gordon, G.G.: Shape from symmetry. In: Proceedings of SPIE (1990)
18. Kontsevich, L.L.: Pairwise comparison technique: a simple solution for depth reconstruction. JOSA A **10**(6), 1129–1135 (1993)
19. Vetter, T., Poggio, T.: Symmetric 3D objects are an easy case for 2D object recognition. Spat. Vis. **8**, 443–453 (1994)
20. Mukherjee, D.P., Zisserman, A., Brady, M.: Shape from symmetry: detecting and exploiting symmetry in affine images. Philos. Trans. Phys. Sci. Eng. **351**, 77–106 (1995)
21. Thrun, S., Wegbreit, B.: Shape from symmetry. In: ICCV (2005)
22. Li, Y., Pizlo, Z.: Reconstruction of shapes of 3D symmetric objects by using planarity and compactness constraints. In: Proceedings of SPIE-IS&T Electronic Imaging (2007)
23. Vicente, S., Carreira, J., Agapito, L., Batista, J.: Reconstructing PASCAL VOC. In: CVPR (2014)
24. Kar, A., Tulsiani, S., Carreira, J., Malik, J.: Category-specific object reconstruction from a single image. In: CVPR (2015)
25. Akhter, I., Sheikh, Y., Khan, S.: In defense of orthonormality constraints for non-rigid structure from motion. In: CVPR (2009)
26. Gao, Y., Yuille, A.L.: Exploiting symmetry and/or Manhattan properties for 3D object structure estimation from single and multiple images (2016). arXiv preprint arXiv:1607.07129
27. Xiang, Y., Mottaghi, R., Savarese, S.: Beyond PASCAL: a benchmark for 3D object detection in the wild. In: WACV (2014)
28. Grossmann, E., Santos-Victor, J.: Maximum likehood 3D reconstruction from one or more images under geometric constraints. In: BMVC (2002)
29. Grossmann, E., Santos-Victor, J.: Least-squares 3D reconstruction from one or more views and geometric clues. Comput. Vis. Image Underst. **99**(2), 151–174 (2005)
30. Grossmann, E., Ortin, D., Santos-Victor, J.: Single and multi-view reconstruction of structured scenes. In: ACCV (2002)
31. Kontsevich, L.L., Kontsevich, M.L., Shen, A.K.: Two algorithms for reconstructing shapes. Optoelectron. Instrum. Data Process. **5**, 76–81 (1987)
32. Bregler, C., Hertzmann, A., Biermann, H.: Recovering non-rigid 3D shape from image streams. In: CVPR (2000)
33. Hong, J.H., Fitzgibbon, A.: Secrets of matrix factorization: approximations, numerics, manifold optimization and random restarts. In: ICCV (2015)
34. Olsen, S.I., Bartoli, A.: Implicit non-rigid structure-from-motion with priors. J. Math. Imaging Vis. **31**(2–3), 233–244 (2008)
35. Akhter, I., Sheikh, Y., Khan, S., Kanade, T.: Nonrigid structure from motion in trajectory space. In: NIPS (2008)
36. Ceylan, D., Mitra, N.J., Zheng, Y., Pauly, M.: Coupled structure-from-motion and 3D symmetry detection for urban facades. ACM Trans. Graph. **33**, 2 (2014)
37. Morris, D.D., Kanatani, K., Kanade, T.: Gauge fixing for accurate 3D estimation. In: CVPR (2001)
38. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, New York (2006)

39. Bourdev, L., Maji, S., Brox, T., Malik, J.: Detecting people using mutually consistent poselet activations. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part VI. LNCS, vol. 6316, pp. 168–181. Springer, Heidelberg (2010)
40. Schönemann, P.H.: A generalized solution of the orthogonal procrustes problem. Psychometrika **31**, 1–10 (1966)
41. Chen, X., Yuille, A.L.: Articulated pose estimation by a graphical model with image dependent pairwise relations. In: NIPS, pp. 1736–1744 (2014)