Exploiting Symmetry and/or Manhattan Properties for 3D Object Structure Estimation from Single and Multiple Images

Yuan Gao
Tencent AI Lab, Shenzhen, China
ethanygao@tencent.com

Alan L. Yuille
Johns Hopkins University, Baltimore, MD
UCLA, Los Angeles, CA
alan.yuille@jhu.edu

Abstract

Many man-made objects have intrinsic symmetries and Manhattan structure. By assuming an orthographic projection model, this paper addresses the estimation of 3D structures and camera projection using symmetry and/or Manhattan structure cues, which occur when the input is single- or multiple-image from the same category, e.g., *multiple different cars. Specifically, analysis on the single image case implies that Manhattan alone is sufficient to recover the camera projection, and then the 3D structure can be reconstructed uniquely exploiting symmetry. However, Manhattan structure can be difficult to observe from a single image due to occlusion. To this end, we extend to the multiple-image case which can also exploit symmetry but does not require Manhattan axes. We propose a novel rigid structure from motion method, exploiting symmetry and using multiple images from the same category as input. Experimental results on the Pascal3D+ dataset show that our method significantly outperforms baseline methods.*

1. Introduction

Many objects, especially these made by humans, have intrinsic symmetry [43, 25] and Manhattan properties (meaning that 3 perpendicular axes are inferable on the object [9, 10, 13]), such as cars, aeroplanes, see Fig 1. The purpose of this paper is to investigate the benefits of using symmetry and/or Manhattan constraints to estimate the 3D structures of objects from one or more images. As a key task in computer vision, numerous studies have been conducted on estimating the 3D shapes of objects from multiple images [23, 47, 52, 48, 4, 17, 21, 11, 12, 1]. There is also a long history of research on the use of symmetry [16, 30, 49, 40, 25, 45, 32, 28] and a growing body of work on Manhattan world [9, 10, 13]. There is, however, little work that combines these cues.

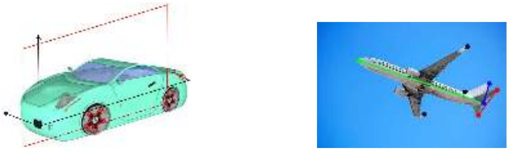

Figure 1. Left Panel: Illustration of symmetry and Manhattan structure. The car has a bilateral symmetry with respect to the plane in red. There are three Manhattan axes. The first is normal to the symmetry plane of the car (*e.g.*, from *left wheels* to *right wheels*). The second is from the front to the back of the car (*e.g.*, from *back wheels* to *front wheels*) while the third is in the vertical direction. Right Panel: Illustration of the 3 Manhattan directions on a real aeroplane image, shown by Red, Green, Blue lines. These 3 Manhattan directions can be obtained directly from the labeled keypoints.

This paper aims at estimating the 3D structure of an object class, taking a single or multiple intra-class instances as input, *e.g.*, different cars from various viewpoints. Following [50, 29, 15], we use 2D positions of keypoints as input to estimate the 3D structure and the camera projection, leaving the detection of the 2D keypoints to methods such as [8]. In this paper, different combinations of the three cues, *i.e.*, symmetry, Manhattan and multiple images, are investigated and two algorithms/derivations are proposed (assuming orthographic projection), *i.e.*, single image reconstruction using both symmetry and Manhattan constraints, and multiple-image reconstruction using symmetry[1].

Specifically, we start with the single image reconstruction case, using the symmetry and Manhattan constraints, see Fig. 1. Our derivation is inspiring that a single image is sufficient for reconstruction when the symmetry and Manhattan axes can be inferred. Specifically, our analysis shows that using Manhattan alone is sufficient to recover the camera projection (up to several axis-sign ambiguities) by a single input image, then the 3D structure can be reconstructed uniquely by symmetry thereafter. We show these results on

[1] We experimented with using Manhattan for the multiple-image case, but found it gave negligible improvement. Please see the derivations of using Manhattan for the multiple-image case in the supplementary material.

aeroplane in Pascal3D+. But we note that all the 3 Manhattan axes are hard to observe from a single image sometimes, particularly if the keypoints we rely on are occluded.

Hence, we extend the use of symmetry to the multiple-image case using structure from motion (SfM). The input is different intra-class object instances with various viewpoints. We formulate the problem in terms of energy minimization (*i.e.*, MLE) of a probabilistic model, in which symmetry constraints are included. The energy also involves missing/latent variables for unobserved keypoints due to occlusion. These complications, and in particular symmetry, implies that we cannot directly apply singular value decomposition (SVD) (as in the previous SfM based work [31, 46]) to directly minimize the energy function. Instead we must rely on coordinate descent (or hard EM methods if we treat the missing points are latent variable) which risk getting stuck in local minimum. To address this issue we define a *surrogate energy function* which exploits symmetry, by grouping the keypoints into symmetric keypoint pairs, and assumes that the missing data are known (*e.g.*, initialized by another process). We show that the surrogate energy can be decomposed into the sum of two independent energies, each of which can be minimized directly using SVD. This leads to a two-stage strategy where we first minimize the surrogate energy function as initialization for coordinate descent on the original energy function.

Recall that the classic SfM has a "gauge freedom" [39] because we can rotate the object and camera pose by equivalent amounts without altering the observations in the 2D images. This gauge freedom can be thought of as the freedom to *choose our coordinate system*. In this paper, we exploit this freedom to choose the symmetry axis to be in the x axis (and the other two Manhattan axes to be in the y and z directions). In the following, we group keypoints into keypoint pairs and use a superscript † to denote symmetry, *e.g.*, Y and $Y^\dagger$ are symmetric keypoint pairs.

The rest of the paper is organized as follows: firstly, we review related work in Section 2. In Section 3, we describe the common experiment design for all our methods. Then the mathematical details and evaluations on the single image reconstruction are given in Section 4, followed by the derivation and experiments on the multiple-image case, *i.e.*, the symmetric rigid structure from motion (Sym-RSfM), in Section 5. Finally, we give our conclusions in Section 6.

2. Related Works

Symmetry has been studied in computer vision for several decades. For example, symmetry has been used as a cue in depth recovery [16, 30, 40] as well as for recognizing symmetric objects [49]. Grossmann and Santos-Victor utilized various geometric clues, such as planarity, orthogonality, parallelism and symmetry, for 3D scene reconstruction [19, 20], where the camera rotation matrix was pre-computed by vanishing points [18]. Recently, researchers applied symmetry to scene reconstruction [25], and 3D mesh reconstruction with occlusion [45]. In addition, symmetry, incorporated with planarity and compactness priors, has also been studied to reconstruct structures defined by 3D keypoints [32]. By contrast, the Manhattan world assumption was developed originally for scenes [9, 10, 13], where the authors assumed visual scenes were based on a Manhattan 3D grid which provided 3 perpendicular axis constaints. Both symmetry and Manhattan can be straightforwardly combined, and adapted to 3D object reconstruction, particularly for man made objects.

The estimation of 3D structure from multiple images is one of the most active research areas in computer vision. Classic SfM for rigid objects built on matrix factorization methods [31, 46]. Then, more general non-rigid deformation was considered, and the rigid SfM in [31, 46] was extended to non-rigid case by Bregler *et al*. [6]. Non-rigid SfM was shown to have ambiguities [52] and various non-rigid SfM methods were proposed using priors on the non-rigid deformations [52, 48, 41, 3, 17, 4]. Gotardo and Martinez proposed a Column Space Fitting (CSF) method for rank-r matrix factorization and applied it to SfM with smooth time-trajectories assumption [17]. A more general framework for rank-r matrix factorization was proposed in [24], containing the CSF method as a special case[2]. More recently, it has been proved that the ambiguities in non-rigid SfM do not affect the estimated 3D structure, [2] which leaded to prior free matrix factorization methods [11, 12].

SfM methods have been used for category-specific object reconstruction, *e.g.*, estimating the structure of *cars* from images of different *cars* under various viewing conditions [29, 50], but these did not exploit symmetry or Manhattan. We point out that in [7], the repetition patterns have been incorporated into SfM for urban facades reconstruction, but [7] focused mainly on repetition detection and registration.

3. Experimental Design

This paper discusses 2 different scenarios to reconstruct the 3D structure: (i) reconstruction from a single image using symmetry and the Manhattan assumptions, (ii) reconstruction from multiple images using symmetric rigid SfM. The experiments are performed on Pascal3D+ dataset. This contains object categories such as *aeroplane* and *car*. These object categories are sub-divided into subtypes, such as *sedan* car. For each object subtype, we estimate an 3D structure and the viewpoints of all the within-subtype instances. The 3D structure is specified by the 3D keypoints in Pascal3D+ [51] and the corresponding keypoints in the

[2] However, the general framework in [24] cannot be used to SfM directly, because it did not constrain that all the keypoints within the same frame should have the same translation. Instead, [24] focused on better optimization of rank-r matrix factorization and pursuing better runtime.

2D images are from Berkeley [5]. These are the same experimental settings as used in [29, 15].

For evaluation we report the rotation error e_R and the shape error e_S, as in [11, 12, 3, 17, 15]. The 3D groundtruth and our estimates may have different scales, so we normalize them before evaluation. 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 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 result, we use the Procrustes method [44] to align them. Assuming we have $2P$ keypoints, *i.e.*, P keypoint pairs, the rotation error e_R and the shape error e_S are calculated as:

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

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 structure and the normalized groundtruth structure for the p'th point of image n. These are aligned by the Procrustes method [44].

4. 3D Reconstruction of A Single Image

In this section, we describe how to reconstruct the 3D structure of an object from a single image using its symmetry and Manhattan properties. Theoretical analysis shows that this can be done with a bilateral symmetry and three Manhattan axes, but the estimation is ambiguous if less than three Manhattan axes are visible. Specifically, the three Manhattan constraints alone are sufficient to determine the camera projection up to sign ambiguities (*e.g.*, we cannot distinguish between front-to-back and back-to-front directions). Then, the symmetry property is sufficient to estimate the 3D structure uniquely thereafter.

Let $Y, Y^\dagger \in \mathbb{R}^{2\times P}$ be the observed 2D coordinates of the P symmetric pairs, then the orthographic projection implies:

$$Y = RS, \qquad Y^\dagger = RS^\dagger, \qquad (2)$$

where $S, S^\dagger \in \mathbb{R}^{3\times P}$ are the 3D structure and $R \in \mathbb{R}^{2\times 3}$ is the camera projection matrix. We have eliminated translation by centralizing the 2D keypoints.

Remark 1 *We first estimate the camera projection matrix using the Manhattan constraints. Each Manhattan axis gives us one constraint on the camera projection. Hence, three axes give us the well defined linear equations to estimate the camera projection. We now describe this in detail.*

Consider a single Manhattan axis specified by 3D points S_a and S_b. Without loss of generality, assume that these points are along the x-axis, *i.e.*, $S_a - S_b = [x, 0, 0]^T$. It follows from the orthographic projection that:

$$Y_a - Y_b = R(S_a - S_b) = \begin{bmatrix} r_{11}, & r_{12}, & r_{13} \\ r_{21}, & r_{22}, & r_{23} \end{bmatrix} \begin{bmatrix} x \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} r_{11}x \\ r_{21}x \end{bmatrix}. \qquad (3)$$

where $R = \begin{bmatrix} r_{11}, & r_{12}, & r_{13} \\ r_{21}, & r_{22}, & r_{23} \end{bmatrix}$. Setting $Y_a = [y_a^1, y_a^2]^T$, $Y_b = [y_b^1, y_b^2]^T$, yields $r_{21}/r_{11} = (y_a^2 - y_b^2)/(y_a^1 - y_b^1)$. With other Manhattan axes, *e.g.*, if S_c, S_d are along the y-axis, S_e, S_f are along the z-axis, we can get similar constraints.

Let $\mu_1 = r_{21}/r_{11}$, $\mu_2 = r_{22}/r_{12}$, $\mu_3 = r_{23}/r_{13}$, we have:

$$\mu_1 = (y_a^2 - y_b^2)/(y_a^1 - y_b^1),$$
$$\mu_2 = (y_c^2 - y_d^2)/(y_c^1 - y_d^1),$$
$$\mu_3 = (y_e^2 - y_f^2)/(y_e^1 - y_f^1), \qquad (4)$$

Now, consider the orthogonality constraint on R, *i.e.*, $RR^T = I$, which implies:

$$r_{11}^2 + r_{12}^2 + r_{13}^2 = 1,$$
$$r_{21}^2 + r_{22}^2 + r_{23}^2 = 1,$$
$$r_{11}r_{21} + r_{12}r_{22} + r_{13}r_{23} = 0. \qquad (5)$$

Replacing r_{21}, r_{22}, r_{23} by the known values μ_1, μ_2, μ_3 of Eq. (4) indicates the following linear equations:

$$\begin{bmatrix} 1, & 1, & 1 \\ \mu_1^2, & \mu_2^2, & \mu_3^2 \\ \mu_1, & \mu_2, & \mu_3 \end{bmatrix} \begin{bmatrix} r_{11}^2 \\ r_{12}^2 \\ r_{13}^2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \qquad (6)$$

These equations can be solved for the unknowns r_{11}^2, r_{12}^2 and r_{13}^2 provided the coefficient matrix (above) is invertible (*i.e.*, has full rank). This requires that $(\mu_1 - \mu_2)(\mu_2 - \mu_3)(\mu_3 - \mu_1) \neq 0$. Because μ_1, μ_2, μ_3 are the slopes of the projected Manhattan axes in 3D space, this constraint is violated only in the very special case when the camera principal axis and two Manhattan axes are in the same plane.

Note that there are sign ambiguities for solving r_{11}, r_{12}, r_{13} from $r_{11}^2, r_{12}^2, r_{13}^2$. But these ambiguities do not affect the estimation of the 3D shape, because they are just choices of the coordinate system. Next we can calculate r_{21}, r_{22}, r_{23} directly based on r_{11}, r_{12}, r_{13} and μ_1, μ_2, μ_3. This recovers the projection matrix.

Remark 2 *We have shown that the camera projection matrix R can be recovered if the three Manhattan axes are known. Next we show that the 3D structure can be estimated using symmetry (provided the projection R is recovered).*

Assume, without loss of generality, the object is along the x-axis. Let $Y \in \mathbb{R}^{2\times P}$ and $Y^\dagger \in \mathbb{R}^{2\times P}$ be the P symmetric pairs, $S \in \mathbb{R}^{3\times P}$ and $S^\dagger \in \mathbb{R}^{3\times P}$ be the corresponding P symmetric pairs in the 3D space. Note that for the

p'th point pair in the 3D space, we have $S_p = [x_p, y_p, z_p]^T$ and $S_p^\dagger = [-x_p, y_p, z_p]$. Thus, we can re-express the camera projection in Eq. 2 by:

$$L = \frac{Y - Y^\dagger}{2} = R \begin{bmatrix} x_1, & ..., & x_P \\ 0, & ..., & 0 \\ 0, & ..., & 0 \end{bmatrix} = \begin{bmatrix} r_{11}x_1, & ..., & r_{11}x_P \\ r_{21}x_1, & ..., & r_{21}x_P \end{bmatrix}, \tag{7}$$

$$\begin{aligned} M = \frac{Y + Y^\dagger}{2} &= R \begin{bmatrix} 0, & ..., & 0 \\ y_1, & ..., & y_P \\ z_1, & ..., & z_P \end{bmatrix} \\ &= \begin{bmatrix} r_{12}y_1 + r_{13}z_1, & ..., & r_{12}y_P + r_{13}z_P \\ r_{22}y_1 + r_{23}z_1, & ..., & r_{22}y_P + r_{23}z_P \end{bmatrix}. \end{aligned} \tag{8}$$

Finally, we can solve Eqs. (7) and (8) to estimate the components (x_p, y_p, z_p) of all the points S_p (since R is known), and hence, recover the 3D structure. Observe that we have only just enough equations to solve (y_p, z_p) uniquely. On the other hand, the x_p is over-determined due to symmetry. We also note that the problem is ill-posed if we do not exploit symmetry, *i.e.*, it involves inverting a 2 × 3 projection matrix R if not exploiting symmetry.

4.1. Experiments on 3D Reconstruction Using A Single Image

We use *aeroplane*s for this experiment, because the 3 Manhattan directions (*e.g.*, left wing → right wing, nose → tail and top rudder → bottom rudder) can be obtained directly on aeroplanes, see Fig. 1. Also aeroplanes are generally far away from the camera, implying orthographic projection is a good approximation.

We selected 42 images with clear 3 Manhattan directions and with no occluded keypoints from the *aeroplane* category of Pascal3D+ dataset, and evaluated the results by the Rotation Error and Shape Error (Eq. (1)). The shape error is obtained by comparing the reconstructed structure with their subtype groundtruth model of Pascal3D+ [51].

The *average rotation and shape errors* for aeroplane using the Manhattan and symmetry constraints on the single image case are *0.3210* and *0.6047*, respectively. These results show that using the symmetry and Manhattan properties alone can give good results for the single image reconstruction. Indeed, the performance is better than some of the structure from motion (SfM) methods which use multiple images, see Tables 1 and 2 (on Page 8). But this is not a fair comparison, because these 42 images are selected to ensure that all the Manhattan axes are visible, while the SfM methods have been evaluated on all the aeroplane images. Some reconstruction results are illustrated in Fig. 2.

5. Symmetric Rigid Structure from Motion

This section describes symmetric rigid structure from motion (Sym-RSfM). We start by defining the a full energy function for the problem, see Section 5.1, and the coordinate descent algorithm to minimize it, see Section 5.2. Then, the missing points are initialized in Section 5.3. After that, we describe the surrogate energy and the algorithm to minimize it, see Section 5.4, which serves to initialize coordinate descent on the full energy function.

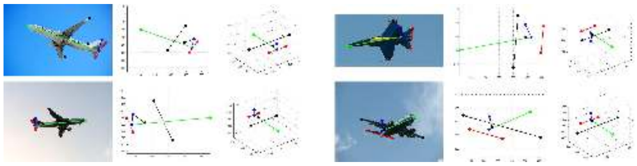

Figure 2. Illustration of the reconstruction results for *aeroplane* using the symmetry and Manhattan constraints on a single image. For each subfigure triplet, the first subfigure is the 2D image with input keypoints, the second and third subfigures are the 3D structure from the original and rectified viewpoints. The gauge freedom of sign ambiguities can also be observed by comparing the rectified 3D reconstructions, *i.e.*, the third subfigures. The Red, Green, Blue lines represent the three Manhattan directions we used.

For consistency with our baseline methods [46], we assume orthographic projection and the keypoints are centralized without translation for the Sym-RSfM. Note that due to the iterative estimation and recovery of the occluded data, the translation has to be re-estimated to re-centralize the data during each iteration. The update of the translation is straightforward and will be given in Section 5.2.

5.1. Problem Formulation: The Full Energy

We extend the image formation model from a single image to multiple images indexed by $n = 1, ..., N$. The keypoints are grouped into P keypoint pairs, which are symmetric across the x-axis. We introduce noise in the image formation, and the translation is removed as discussed above.

Then, by assuming that the 3D structure S is symmetric along the x-axis, the 2D keypoint pairs for image n, *i.e.*, $Y_n \in \mathbb{R}^{2\times P}$ and $Y_n^\dagger \in \mathbb{R}^{2\times P}$, are given by:

$$\begin{aligned} Y_n = R_n S + N_n \ &\Rightarrow\ P(Y_n | R_n, S) \sim \mathcal{N}(R_n S, 1) \\ Y_n^\dagger = R_n \mathcal{A} S + N_n \ &\Rightarrow\ P(Y_n^\dagger | R_n, S) \sim \mathcal{N}(R_n \mathcal{A} S, 1) \end{aligned} \tag{9}$$

where $R_n \in \mathbb{R}^{2\times 3}$ is the projection matrix, $S \in \mathbb{R}^{3\times P}$ is the 3D structure, and $\mathcal{A}$ is a matrix operator $\mathcal{A} = \text{diag}([-1, 1, 1])$, by which $\mathcal{A}S$ changes the sign of the first row of S, making $S^\dagger = \mathcal{A}S$. N_n is zero mean Gaussian noise with unit variance $N_n \sim \mathcal{N}(0, 1)$[3].

Observe that the noise is independent for all keypoints and for all images. Hence, the 2D keypoints are indepen-

[3] Note that we experimented with another model which treated the variance as an unknown parameter σ_n^2 and estimated it during optimization, but found this made negligible difference to the experimental results.

dent when conditioned on the 3D structure S and the camera projections R_n. Therefore, the problem is formulated in terms of minimizing the following energy function with unknown R_n, S:

$$\begin{aligned}\mathcal{Q}(R_n, S) &= -\sum_n \ln P(Y_n, Y_n^\dagger | R_n, S) \\ &= -\sum_n \left(\ln P(Y_n|R_n, S) - \ln P(Y_n^\dagger|R_n, S)\right) \\ &\sim \sum_n ||Y_n - R_n S||_2^2 + \sum_n ||Y_n^\dagger - R_n \mathcal{A} S||_2^2.\end{aligned} \tag{10}$$

This formulation relates to the classic structure from motion problems [31, 46], but those classic methods do not impose symmetry, and therefore, they have only the first term in Eq. (10), *i.e.*, $\mathcal{Q}(R_n, S) = \sum_n ||Y_n - R_n S||_2^2$. If all the data is fully observed then S is of rank 3 (assuming the 3D points do not lie in a plane or a line) and so the energy can be minimized by first stacking the keypoints for all the images together, *i.e.*, $\mathbf{Y} = [Y_1^T, ..., Y_N^T]^T \in \mathbb{R}^{2N \times P}$, then applying SVD to $\mathbf{Y}$. The solution is unique up to some coordinate transformations, *i.e.*, the rotation ambiguities [31, 46].

Remark 3 *Our problem is more complex than the classic rigid SfM because of two confounding issues: (i) some keypoints will be unobserved in each image, (ii) we cannot directly solve Eq.* (10) *by SVD because it consists of two energy terms which are not independent (even if all data is observed). Hence, we first formulate a full energy function with missing points, where the missing points are initialized in Section 5.3. After that, in Section 5.4, a surrogate energy is defined, which exploits symmetry and can be minimized by SVD, and therefore, can be used for initializing camera projection and the 3D structure.*

To deal with unobserved keypoints we divide them into *visible sets* VS, $VS^\dagger$ and *invisible sets* IVS, $IVS^\dagger$. Then the *full energy function* can be formulated as:

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

Here $\{Y_{n,p}, (n,p) \in IVS\}$, $\{Y_{n,p}^\dagger, (n,p) \in IVS^\dagger\}$ are the missing points.

Note that it is easy to add the Manhattan constraints in Eq. (11) as a regularization term for the Sym-RSfM on multiple images. But we found no significant improvement in our experiments when we used Manhattan, perhaps because it was not needed due to the extra images available as input. Please see the supplementary materials.

Algorithm 1: Optimization of the full energy Eq. (11).

```
Input: The stacked keypoint sets (for all the $N$
       images) $\mathbf{Y}$ and $\mathbf{Y}^\dagger$ with occluded points, in
       which each occluded point is set to $\mathbf{0}$ initially.
Output: The camera projection matrix $R_n$ for each
        image, the 3D structure $S$, and the keypoints
        with recovered occlusions $(\mathbf{Y})^t$ and $(\mathbf{Y}^\dagger)^t$.
Initialize the occluded points by Algorithm 2.
Initialize the each camera projection $R_n$ and the 3D
  structure $S$ by Algorithm 3.
repeat
    Update $S$ by Eq. (12) and update each $R_n$ (see the
      supplementary materials).
    Calculate the occluded points by Eq. (13), and
      update them in $Y_n, Y_n^\dagger$.
    Centralize the $Y_n, Y_n^\dagger$ by Eq. (14).
until Eq. (11) converge;
```

5.2. Optimization of The Full Energy Function

We now define a *coordinate descent* algorithm to estimate the 3D structure, the camera projection, and the missing data. This algorithm is not guaranteed to converge to the global minimum, so it will be initialized using the surrogate energy function, described in Section 5.4.

We use a *coordinate descent* method to optimize Eq. (11) by updating R_n, S and the missing points $\{Y_{n,p}, (n,p) \in IVS\}, \{Y_{n,p}^\dagger, (n,p) \in IVS^\dagger\}$ iteratively. Note that the energy in Eq. (11) *w.r.t* R_n, S, *i.e.*, when the missing points are fixed, is given in Eq. (10).

Firstly, we vectorize Eq. (10) and S to update it in matrix form by $\mathbb{S}$:

$$\mathbb{S} = \left(\textstyle\sum_{n=1}^N (G_n^T G_n + \mathcal{A}_P^T G_n^T G_n \mathcal{A}_P)\right)^{-1} \left(\textstyle\sum_{n=1}^N (G_n^T \mathbb{Y}_n + \mathcal{A}_P^T G_n^T \mathbb{Y}_n^\dagger)\right). \tag{12}$$

where $\mathbb{S} \in \mathbb{R}^{3P\times 1}, \mathbb{Y}_n \in \mathbb{R}^{2P\times 1}, \mathbb{Y}_n^\dagger \in \mathbb{R}^{2P\times 1}$ are vectorized $S, Y_n, Y_n^\dagger$, respectively. $G_n = I_P \otimes R_n$ and $\mathcal{A}_P = I_P \otimes \mathcal{A}$. $I_P \in \mathbb{R}^{P\times P}$ is an identity matrix.

Each R_n is updated under the nonlinear orthogonality constraints $R_n R_n^T = I$ similar to the idea in EM-PPCA [48]: 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 materials for the details.

From Eq. (11), the occluded points of $\mathbf{Y}$ and $\mathbf{Y}^\dagger$ (*i.e.*, the p-th point $Y_{n,p}$ and $Y_{n,p}^\dagger$) can be updated by minimizing: $\mathcal{Q}(Y_{n,p}, Y_{n,p}^\dagger) = \sum_{(n,p)\in IVS} ||Y_{n,p} - R_n S_p||_2^2 + \sum_{(n,p)\in IVS^\dagger} ||Y_{n,p}^\dagger - R_n \mathcal{A} S_p||_2^2$, which implies the update rule for the missing points:

$$Y_{n,p} = R_n S_p, \qquad Y_{n,p}^\dagger = R_n \mathcal{A} S_p, \tag{13}$$

where $(n,p) \in IVS$.

Note that we do not model the translation explicitly for the sake of consistency with the baseline method [46], where the translation is assumed to be eliminated by centralizing the data. However, since the occluded points have been updated iteratively in our method, we have to re-estimate the translation and re-centralize the data during each iteration. This can be done by:

$$Y_n \leftarrow Y_n - \mathbf{1}_{2P}^T \otimes t_n, \qquad Y_n^\dagger \leftarrow Y_n^\dagger - \mathbf{1}_{2P}^T \otimes t_n,$$
$$t_n = \sum_p (Y_{n,p} - R_n S_p + Y_{n,p}^\dagger - R_n \mathcal{A} S_p). \tag{14}$$

The algorithm to optimize Eq. (11) is summarized in Algorithm 1, in which the initialization of the missing points, the 3D structure and the camera projection, *i.e.*, Algorithms 2 and 3, will be discussed in the following sections.

5.3. Initialization of The Missing Data

In this section, the missing data is initialized by the whole input data ignoring symmetry. This will be used both for coordinate descent of the full energy and for applying singular value decomposition to the surrogate energy.

Let $\mathbf{Y} = [Y_1^T, ..., Y_N^T]^T, \mathbf{Y}^\dagger = [(Y_1^\dagger)^T, ..., (Y_N^\dagger)^T]^T \in \mathbb{R}^{2N\times P}$ are the stacked keypoints for all the images, and $\mathbf{R} = [R_1^T, ...R_N^T]^T \in \mathbb{R}^{2N\times 3}$ are the stacked camera projection. Thus, we have $\mathbf{Y}^{\text{All}} = [\mathbf{Y}, \mathbf{Y}^\dagger] = \mathbf{R}[S, \mathcal{A}S]$. It implies that $\mathbf{Y}^{\text{All}}$ has the same rank, namely 3, with $\mathbf{R}[S, \mathcal{A}S]$ given all the points of $[S, \mathcal{A}S]$ do not lie on a plane or a line. Therefore, rank 3 recovery can be used to initialize the missing points. Also, the same centralization as in the previous section has to be done after each iteration of the missing points, so as to eliminate the translations.

The occlusions initialization is shown in Algorithm 2.

5.4. The Surrogate Energy: Initialization of Camera Projection and 3D Structure

Remark 4 *We now define a surrogate energy function that exploits the symmetry constraints, which enables us to decompose the energy into two independent terms and leads to an efficient minimization algorithm using SVD.*

To construct the surrogate energy, we first change the coordinates to exploit symmetry, so that the problem breaks down into two independent energy terms. Since S and $S^\dagger$ are symmetric along x-axis, we can decompose S by:

$$\mathbf{L} = \frac{\mathbf{Y} - \mathbf{Y}^\dagger}{2} = \mathbf{R}\begin{bmatrix} x_1, & ..., & x_P \\ 0, & ..., & 0 \\ 0, & ..., & 0 \end{bmatrix} = \mathbf{R}^1 S_x,$$
$$\mathbf{M} = \frac{\mathbf{Y} + \mathbf{Y}^\dagger}{2} = \mathbf{R}\begin{bmatrix} 0, & ..., & 0 \\ y_1, & ..., & y_P \\ z_1, & ..., & z_P \end{bmatrix} = \mathbf{R}^2 S_{yz}, \tag{15}$$

where $\mathbf{R}^1 \in \mathbb{R}^{2N\times 1}, \mathbf{R}^2 \in \mathbb{R}^{2N\times 2}$ are the first single column and second-third double columns of $\mathbf{R}$, $S_x \in \mathbb{R}^{1\times P}, S_{yz} \in \mathbb{R}^{2\times P}$ are the first single row and second-third double rows of S, respectively. Equation (15) gives us the energy function on $\mathbf{R}, S$ to replace Eq. (10) into:

Algorithm 2: The initialization of the occluded points.

```
Input: The stacked keypoint sets (for all the $N$
       images) $\mathbf{Y}$ and $\mathbf{Y}^\dagger$ with occluded points, in
       which each occluded point is set to $\mathbf{0}$ initially.
       The number of iterations $T$ (default 10).
Output: The keypoints with initially recovered
        occlusions $(\mathbf{Y})^t$ and $(\mathbf{Y}^\dagger)^t$.
Set $t = 0$, initialize the occluded points ignoring
  symmetry by:
while $t < T$ do
    Centralize $\mathbf{Y}^{\text{All}} = [(\mathbf{Y})^t, (\mathbf{Y}^\dagger)^t]$ by Eq. (14).
    Do SVD on $\mathbf{Y}^{\text{All}}$ ignoring the symmetry, i.e.,
      $[\mathbf{A}, \Sigma, \mathbf{B}] = \text{SVD}\left(\mathbf{Y}^{\text{All}}\right)$.
    Use the first 3 component of $\Sigma$ to reconstruct the
      keypoints $(\mathbf{Y}^{\text{All}})^{\text{new}}$.
    Replace the occluded points in $(\mathbf{Y})^t$, $(\mathbf{Y}^\dagger)^t$ by
      these in $(\mathbf{Y}^{\text{All}})^{\text{new}}$ and set $t \leftarrow t + 1$.
end
```

$$Q(\mathbf{R}, S) = ||\mathbf{L} - \mathbf{R}^1 S_x||_2^2 + ||\mathbf{M} - \mathbf{R}^2 S_{yz}||_2^2. \tag{16}$$

This is essentially changing the coordinate system by rotating $\mathbf{Y}, \mathbf{Y}^T$ with 45° (except a scale factor of $\sqrt{2}$).

Remark 5 *We have decomposed the energy into two independent terms, and therefore, they can be solved separately by SVD up to some ambiguities. Then we will combine them to study and resolve the ambiguities. Note that we assume the occluded keypoints are replaced by the initialization described in the previous section.*

Equation (16) implies that we can estimate $\mathbf{R}^1, S_x$ and $\mathbf{R}^2, S_{yz}$ by matrix factorization on $\mathbf{L}$ and $\mathbf{M}$ independently up to ambiguities. Then we combine them to remove this ambiguity by exploiting the orthogonality constraints on each R_n: *i.e.*, $R_n R_n^T = I$. Applying SVD to $\mathbf{L}$ and $\mathbf{M}$ gives us estimates, *i.e.*, $(\hat{\mathbf{R}}^1, \hat{S}_x)$ of $(\mathbf{R}^1, S_x)$, and $(\hat{\mathbf{R}}^2, \hat{S}_{yz})$ of $(\mathbf{R}^2, S_{yz})$, up to ambiguities λ and B:

$$\mathbf{L} = \mathbf{R}^1 S_x = \hat{\mathbf{R}}^1 \lambda \lambda^{-1} \hat{S}_x, \quad \mathbf{M} = \mathbf{R}^2 S_{yz} = \hat{\mathbf{R}}^2 B B^{-1} \hat{S}_{yz}, \tag{17}$$

here $\mathbf{R}^1$ and $\mathbf{R}^2$ are the decomposition of the true projection matrix $\mathbf{R}$, *i.e.*, $\mathbf{R} = [\mathbf{R}^1, \mathbf{R}^2]$, and $\hat{\mathbf{R}}^1$ and $\hat{\mathbf{R}}^2$ are the output estimates from SVD. Equation (17) shows that there is a scale ambiguity λ between $\hat{\mathbf{R}}^1$ and $\mathbf{R}^1$, and a 2-by-2 matrix ambiguity $B \in \mathbb{R}^{2\times 2}$ between $\hat{\mathbf{R}}^2$ and $\mathbf{R}^2$.

Remark 6 *Next we show how to resolve the ambiguities λ and B. This is done by using the* orthogonality constraints, *namely $R_n R_n^T = I$.*

Observe from Eqs. (17) that the ambiguities (*i.e.*, λ and B) are the same for the projection matrices of all the images. In the following derivation, we analyze the ambiguity for the n'th image, *i.e.*, projection matrix R_n.

Using Eqs. (17), the true R_n can be represented by:

$$R_n = [R_n^1, R_n^2] = [\hat{R}_n^1, \hat{R}_n^2] \begin{bmatrix} \lambda, & \mathbf{0} \\ \mathbf{0}, & B \end{bmatrix} = \hat{R}_n \begin{bmatrix} \lambda, & \mathbf{0} \\ \mathbf{0}, & B \end{bmatrix} \tag{18}$$

where $R_n^1 \in \mathbb{R}^{2\times 1}$ and $R_n^2 \in \mathbb{R}^{2\times 2}$ are the first single column and second-third double columns of the true projection matrix R_n. $\hat{R}_n^1 \in \mathbb{R}^{2\times 1}$ and $\hat{R}_n^2 \in \mathbb{R}^{2\times 2}$ are the initial estimation of R_n^1 and R_n^2 from the matrix factorization.

Let $\hat{R}_n = [\hat{R}_n^1, \hat{R}_n^2] = \begin{bmatrix} \hat{r}_n^{1,1}, & \hat{r}_n^{1,2:3} \\ \hat{r}_n^{2,1}, & \hat{r}_n^{2,2:3} \end{bmatrix} \in \mathbb{R}^{2\times 3}$, imposing the orthogonality constraints $R_n R_n^T = I$ using Eq. (18) gives:

$$\begin{aligned} R_n R_n^T &= \hat{R}_n \begin{bmatrix} \lambda^2, & \mathbf{0} \\ \mathbf{0}, & BB^T \end{bmatrix} \hat{R}_n^T \\ &= \begin{bmatrix} \hat{r}_n^{1,1}, & \hat{r}_n^{1,2:3} \\ \hat{r}_n^{2,1}, & \hat{r}_n^{2,2:3} \end{bmatrix} \begin{bmatrix} \lambda^2, & \mathbf{0} \\ \mathbf{0}, & BB^T \end{bmatrix} \begin{bmatrix} \hat{r}_n^{1,1}, & \hat{r}_n^{1,2:3} \\ \hat{r}_n^{2,1}, & \hat{r}_n^{2,2:3} \end{bmatrix}^T = I \end{aligned} \tag{19}$$

Vectorizing BB^T of Eq. (19) using $\text{vec}(AXB^T) = (B \otimes A)\text{vec}(X)$, we can get the following linear equations:

$$\begin{bmatrix} (\hat{r}_n^{1,1})^2, & \hat{r}_n^{1,2:3} \otimes \hat{r}_n^{1,2:3} \\ (\hat{r}_n^{2,1})^2, & \hat{r}_n^{2,2:3} \otimes \hat{r}_n^{2,2:3} \\ \hat{r}_n^{1,1}\hat{r}_n^{2,1}, & \hat{r}_n^{1,2:3} \otimes \hat{r}_n^{2,2:3} \end{bmatrix} \begin{bmatrix} \lambda^2 \\ \text{vec}(BB^T) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}. \tag{20}$$

Note that BB^T is a symmetric matrix, the second and third elements of $\text{vec}(BB^T)$ are the same. Let $\text{vec}(BB^T) = [bb_1, bb_2, bb_2, bb_3]^T$, we can enforce the symmetriy of BB^T by rewriting Eq. (20):

$$\begin{bmatrix} (\hat{r}_n^{1,1})^2, & \hat{r}_n^{1,2:3} \otimes \hat{r}_n^{1,2:3} \\ (\hat{r}_n^{2,1})^2, & \hat{r}_n^{2,2:3} \otimes \hat{r}_n^{2,2:3} \\ \hat{r}_n^{1,1}\hat{r}_n^{2,1}, & \hat{r}_n^{1,2:3} \otimes \hat{r}_n^{2,2:3} \end{bmatrix} \begin{bmatrix} 1\,0\,0\,0 \\ 0\,1\,0\,0 \\ 0\,0\,1\,0 \\ 0\,0\,1\,0 \\ 0\,0\,0\,1 \end{bmatrix} \begin{bmatrix} \lambda^2 \\ bb_1 \\ bb_2 \\ bb_3 \end{bmatrix} = A_i\mathbf{x} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \tag{21}$$

where the constant matrix of the left term is a matrix operator to sum the third and forth columns of the coefficient matrix with r's, $\mathbf{x}$ is the unknown variables $[\lambda^2, bb_1, bb_2, bb_3]^T$, and $A_i \in \mathbb{R}^{3\times 4}$ is the corresponding coefficients.

Stacking all the $\hat{\mathbf{R}}$'s together, *i.e.*, let $\mathbf{A} = [A_1^T, ..., A_N^T]^T \in \mathbb{R}^{3N\times 4}$ and $\mathbf{b} = \mathbf{1}_N \otimes [1,1,0]^T$, we have a over-determined equations for the unknown $\mathbf{x}$: $\mathbf{Ax} = \mathbf{b}$ (*i.e.*, $3N$ equations for 4 unknowns), which can be solved efficiently by LSE: $\mathbf{x} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$.

Remark 7 *The ambiguity of* $\mathbf{R}^1$ *and* $\hat{\mathbf{R}}^1$, *(*i.e.*, in the symmetry direction), is just a sign change, which cause by calculating* λ *from* λ^2*. In other words, the symmetry direction can be fixed as* x*-axis in our coordinate system using the decomposition Eq. (15).*

Algorithm 3: The initialization of the camera projection and structure.

```
Input: The keypoint sets $\mathbf{Y}$ and $\mathbf{Y}^\dagger$ with initially
       recovered occluded points by Algorithm 2.
Output: The initialized camera projection $\mathbf{R}$ and the
        3D structure $S$.
1 Change the coordinates to decouple the symmetry
  constraints by Eq. (15).
2 Get $\hat{\mathbf{R}}^1, \hat{\mathbf{R}}^2, \hat{S}_x, \hat{S}_{yz}$ by SVD on $\mathbf{L}, \mathbf{M}$, i.e., Eq. (17).
3 Solve the squared ambiguities $\lambda^2$, $BB^T$ by Eq. (21).
4 Solve for $\lambda$ from $\lambda^2$, and $B$ from $BB^T$, up to sign and
  rotation ambiguities.
5 Obtain the initialized $\mathbf{R}$ and $S$ by Eq. (22).
```

After obtained BB^T, B can be recovered up to a rotation ambiguity on yz-plane, which does not affect the reconstructed 3D structure (See the supplementary materials).

Given $\lambda, B, \hat{\mathbf{R}}, \hat{S}$, we can get the true $\mathbf{R}$ and S by:

$$\mathbf{R} = \hat{\mathbf{R}} \begin{bmatrix} \lambda, & \mathbf{0} \\ \mathbf{0}, & B \end{bmatrix}, \qquad S = \begin{bmatrix} \lambda, & \mathbf{0} \\ \mathbf{0}, & B \end{bmatrix}^{-1} \hat{S}. \tag{22}$$

5.5. Experiments on The Symmetric Rigid Structure from Motion

We estimate the 3D structures of each subtype and the orientations of all the images within that subtype for *aeroplane, bus, car, sofa, train, tv* in Pascal3D+ [51]. Note that Pascal3D+ provides a single 3D shape for each subtype rather than for each object. For example, it provides 10 subtypes for the *car* category, such as *sedan, truck*, but ignores the within-subtype variation [14]. Thus, we divide the images of the same category into subtypes, and then input the images of each subtype for the experiments.

Following [29, 15], images with more than 5 visible keypoints are used. The rotation and shape errors are calculated by Eq. (1). The rigid SfM (RSfM) [46] and a more recent CSF method [17], which both do not exploit symmetry, are used for comparison. Note that the CSF method [17] utilized smooth time-trajectories as initialization, which does not always hold in our application, as the input images here are not from a continuous video. Thus, we also investigate the results from CSF method with random initialization. We report the CSF results with smooth prior as *CSF (S)* and the *best* results with 10 random initialization as *CSF (R)*.

The results (mean rotation and shape errors) are shown in Tables 1 and 2, which indicate that our method outperforms the baseline methods for most cases. The cases that our method does not perform as the best may be caused by that Pascal3D+ assumes the shapes from objects within the same subtype are very similar to each other, but this might be violated sometimes. Moreover, our method is robust to imperfect annotations (*i.e.*, result in imperfect symmetric pairs)

	aeroplane							bus					
	I	II	III	IV	V	VI	VII	I	II	III	IV	V	VI
RSfM	0.52	1.17	1.99	0.28	0.94	1.77	1.74	0.32	1.05	0.27	0.27	1.13	0.51
CSF (S)	0.92	0.70	0.87	0.83	**0.89**	0.96	**1.02**	0.72	0.86	0.68	0.92	0.94	1.04
CSF (R)	0.93	0.82	**0.80**	0.91	0.99	1.02	1.37	0.69	0.88	0.81	0.93	0.88	1.08
Sym-RSfM	**0.13**	**0.46**	2.00	**0.17**	1.81	**0.89**	1.69	**0.19**	**0.33**	**0.03**	**0.22**	**0.58**	**0.50**

	car										sofa	
	I	II	III	IV	V	VI	VII	VIII	IX	X	I	II
RSfM	0.58	0.71	0.54	1.10	0.67	1.51	0.67	1.41	0.97	0.37	1.18	0.75
CSF (S)	0.95	1.22	1.06	1.12	1.00	1.03	1.13	**1.04**	1.33	1.03	1.02	0.75
CSF (R)	0.95	1.32	1.08	1.06	1.09	0.98	1.22	1.05	1.29	1.17	0.85	0.76
Sym-RSfM	**0.36**	**0.43**	**0.30**	**0.43**	**0.31**	**0.26**	**0.32**	**1.04**	**0.25**	**0.16**	**0.67**	**0.26**

	sofa				train				tv			
	III	IV	V	VI	I	II	III	IV	I	II	III	IV
RSfM	1.90	1.00	1.99	1.90	1.95	1.44	1.33	1.01	0.86	0.38	0.39	1.38
CSF (S)	1.16	0.99	1.66	1.21	0.84	0.69	0.86	0.85	0.99	0.79	0.95	0.73
CSF (R)	0.87	0.86	**0.98**	1.70	0.92	**0.67**	**0.84**	**0.82**	1.00	0.82	0.91	0.84
Sym-RSfM	**0.11**	**0.69**	1.57	**0.97**	**0.18**	0.68	0.88	0.97	**0.23**	**0.14**	**0.26**	**0.44**

Table 1. The mean *rotation* errors for *aeroplane, bus, car, sofa, train, tv*, calculated using the images from the same subtype (denoted by the Roman numerals) as input.

	aeroplane							bus					
	I	II	III	IV	V	VI	VII	I	II	III	IV	V	VI
RSfM	0.44	1.17	0.52	**0.29**	0.76	0.54	0.61	1.29	1.27	1.16	0.97	1.52	1.21
CSF (S)	1.25	**0.36**	1.42	0.84	0.33	0.47	**0.59**	1.11	**0.39**	0.56	**0.16**	3.02	**0.44**
CSF (R)	0.25	0.44	0.34	1.40	0.58	1.73	0.69	0.99	1.06	1.33	0.88	1.94	2.00
Sym-RSfM	**0.19**	0.88	**0.27**	0.34	**0.33**	**0.30**	0.62	**0.68**	0.58	**0.35**	0.24	**0.76**	0.47

	car										sofa	
	I	II	III	IV	V	VI	VII	VIII	IX	X	I	II
RSfM	1.48	1.49	1.33	1.38	1.45	1.39	1.21	1.81	1.22	1.07	2.50	1.09
CSF (S)	1.06	2.33	1.15	1.17	1.36	1.17	1.03	1.10	2.03	0.99	1.78	0.24
CSF (R)	1.34	1.07	1.03	1.16	1.18	1.26	0.88	**0.90**	1.65	1.13	**0.76**	0.25
Sym-RSfM	**1.03**	**0.96**	**0.95**	**1.07**	**0.89**	**1.00**	**0.81**	1.66	**0.88**	**0.71**	2.27	**0.22**

	sofa				train				tv			
	III	IV	V	VI	I	II	III	IV	I	II	III	IV
RSfM	1.49	1.60	3.44	2.56	1.68	0.39	0.28	0.22	0.23	0.88	0.64	1.77
CSF (S)	3.14	1.54	2.74	1.55	0.83	0.85	0.25	0.26	0.66	0.77	0.34	0.34
CSF (R)	1.82	1.19	1.42	1.20	1.05	**0.37**	0.24	**0.17**	0.22	0.97	0.55	0.36
Sym-RSfM	**0.40**	**1.07**	**0.87**	**1.14**	**0.73**	0.61	**0.13**	0.24	**0.09**	**0.29**	**0.32**	**0.14**

Table 2. The mean *shape* errors for *aeroplane, bus, car, sofa, train, tv*, calculated using the images from the same subtype (denoted by the Roman numerals) as input.

for practical use. This was simulated by adding Gaussian noise to the 2D annotations in the supplementary material.

6. Conclusions

We show that symmetry, Manhattan and multiple-image cues can be utilized to achieve good quality performance on object 3D structure reconstruction. For the single image case, symmetry and Manhattan together are sufficient if we can identify suitable keypoints. For the multiple-image case, we formulate the problem in terms of energy minimization exploiting symmetry, and optimize it by a coordinate descent algorithm. To initialize this algorithm, we define a surrogate energy function exploiting symmetry, and decompose it into a sum of two independent terms that can be solved by SVD separately. We further study the ambiguities of the surrogate energy and show that they can be resolved assuming the orthographic projection. Our results outperform the baselines on most object classes in Pascal3D+. Future works involve using richer camera models, like perspective [42, 22], and keypoints extracted and matched automatically from images [8, 38, 36] with outliers [35, 37, 34] and occlusions [27, 26, 33] handled.

Acknowledgments. We would like to thank Ehsan Jahangiri, Wei Liu, Cihang Xie, Weichao Qiu, Xuan Dong, and Siyuan Qiao for giving feedbacks on this paper. This work is partially supported by ONR N00014-15-1-2356 and the NSF award CCF-1317376.

References

[1] A. Agudo, L. Agapito, B. Calvo, and J. Montiel. Good vibrations: A modal analysis approach for sequential non-rigid structure from motion. In *CVPR*, pages 1558–1565, 2014. 1

[2] I. Akhter, Y. Sheikh, and S. Khan. In defense of orthonormality constraints for nonrigid structure from motion. In *CVPR*, 2009. 2

[3] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Nonrigid structure from motion in trajectory space. In *NIPS*, 2008. 2, 3

[4] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Trajectory space: A dual representation for nonrigid structure from motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(7):1442–1456, 2011. 1, 2

[5] L. Bourdev, S. Maji, T. Brox, and J. Malik. Detecting people using mutually consistent poselet activations. In *ECCV*, 2010. 3

[6] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3d shape from image streams. In *CVPR*, 2000. 2

[7] D. Ceylan, N. J. Mitra, Y. Zheng, and M. Pauly. Coupled structure-from-motion and 3d symmetry detection for urban facades. *ACM Transactions on Graphics*, 33, 2014. 2

[8] X. Chen and A. L. Yuille. Articulated pose estimation by a graphical model with image dependent pairwise relations. In *NIPS*, pages 1736–1744, 2014. 1, 8

[9] J. M. Coughlan and A. L. Yuille. Manhattan world: Compass direction from a single image by bayesian inference. In *ICCV*, 1999. 1, 2

[10] J. M. Coughlan and A. L. Yuille. Manhattan world: Orientation and outlier detection by bayesian inference. *Neural Computation*, 15(5):1063–1088, 2003. 1, 2

[11] Y. Dai, H. Li, and M. He. A simple prior-free method for non-rigid structure-from-motion factorization. In *CVPR*, 2012. 1, 2, 3

[12] Y. Dai, H. Li, and M. He. A simple prior-free method for non-rigid structure-from-motion factorization. *International Journal of Computer Vision*, 107:101–122, 2014. 1, 2, 3

[13] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski. Manhattan-world stereo. In *CVPR*, 2009. 1, 2

[14] Y. Gao, J. Ma, and A. L. Yuille. Semi-supervised sparse representation based classification for face recognition with insufficient labeled samples. *IEEE Transactions on Image Processing*, 2017. 7

[15] Y. Gao and A. L. Yuille. Symmetry non-rigid structure from motion for category-specific object structure estimation. In *ECCV*, 2016. 1, 3, 7

[16] G. G. Gordon. Shape from symmetry. In *Proc. SPIE*, 1990. 1, 2

[17] P. Gotardo and A. Martinez. Computing smooth time-trajectories for camera and deformable shape in structure from motion with occlusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33:2051–2065, 2011. 1, 2, 3, 7

[18] E. Grossmann, D. Ortin, and J. Santos-Victor. Single and multi-view reconstruction of structured scenes. In *ACCV*, 2002. 2

[19] E. Grossmann and J. Santos-Victor. Maximum likehood 3d reconstruction from one or more images under geometric constraints. In *BMVC*, 2002. 2

[20] E. Grossmann and J. Santos-Victor. Least-squares 3d reconstruction from one or more views and geometric clues. *Computer Vision and Image Understanding*, 99(2):151–174, 2005. 2

[21] O. C. Hamsici, P. F. Gotardo, and A. M. Martinez. Learning spatially-smooth mappings in non-rigid structure from motion. In *ECCV*, pages 260–273, 2012. 1

[22] R. Hartley and R. Vidal. Perspective nonrigid shape and motion recovery. In *European Conference on Computer Vision*, pages 276–289. Springer, 2008. 8

[23] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. 1

[24] J. H. Hong and A. Fitzgibbon. Secrets of matrix factorization: Approximations, numerics, manifold optimization and random restarts. In *ICCV*, 2015. 2

[25] W. Hong, A. Y. Yang, K. Huang, and Y. Ma. On symmetry and multiple-view geometry: Structure, pose, and calibration from a single image. *International Journal of Computer Vision*, 60:241–265, 2004. 1, 2

[26] D. Jacobs. Linear fitting with missing data: Applications to structure-from-motion and to characterizing intensity images. In *CVPR*, pages 206–212. IEEE, 1997. 8

[27] D. W. Jacobs. Linear fitting with missing data for structure-from-motion. *Computer Vision and Image Understanding*, 82(1):57–81, 2001. 8

[28] C. Jiang, C. Tang, A. Vaxman, P. Wonka, and H. Pottmann. Polyhedral patterns. *ACM Transactions on Graphics*, 34(6), 2015. 1

[29] A. Kar, S. Tulsiani, J. Carreira, and J. Malik. Category-specific object reconstruction from a single image. In *CVPR*, 2015. 1, 2, 3, 7

[30] L. L. Kontsevich. Pairwise comparison technique: a simple solution for depth reconstruction. *JOSA A*, 10(6):1129–1135, 1993. 1, 2

[31] L. L. Kontsevich, M. L. Kontsevich, and A. K. Shen. Two algorithms for reconstructing shapes. *Optoelectronics, Instrumentation and Data Processing*, 5:76–81, 1987. 2, 5

[32] Y. Li and Z. Pizlo. Reconstruction of shapes of 3d symmetric objects by using planarity and compactness constraints. In *Proc. of SPIE-IS&T Electronic Imaging*, 2007. 1, 2

[33] Z. Lin, M. Chen, and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint arXiv:1009.5055*, 2010. 8

[34] J. Ma, W. Qiu, J. Zhao, Y. Ma, A. L. Yuille, and Z. Tu. Robust l2e estimation of transformation for non-rigid registration. *IEEE Trans. Signal Processing*, 63(5):1115–1129, 2015. 8

[35] J. Ma, J. Zhao, Y. Ma, and J. Tian. Non-rigid visible and infrared face registration via regularized gaussian fields criterion. *Pattern Recognition*, 48(3):772–784, 2015. 8

[36] J. Ma, J. Zhao, J. Tian, X. Bai, and Z. Tu. Regularized vector field learning with sparse approximation for mismatch removal. *Pattern Recognition*, 46(12):3519–3532, 2013. 8

[37] J. Ma, J. Zhao, J. Tian, A. L. Yuille, and Z. Tu. Robust point matching via vector field consensus. *IEEE Transactions on Image Processing*, 23(4):1706–1721, 2014. 8

[38] J. Ma, H. Zhou, J. Zhao, Y. Gao, J. Jiang, and J. Tian. Robust feature matching for remote sensing image registration via locally linear transforming. *IEEE Transactions on Geoscience and Remote Sensing*, 53(12):6469–6481, 2015. 8

[39] D. D. Morris, K. Kanatani, and T. Kanade. Gauge fixing for accurate 3d estimation. In *CVPR*, 2001. 2

[40] D. P. Mukherjee, A. Zisserman, and M. Brady. Shape from symmetry: Detecting and exploiting symmetry in affine images. *Philosophical Transactions: Physical Sciences and Engineering*, 351:77–106, 1995. 1, 2

[41] S. I. Olsen and A. Bartoli. Implicit non-rigid structure-from-motion with priors. *Journal of Mathematical Imaging and Vision*, 31(2-3):233–244, 2008. 2

[42] H. S. Park, T. Shiratori, I. Matthews, and Y. Sheikh. 3d trajectory reconstruction under perspective projection. *International Journal of Computer Vision*, 115(2):115–135, 2015. 8

[43] J. Rosen. *Symmetry discovered: Concepts and applications in nature and science*. Dover Publications, 2011. 1

[44] P. H. Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31:1–10, 1966. 3

[45] S. Thrun and B. Wegbreit. Shape from symmetry. In *ICCV*, 2005. 1, 2

[46] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992. 2, 4, 5, 6, 7

[47] L. Torresani, A. Hertzmann, and C. Bregler. Learning non-rigid 3d shape from 2d motion. In *NIPS*, 2003. 1

[48] L. Torresani, A. Hertzmann, and C. Bregler. Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30:878–892, 2008. 1, 2, 5

[49] T. Vetter and T. Poggio. Symmetric 3d objects are an easy case for 2d object recognition. *Spatial Vision*, 8:443–453, 1994. 1, 2

[50] S. Vicente, J. Carreira, L. Agapito, and J. Batista. Reconstructing pascal voc. In *CVPR*, 2014. 1, 2

[51] Y. Xiang, R. Mottaghi, and S. Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *WACV*, 2014. 2, 4, 7

[52] J. Xiao, J. Chai, and T. Kanade. A closed-form solution to nonrigid shape and motion recovery. In *ECCV*, 2004. 1, 2