Structural Symmetries from Motion for Scene Reconstruction and Understanding

Natesh Srinivasan
natesh.srinivasan@gatech.edu
Luca Carlone
luca.carlone@gatech.edu
Frank Dellaert
dellaert@cc.gatech.edu

College of Computing
Georgia Institute of Technology
Atlanta, GA, USA

Abstract

The identification and description of partial symmetries in man-made structures is a powerful tool to improve the quality of 3D reconstruction from unordered images and to enable high-level understanding of scene geometry. In this work we propose an approach to identify symmetries and exploit them in Structure from Motion (SfM). Our first contribution is a symmetry detection approach that uses the 3D geometry of the scene as well as 2D appearance clues. We show that a particular parametrization of the *transformation space* (space in which each point represents a candidate symmetry relation) exposes the dominant symmetries in the scene. Then, we use appearance information to prune incorrect symmetry hypotheses. The second contribution is a *constrained bundle adjustment* (CBA) scheme that jointly optimizes for the best 3D reconstruction and the symmetry *generators*. Contrarily to related work on CBA, our approach models *n*-fold (rotational and translational) repetitions of architectural elements, and allows estimating a *generative model* of the 3D geometry. Experimental results confirm that our method can correctly identify and exploit partial symmetries in noisy and sparse SfM datasets.

1 Introduction

Several man-made 3D structures exhibit some sort of *partial symmetry*. The concept of partial symmetry can be informally understood as the presence of an architectural element (e.g., a window, a column), that is repeated regularly within the 3D structure (a formal definition is given in Section 2). Detecting and describing the symmetric elements is crucial for high-level understanding of the 3D geometry (e.g, shape classification), to reduce model complexity, or to edit the 3D model in a consistent manner [32]. Moreover, symmetries can inform SfM and improve the quality of 3D reconstruction from an unordered set of images [6, 7].

While existing literature offers well established approaches for symmetry detection from 3D models (e.g., point cloud, mesh) these techniques assume the model to be dense and geometrically correct [32]. Therefore, direct application to the noisy and sparse point clouds resulting from SfM remains challenging. This observation triggered recent interest towards joint solutions to detect symmetries and exploit them in SfM [6, 7]. While these works provide excellent contributions towards the goal of symmetry-aware SfM algorithms, they rely on strong assumptions (e.g., geometric priors [7]) or require manual annotation [6].

Pages 136.1-136.13
DOI: https://dx.doi.org/10.5244/C.29.136

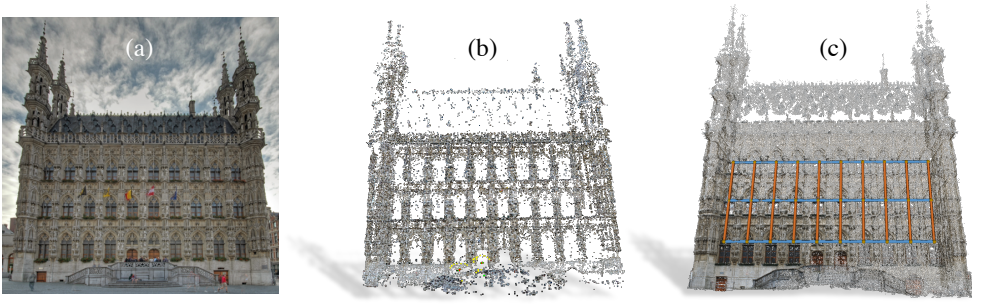

Figure 1: The proposed approach uses (a) a set of 2D images, and (b) a sparse 3D reconstruction to detect repetitive patterns (regular structure) and exploits them for SfM. (c) Generative model of the regular structure obtained from the proposed approach, for the Leuven dataset.

Related work. Our literature review is organized according to the type of input data fed to the symmetry detection approaches. We distinguish *2D*, *3D*, and *2D-3D* approaches.

2D approaches search for repeated patterns in a single image, and can be classified in *local* and *global* approaches. Local approaches hypothesize symmetry generators from pairwise matches and then extract dominant symmetries using voting schemes [28, 38], greedy or exhaustive search [22, 26, 42], vanishing points [48], or invariance-driven techniques [11]. Global approaches look for periodic texture over the entire image, using the Fourier transform [21, 36, 45], Hough transform [6, 46, 50], autocorrelation [23, 25], tiling theory [8], and moment-based methods [12]. Related work also uses 2D symmetries for 3D reconstruction from a single image [14, 18, 19, 33], or to obtain procedural models [34, 49].

3D approaches look for symmetries in 3D models. Also in this case it is possible to distinguish global and local approaches. Global approaches use moment-based methods [29, 47], correlation of the extended Gaussian image [44], spherical harmonics [30], Fourier transform [9, 20], and multidimensional scaling [39]. Local approaches include geometric hashing [10], voting schemes [31, 35], the reflective symmetry transform [37], graph-based approaches [3], and spectral methods [24]. We refer to [27, 32] for a comprehensive review.

2D-3D approaches detect symmetries from a set of images picturing a 3D scene. We call them "2D-3D" as most of these approaches exploit both 2D images and a 3D reconstruction, obtained via standard SfM; this initial 3D reconstruction has to discard false matches corresponding to symmetric elements [40, 51]. 3D Symmetries do not necessarily produce regular patterns under perspective projection and this prevents direct use of 2D approaches [17]. Moreover, 3D approaches perform poorly on the models produced via SfM, which are noisy and incomplete [5]. Jiang *et al.* [16] propose to rectify the 3D geometry to a surface and use 2D lattice detection to find repeated elements. Cohen *et al.* [7] look for inconsistencies between the essential matrix (estimated from feature matches) and the geometry of the scene to detect symmetries, and use them for *constrained bundle adjustment* (CBA). Ceylan *et al.* [5] focus on translational symmetry and detect repetitions in 2D images, with a user-guided procedure. Ceylan *et al.* [4] use line features for symmetry-aided 3D reconstruction.

Contribution. We propose an approach for 2D-3D symmetry detection and we show how to leverage the presence of repetitive structure to improve SfM reconstruction. Our approach includes three building blocks. The first is a multi-hypothesis estimator for the 3D symmetry generators. We borrow key insights from [35], which shows how to map putative symmetry relations into a suitable transformation space. However, we skip grid fitting (which is unreliable on SfM data), and we show that a polar parametrization of the transformation space clearly exposes dominant symmetries. The second block prunes the multiple hypotheses on the generators and returns the generators that are most consistent with 2D

appearance. This is similar to [17], while we avoid clustering and 3D surface fitting. Finally, the last block takes the estimate for the generators and jointly refines this estimate and the 3D reconstruction. Contrarily to standard CBA we model n-fold translational and rotational repetitions, we avoid priors [7] and user intervention [4], and we output a *generative model*.

2 Preliminaries on Symmetry

An object $\mathcal{O}$ is *symmetric* with respect to a transformation T if $\mathcal{O} = T(\mathcal{O})$, i.e., the object is invariant under the action of T. This notion is also referred to as a *global* symmetry. In many practical cases, only part of the object exhibits a repetitive structure, which does not correspond to a symmetry in the whole object. For instance, the translational symmetry of the windows in Fig. 1a does not correspond to a translational symmetry of the overall building. The notion of *partial* symmetry [31, 32] captures these local similarities. An object $\mathcal{O}$ exhibits partial symmetry with respect to a transformation T if there exist two non-overlapping subsets $\mathcal{S}_0, \mathcal{S}_1 \subseteq \mathcal{O}$ that are such that $\mathcal{S}_1 = T(\mathcal{S}_0)$.

While the notion of partial symmetry implies the presence of a single repetition ($\mathcal{S}_1$) of a basic element ($\mathcal{S}_0$), it is often the case that multiple copies of a single element occur in the 3D model. This phenomenon is captured by the notion of *regular structure* [35]. A *K-parameters regular structure* is a subset $\mathcal{R} \subseteq \mathcal{O}$ that can be completely reconstructed given a *representative element* $\mathcal{E} \subseteq \mathcal{R}$, a set of K *generators* $g_1, \ldots, g_k$ and a set of integer dimensions $n_1, \ldots, n_k$. For instance, the lateral facade of the Neptune temple in Fig. 4e-f exhibits a 1-parameters regular structure, as the facade can be reconstructed given the first column (the representative element $\mathcal{E}$), a vector describing the direction of the repetition (the *generator* g) and the number of repetitions (the integer dimension n). The last column is an *n-fold repetition* of the representative element. Another example is the facade of the Leuven Cathedral in Fig. 1c, which exhibits a 2-parameter regular structure as a representative element (a window) is repeated both along the horizontal direction (generator g_1) and the vertical direction (generator g_2). The tuple $(\mathcal{E}, \{g_1, \ldots, g_K\}, \{n_1, \ldots, n_K\})$ is called a *generative model* [35].

3 Structural Symmetries from Motion

We are given a set of images picturing a 3D scene and our goal is to detect regular structures and possibly exploit them to obtain an accurate 3D model. As in related work [7, 17], we start by creating a sparse point cloud $\mathcal{P}$ from the images, using an SfM technique that discards false matches resulting from symmetries. Our approach computes a set of hypotheses for the generators of the regular structure using the sparse 3D model (Section 3.2). Then incorrect generators are pruned using appearance information (Section 3.3). Finally, the generators and the 3D model are jointly refined (Section 3.4). Before introducing our first contribution (Section 3.2), Section 3.1 recalls the notion of transformation space from [35].

3.1 Feature Extraction and Transformation Space

This section describes how to create a set of putative symmetry transformations and represent them in a suitable transformation space. We first detect 3D features on the point cloud $\mathcal{P}$. While related work obtains these features by uniform sampling [35], we use 3D SIFT keypoints [41], and obtain N points $s_1, \ldots, s_N \in \mathbb{R}^3$ (an example is given in [1, Fig. 3]). For each feature s_i, we associate a set of neighboring points $\mathcal{P}_i$, which are within a ball of radius $\bar{r}_s$ from s_i. Then, for each pair (i, j) of features, we compute a relative transformation by applying ICP to the patches $\mathcal{P}_i$ and $\mathcal{P}_j$. This gives a set of putative transformations $\mathcal{T} \doteq \{T_{ij} : i, j = 1, \ldots, N\}$. Each T_{ij} describes a rigid transformation and can be written as $T_{ij} \doteq (R_{ij}, t_{ij})$, where $R_{ij} \in \mathrm{SO}(3)$ and $t_{ij} \in \mathbb{R}^3$. The set $\mathcal{T}$ contains many outliers (pairs (i, j)

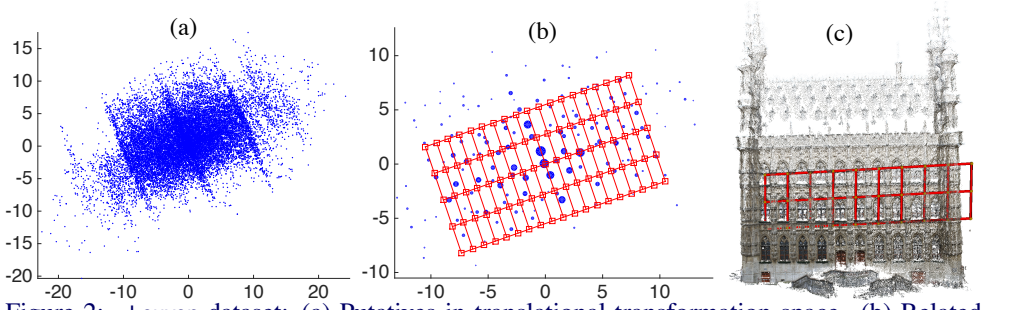

Figure 2: Leuven dataset: (a) Putatives in translational transformation space. (b) Related works apply clustering, followed by grid fitting (e.g., [35]), to infer the regular structure from the grid. (c) Since the 3D model is noisy and sparse, grid fitting performs poorly.

that are not related by a symmetry relation) and, even for the pairs that belong to the regular structure, the transformation T_{ij} is produced by a composition of an unknown number of unknown generators $g_1, \ldots, g_k$. To make the problem more tractable and since we are mainly interested in symmetries in urban architecture, we focus on the following regular structures:

- 1-parameter regular structures: these have a single generator g_1. We consider structure with $g_1 \in \mathrm{SO}(3)$ (rotational symmetry) or $g_1 \in \mathbb{R}^3$ (translational symmetry).
- 2-parameter regular structures: these have two generators g_1 and g_2. Without considering scaling, there are only two admissible set of generators [35]: (i) $g_1 \in \mathbb{R}^3$, $g_2 \in \mathbb{R}^3$ (translation along two independent directions as in Fig. 1c), and (ii) $g_1 \in \mathrm{SO}(3)$, $g_2 \in \mathbb{R}^3$ (rotation along an axis u, and translation parallel to u, as in Fig. 5a).

As in [35], we partition the set of putative transformations $\mathcal{T}$ into two subsets: the subset $\mathcal{T}_t$ of transformations having almost null rotation, and the subset $\mathcal{T}_R$ of transformations having a significant rotation component (i.e., rotation angle greater than a threshold ε_R).

The putatives in $\mathcal{T}_t$ describe pure translations, hence we can visualize them as points in $\mathbb{R}^3$. This set has to contain any evidence of translational symmetries. The seminal work [35] shows that, when considering 1 and 2-parameter regular structures, the translations defining the regular structures lie in a subspace of $\mathbb{R}^3$ (intuitively, the 2 generators of a 2-parameter regular structure only span a plane in 3D space). Therefore, according to [35] we re-parametrize the (translational) transformation space as a 2D space[1], and we write each putative transformation as a 2D vector $x_{ij} \in \mathbb{R}^2$. The set of these putatives, $\mathcal{T}_t^{2d} \doteq \{x_{ij}\}$, is called the *translational transformation space*. An example of $\mathcal{T}_t^{2d}$ is given in Fig. 2a.

Similarly, the set $\mathcal{T}_R$ has to contain evidence of rotational symmetry. A mild assumption (when dealing with urban architecture) is that the axis of the rotational symmetry is vertical. Hence we select the subset of $\mathcal{T}_R$ having rotation axis close to the vertical, and we store the corresponding rotation angles. This allows us to reparametrize the putatives as a set of rotation angles $\phi_{ij} \in (-\pi, +\pi]$. We call $\mathcal{T}_R^{1d} \doteq \{\phi_{ij}\}$ the *rotational transformation space*.

While the creation of the transformation space leverages the elegant formulation of [35], we follow a different approach to "explore" this transformation space, searching for the symmetry generators (Section 3.2). To motivate our approach, we now briefly discuss issues arising from the direct application of grid fitting [35] to estimate the generators from the SfM reconstruction. Pauly *et al.* [35] propose to first perform clustering, to expose recurrent transformations; the result of clustering for the Leuven dataset is shown in Fig. 2b in blue. Then, they fit a regular lattice using nonlinear optimization. The issue with this approach is that the result of clustering is very cluttered when dealing with SfM data; moreover, the

[1] This reparametrization requires to fit a plane to the 3D points in $\mathcal{T}_t$ (using RANSAC), and then projecting the resulting inliers to the estimated plane [35]. Fig. 7 in [35] provides an intuitive description.

optimization problem underlying grid fitting is nonconvex and has many local minima, hence it can easily converge to an incorrect estimate of the generators (Fig. 2c). In the following section we propose an alternative approach that avoids clustering and grid fitting.

3.2 Generator Hypotheses using the Polar Transformation Space

Let us start our discussion with the translational transformation space $\mathcal{T}_t^{2d}$. The key insight of our approach is the following: the presence of the repetitive structure in the 3D model implies that many of the transformations $x_{ij} \in \mathcal{T}_t^{2d}$ will be (approximately) in the form $x_{ij} = n \cdot g_k$ ($k \in \{1,2\}$) i.e., will be produced by an n-fold repetition along the generator $g_k \in \mathbb{R}^2$ (this is also one of the motivations for the grid fitting of Fig. 2b). Moreover, the 2-vector g_k can be written as $g_k = \delta_k \cdot u_k$, where $\delta_k \in \mathbb{R}$ is the *repetition period* and the unit vector $u_k \in \mathbb{R}^2$ ($\|u_k\| = 1$) is the *repetition direction*. Therefore, all the putatives generated by g_k become $x_{ij} = (n\delta_k) \cdot u_k$, i.e., they share the same direction, while they possible have different norms.

In order to exploit this insight, we propose to re-parametrize the transformation space $\mathcal{T}_t^{2d}$ in polar coordinates, i.e., we write each $x_{ij} \in \mathcal{T}_t^{2d}$ as an angle/distance pair (θ_{ij}, ρ_{ij}). We call the resulting set of pairs $\{(\theta_{ij}, \rho_{ij})\}$ the *polar* transformation space (PTS). All the putatives corresponding to repetitions along the generator g_k, which we wrote in Cartesian coordinates as $(n\delta_k) \cdot u_k$, can be expressed in polar coordinates as $(\theta_k, n\delta_k)$, where θ_k is the angle between the direction u_k and the horizontal axis; these putatives exhibit the same angle θ_k, but possibly different distances. Therefore, we can expose the dominant symmetry directions from an histogram plot of the angles $\{\theta_{ij}\}$; an example for the Leuven dataset is given in Fig. 3a1. The peaks correspond to directions that occur in many putative pairs, and these are the most likely to capture a generator direction. The histogram in Fig. 3a1 is obtained from the raw transformations in $\mathcal{T}_t^{2d}$ without preprocessing (we do not apply any type of clustering or filtering). The 3D directions corresponding to the peaks of the histogram are shown in Fig. 3b1. Note that the polar histogram also gives a clear picture of the *number* of directions (we do not know a-priori if the repetitions are along 1 or 2 directions). For instance, for the Neptune dataset (Fig. 3b2), the lateral facade of the temple only contains a 1-parameter regular structure, and this is correctly captured by the histogram of Fig. 3a2, which has a single prominent peak. In summary, from the angle histogram we can infer the number of translational generators (this is the "K" in the definition of K-parameter regular structure) and the directions u_k ($k = 1,\ldots,K$), which can be computed from the peaks.

Now, to completely characterize the generators $g_k = \delta_k \cdot u_k$, we need to compute the period δ_k, $k = 1,\ldots,K$. We argue that this is the difficult part of the symmetry detection problem: even the putatives corresponding to the regular structure provide a guess on δ_k only up to an unknown integer. We propose to use our polar representation also for the computation of the period. For each dominant direction u_k (or, equivalently, θ_k), we select the points that are along this direction (in polar coordinates, these are the points (θ_{ij}, ρ_{ij}) with $|\theta_{ij} - \theta_k| \leq \varepsilon_\theta$, where ε_θ is a threshold); then we compute the histogram of the distances ρ_{ij} associated to these points. Examples of distance histograms are given in Fig. 3c1-c2, and in [1, Fig. 4- 6]. Unfortunately, the distance histograms are not as clean as the angular ones (Fig. 3a1-a2). This is due to both the presence of outliers and the fact that a single generator creates multiple peaks, spaced at regular intervals (1-fold repetitions, 2-fold, etc.). The red crosses in Fig. 3c1-c2 denote the 10 highest peaks, selecting the peaks that have distance at least 0.3m.[2] From our experience, selecting the largest peak can lead to erroneous period estimate. While we could attempt fitting a 1D lattice to match the peaks in Fig. 3c1-c2, we

[2]The selection of the peaks can be done using the standard Matlab function `findpeaks`. From this selection, we remove the peak at the origin, which is always present, but uninformative (the putatives at the origin correspond to transformations that map each patch $\mathcal{P}_i$ to itself).

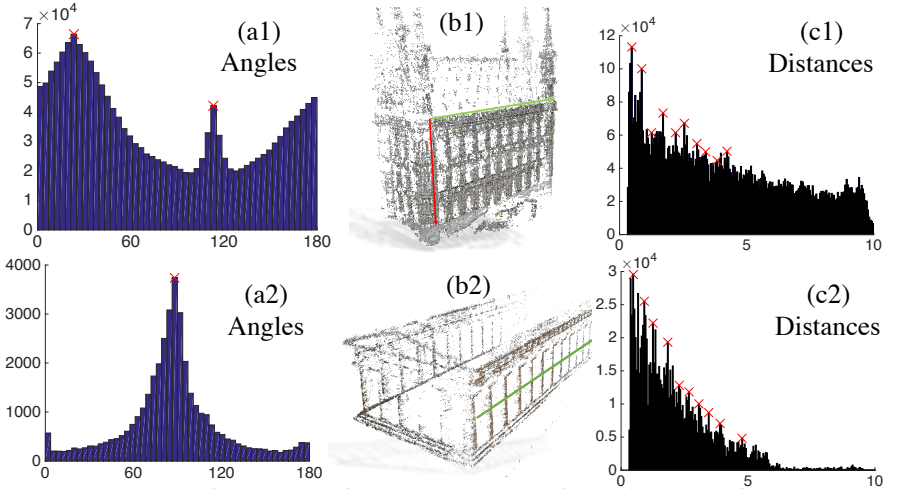

Figure 3: Histogram of the angles θ_{ij} in the polar transformation space for the Leuven (a1) and for the Neptune (a2) datasets. Corresponding 3D directions are visualized in (b1) and (b2), respectively. (c1,c2) Histograms of the distances ρ_{ij} for the points in the transformation space having angles θ_{ij} close to the highest peaks in (a1) and (b1), respectively.

prefer to return multiple hypotheses for the periods, corresponding to the m largest peaks. Then we use appearance information to prune incorrect period hypotheses (Section 3.3).

In summary, the proposed PTS allows computing the number of generators K, the direction of each generator u_k, and m hypothesis for the repetition period $\delta_k^1, \ldots, \delta_k^m$. Hence we build m hypotheses for each generator as $g_k^j = \delta_k^j \cdot u_k$, $j = 1, \ldots, m$. We argue that the advantage of our approach over grid fitting is that we identify a subproblem that is easy to solve (estimating K and the symmetry directions), and a subproblem that is hard (estimating the period), but can be dealt with independently for each direction and reduces to a 1D problem.

We conclude this section by commenting on the detection of rotational repetitions. In Section 3.1 we discussed how to obtain the 1D rotational transformation space $\mathcal{T}_R^{1d} = \{\phi_{ij}\}$. Similarly to the distance histogram of Fig. 3c1, the histogram of the angles in $\mathcal{T}_R^{1d}$, e.g., the one in Fig. 5c, has multiple peaks spaced at regular intervals, corresponding to n-fold rotational repetitions. Our approach retains the highest m peaks, corresponding to m hypotheses for the rotational symmetry, which are then pruned as discussed in Section 3.3.

3.3 Appearance-based Generator Ranking

As discussed in Section 3.2, our approach uses the geometry of the 3D model to produce multiple hypotheses for the generators of the regular structure. In this section we propose to prune these hypotheses using appearance information from the images. The intuition is that a good generator is one that maps a patch to another one having similar appearance (this is essentially the concept or regular structure). Since each point in the 3D model is obtained via SfM, it has a corresponding feature descriptor. Therefore, the appearance check reduces to verifying that the generator maps a point to another point that has a similar descriptor.

We use this insight to devise a generator ranking scheme. For each generator hypothesis g_k^j we do the following. We consider each point p_0 in the 3D model and we call d_0 the corresponding descriptor. We apply the transformation encoded in the generator g_k^j and get a second point p_1. Then we look in a ball of radius $\bar{r}_a$ around p_1 and we check all the descriptors d_1 corresponding to points in this ball. If at least one d_1 is close enough to the original descriptor d_0 (angle between the descriptors smaller than a threshold $\bar{d}$), then we

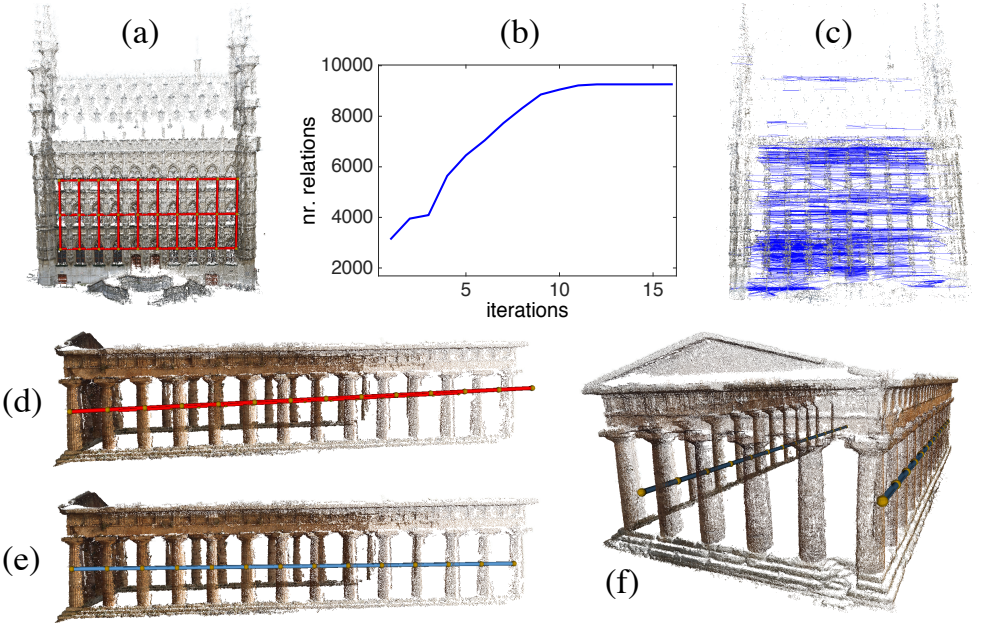

Figure 4: Leuven dataset: (a) initial estimate for the generators (before CBA); (b) number of symmetry relations discovered during the CBA iterations; (c) symmetry relations shown as lines connecting points considered n-fold repetitions of each other. Neptune dataset: (d) initial estimate of the generators, (e,f): refined generator estimate from CBA.

say that p_0 *agrees* with the generator g_k^j. Repeating this process for each point in the cloud we have a score for g_k^j, which is simply the number of points that agrees with the generator. Therefore, we select the best generator as the one having highest score. Scores for the hypotheses in the Leuven and Neptune datasets are reported in [11, Figs. 7- 8]. We remark that considering a ball around the transformed point p_1 allows us to mitigate the impact of noise in the generator estimates as well as the sparsity of the point cloud.

3.4 Bundle adjustment with Regular Structures

In Section 2 we recalled the concept of *generative model* $(\mathcal{E}, \{g_1, \ldots, g_K\}, \{n_1, \ldots, n_K\})$, which is a compact representation of a regular structure. In Section 3.2-3.3 we showed how to compute K and obtain an estimate of the generators $\{g_1, \ldots, g_K\}$. In this section we discuss how to jointly refine the point cloud and the generator estimates, and we hint to a simple approach to estimate the number of repetitions $\{n_1, \ldots, n_K\}$, and the representative element $\mathcal{E}$. For the joint refinement we propose an iterative strategy in which, at each iteration we optimize the point cloud and we try to discover new symmetries. As discussed in the following, we alternate two steps: *n-fold repetition discovery* and *regular structure refinement*.

n-fold repetition discovery. The first step is to discover n-fold repetitions, for each of generator g_k; this step shares the same idea of Section 3.3. We consider each point p_0 in the 3D model and we call d_0 the corresponding descriptor. Then we apply the transformation encoded in the generator g_k and get a second point p_1. Inside a ball of radius $\bar{r}_a$ around p_1, we look for the point having descriptor which is closest to d_0. If the closest descriptor is within a distance $\bar{d}$ from d_0, then we establish a 1-fold symmetry relation between p_0 and p_1, encoded by the tuple $(p_0, p_1, g_k, 1)$ (the 1 stands for 1-fold); we store the tuple in a set $\mathcal{S}$, called *the symmetry relations set*. We repeat the same procedure for p_1, trying to establish

another 1-fold relation with a third point p_2. If this succeeds we add the tuple $(p_0, p_2, g_k, 2)$ to $\mathcal{S}$. We iterate this procedure until no more repetition satisfies the check on the descriptor. As a result, we populate the set $\mathcal{S}$, which describes the discovered n-fold repetitions.

Regular structure refinement. Now we show how to augment bundle adjustment (BA) to encode rotational and translational regular structures. Standard BA [13] estimates the position of a set of points $\{p_j\}$ and the parameters (pose, calibration) of a set of cameras $\{C_i\}$ from measurements $\mathcal{Z}$; each measurement $z_{ij} \in \mathcal{Z}$ represents a noisy projection of point j to camera i. The estimates are computed by solving an optimization problem:

$$\min_{\{p_j\},\{C_i\}} E_{BA}(\{p_j\},\{C_i\}) = \min_{\{p_j\},\{C_i\}} \sum_{z_{ij}\in\mathcal{Z}} f(\pi(C_i, p_j) - z_{ij}) \tag{1}$$

where $\pi(C_i, p_j)$ is a function projecting point p_j to camera C_i, while $f(\cdot)$ is a loss function.

Our goal is to augment problem (1) to encode structural constraints. Contrarily to existing approaches [2, 5, 7], that consider only translational symmetries or 1-fold repetitions, we model n-fold rotational and translational repetitions. These repetitions are encoded in the set $\mathcal{S}$, which contains tuples in the form (p_j, p_l, g_k, n). We partition the set of symmetries $\mathcal{S}$ into two subsets $\mathcal{S}_t$ and $\mathcal{S}_R$, the former containing only translational symmetries, the second only rotational symmetries (we can easily distinguish them by looking at the generator in the tuple). We write the generators in $\mathcal{S}_t$ as $g_k = t_k$ (pure translation), and the generators in $\mathcal{S}_R$ as $g_k = R_k$ (pure rotation). Therefore, we augment the BA cost as follows:

$$\min_{\{p_j\},\{C_i\},\{t_k\},\{R_k\}} E_{BA}(\{p_j\},\{C_i\}) + E_t(\{p_j\},\{C_i\},\{t_k\}) + E_R(\{p_j\},\{C_i\},\{R_k\}) \tag{2}$$

with:

$$E_t(\{p_j\},\{C_i\},\{t_k\}) = \sum_{(p_j,p_l,t_k,n)\in\mathcal{S}_t} \sum_{i\in\mathcal{C}_l} f(\pi(C_i, p_j + n t_k) - z_{il})) \tag{3}$$

$$E_R(\{p_j\},\{C_i\},\{R_k\}) = \sum_{(p_j,p_l,R_k,n)\in\mathcal{S}_R} \sum_{i\in\mathcal{C}_l} f(\pi(C_i, R_k^n p_j) - z_{il})) \tag{4}$$

where $\mathcal{C}_l$ is the set of cameras observing point p_l. Intuitively, for every point l that is considered an n-fold repetition of another point j, beside minimizing the standard reprojection error $\pi(C_i, p_l) - z_{il}$ (included in E_{BA}), we also minimize a corresponding term in which the p_l is written as an n-fold repetition of p_j. For translational repetitions, we indeed substitute p_l with $p_j + n t_k$, while for rotational repetitions we write $R_k^n p_j$ (p_j rotated n times).

We optimize problem (2) using the Levenberg-Marquardt method, which outputs an improved estimate for the 3D points (and the cameras) and a refined estimate for the generators; analytic expressions of the involved Jacobians are given in [1, Section 7]. Since the optimization refined the 3D model, it possibly exposed other points belonging to the regular structure, hence we repeat the n-fold repetition discovery described earlier in this section, trying to expand the set of symmetry relations $\mathcal{S}$. Iterating symmetry discovery and bundle adjustment we converge to a set of symmetry relations $\mathcal{S}$, and an improved estimate for the generators of the regular structure. Indeed, Fig. 4b shows that the number of symmetry relations increases monotonically during the CBA iterations, meaning that more and more points in the regular structure are discovered. In Fig. 4c the relations in the set $\mathcal{S}$ are visualized as lines connecting points that are considered n-fold repetitions of each other.

As a conclusion of this section we mention a simple approach to estimate the number of repetitions $\{n_1, \ldots, n_K\}$, and the representative element $\mathcal{E}$ (for space reasons we postpone a more comprehensive discussion). Estimating the representative element $\mathcal{E}$ is essentially equivalent to finding the "starting point" of the regular structure (e.g., the bottom-left corned of the grid in Fig. 4a); in fact, since the size of the representative element $\mathcal{E}$ is known (it is specified by the generators), this point uniquely identifies $\mathcal{E}$. We proceed as follows. For

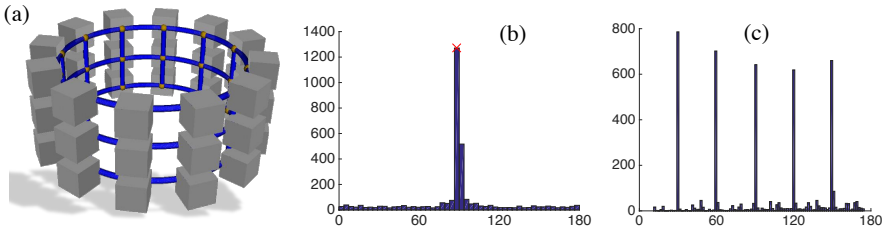

Figure 5: (a) Simulated dataset with 2-parameter regular structure. (b) Histogram of angles in the PTS. (c) Histogram of angles $\{\phi_{ij}\}$ in the rotational transformation space $\mathcal{T}_R^{1d}$.

each generator, we identify the repeated points (these are the extreme points of each segment in Fig. 4c). Then we project these 3D points along the generator direction. Looking at the density of those points along the generator direction, we can estimate an initial and a final point for the regular structure, from which we can also estimate the number of repetitions.

4 Experiments

In this section we complement the results presented so far by reporting all the parameters used in our tests, and we consider a simulated dataset with rotational symmetry.

The Leuven and the Neptune datasets are described in [1, Fig. 1] and [1, Fig. 2]. We perform the initial SfM reconstruction using Bundler [43]; then we extract 3D SIFT keypoints [41] with a search radius of $2.5 \cdot 10^{-2}$. For the multi-hypothesis estimator of Section 3.2, we retain $m = 10$ hypotheses and we set $\varepsilon_\theta = 3$deg. For the ranking technique of Section 3.3, we use $\bar{r}_a = 0.3$ for the Leuven dataset and $\bar{r}_a = 0.2$ for Neptune. The SfM reconstruction is up to scale, hence $\bar{r}_a$ can be dataset-dependent. This radius has to be smaller than the repetition period, to avoid matching a point with itself. For the generator ranking of Section 3.3, we set the threshold on the descriptors' distance to $\bar{d} = 25$deg. The generator estimate produced by Section 3.3 is fairly accurate for the Leuven dataset (compare Fig. 4a with the estimate of [35] in Fig.2c), while for the Neptune dataset it is noticeably incorrect (Fig. 4d). Errors in the generators are more visible for large number of repetitions as inaccuracies are magnified by n for an n-fold repetition. Further results and visualizations are given in [1].

The initial estimate of Section 3.3 is fed to the CBA refinement of Section 3.4. We use the Huber loss as robust cost function in (2), and we set the parameter of the robust kernel to 1.345 as prescribed in [15]. We use the cardinality of the symmetry relations set $\mathcal{S}$ to design a the stopping criterion for CBA. This cardinality quickly increases at the initial iterations, and then tends to stabilize (see Fig. 4b). For both the Leuven and the Neptune dataset, we stop the iterations if the change in the number of relations is less than 5. The refined estimate for the generators is shown in Fig. 1c for Leuven and in Fig. 4e-f for Neptune.

We conclude the experimental section with a dataset that contains a 2-parameter regular structure with a vertical translation symmetry and a rotational symmetry. This is the simulated dataset in Fig. 5a. We set the threshold $\varepsilon_R = 5$deg, to distinguish putatives in the rotational transformation space from the ones containing translations. The peak in Fig. 5b indicates a dominant translation direction (the vertical). Fig. 5c shows the histogram of the angles $\{\phi_{ij}\}$ in the rotational transformation space $\mathcal{T}_R^{1d}$: peaks are repeated with a period of 30deg, capturing the rotational symmetry of the elements in Fig. 5a (more details in [1]).

5 Conclusion

We address the problem of discovering regular structures in SfM and exploiting them to improve 3D reconstruction and understanding. We propose two key contributions: (i) a polar parametrization of the transformation space that exposes dominant symmetry directions and

provides multiple hypotheses for the repetition period; we then leverage appearance information to identify the correct period. (ii) a constrained bundle adjustment formulation that encodes n-fold translational and rotational repetitions; our CBA approach iteratively refines the 3D model and increases the number of detected repeated patterns. We discuss the effectiveness of our approach on real and simulated datasets.

References

[1] Anonymous Authors. Supplemental material: Structural symmetries from motion for scene reconstruction and understanding, BMVC submission: 434.

[2] Sid Ying-Ze Bao and Silvio Savarese. Semantic structure from motion. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[3] A. Berner, M. Bokeloh, M. Wand, A. Schilling, and H.-P Seidel. A graph-based approach to symmetry detection. In *In Proc. of Symp. on Volume and Point-Based Graphics*, 2008.

[4] D. Ceylan, N.J. Mitra, H. Li, T. Weise, and M. Pauly. Factored facade acquisition using symmetric line arrangements. In *Computer Graphics Forum (EUROGRAPHICS)*, volume 31, pages 671–680, 2012.

[5] 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(2), January 2014.

[6] T. Cham and R. Cipolla. A local approach to recovering global skewed symmetry. In *Intl. Conf. on Image Processing A*, volume 1, pages 222–226, 1994.

[7] A. Cohen, C. Zach, S. Sinha, and M. Pollefeys. Discovering and exploiting 3d symmetries in structure from motion. In *CVPR*. Computer Vision and Patter Recognition, June 2012. URL http://research.microsoft.com/apps/pubs/default.aspx?id=163609.

[8] R.W. Conners and C.A. Harlow. Toward a structural textural analyzer based on statistical methods. In *Proc. Intl. Conf. Color in Graphics and Image Processing*, volume 12, 1980.

[9] Sam Friedman and Ioannis Stamos. Real time detection of repeated structures in point clouds of urban scenes. In Michael Goesele, Yasuyuki Matsushita, Ryusuke Sagawa, and Ruigang Yang, editors, *3DIMPVT*, pages 220–227. IEEE, 2011. URL http://dblp.uni-trier.de/db/conf/3dim/3dimpvt2011.html#FriedmanS11.

[10] R. Gal and D. Cohen-Or. Salient geometric features for partial shape matching and similarity. *ACM Trans. on Graphics*, 25(1), 2006.

[11] L. Van Gool, T. Moons, and M. Proesmans. Mirror and point symmetry under perspective skewing. In *CVPR*, 1996.

[12] A.D. Gross and T.E. Boult. Analyzing skewed symmetries. *IJCV*, 13(1), 1994.

[13] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[14] Wei Hong, Allen Y. Yang, Kun Huang, and Yi Ma. On symmetry and multiple-view geometry: Structure, pose, and calibration from a single image. *International Journal of Computer Vision*, 60(3):241–265, 2004. URL http://dblp.uni-trier.de/db/journals/ijcv/ijcv60.html#HongYHM04.

[15] P. Huber. *Robust Statistics*. John Wiley & Sons, New York, NY, 1981.

[16] Y.-D. Jian, D. Balcan, and F. Dellaert. Generalized subgraph preconditioners for large-scale bundle adjustment. In *Intl. Conf. on Computer Vision (ICCV)*, 2011.

[17] N. Jiang, P. Tan, and L.-F. Cheong. Multi-view repetitive structure detection. In *ICCV*, pages 535–542, 2011.

[18] T. Kanade. Recovery of the 3-dimensional shape of an object from a single view. *Artificial Intelligence*, 17, 1981.

[19] T. Kanade and J.R. Kender. Mapping image properties into shape constraints: Skewed symmetry, affine-transformable patterns, and the shape-from-texture paradigm. *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms, M.A. Fischler and O. Firschein, eds.*, 1997.

[20] M. Kazhdan, B. Chazelle, D. Dobkin, T. Funkhouser, and S. Rusinkiewicz. A reflective symmetry descriptor for 3d models. *Algorithmica*, 38, 2003.

[21] S. Lee, R. Collins, and Y. Liu. Rotation symmetry group detection via frequency analysis of Frieze-expansions. In *CVPR*, 2008.

[22] T. Leung and J. Malik. Detecting, localizing and grouping repeated scene elements. In *ECCV*, volume 1, 1996.

[23] H.-C. Lin, L.-L. Wang, and S.-N. Yang. Extracting periodicity of a regular texture based on autocorrelation functions. *Pattern Recognition Letters*, 18, 1997.

[24] Y. Lipman, X. Chen, I. Daubechies, and T. Funkhouser. Symmetry factored embedding and distance. In *SIGGRAPH*, 2010.

[25] Y. Liu, R. Collins, and Y. Tsin. A computational model for periodic pattern perception based on frieze and wallpaper groups. *IEEE Trans. Pattern Anal. Machine Intell.*, 26 (3):354 – 371, March 2004.

[26] Y. Liu, J. Hays, Y-Q. Xu, and H-Y. Shum. Digital papercutting. In *SIGGRAPH*, 2005.

[27] Y. Liu, H. Hel-Or, C.S. Kaplan, and L. Van Gool. Computational symmetry in computer vision and computer graphics. *Foundations and Trends in Computer Graphics and Vision*, 5(1–2), 2010.

[28] G. Loy and J-O. Eklundh. Detecting symmetry and symmetric constellations of features. In *ECCV*, pages 508–521, 2006.

[29] A. Martinet, C. Soler, N. Holzschuch, and F. Sil-Lion. Accurate detection of symmetries in 3d shapes. *ACM Trans. on Graphics*, 25(2), 2006.

[30] M. Mazhdan, T. Funkhouser, and S. Rusinkiewicz. Symmetry descriptors and 3D shape matching. In *In Proc. of Symp. of Geometry Processing*, 2004.

[31] Niloy J. Mitra, Leonidas J. Guibas, and Mark Pauly. Partial and approximate symmetry detection for 3d geometry. In *ACM SIGGRAPH 2006 Papers*, SIGGRAPH '06, pages 560–568, New York, NY, USA, 2006. ACM. ISBN 1-59593-364-6. doi: 10.1145/1179352.1141924. URL http://doi.acm.org/10.1145/1179352.1141924.

[32] N.J. Mitra, M. Pauly, M. Wand, and D. Ceylan. Symmetry in 3D geometry: Extraction and applications. In *EUROGRAPHICS State-of-the-art Report*, 2012.

[33] D. Mukherjee, A. Zisserman, and J. Brady. Shape from symmetry—detecting and exploiting symmetry in affine images. *Phil. Trans. R. Soc. Lond. A*, 351:77–106, 1995.

[34] Pascal Müller, Peter Wonka, Simon Haegler, Andreas Ulmer, and Luc Van Gool. Procedural modeling of buildings. *ACM Trans. Graph.*, 25(3):614–623, July 2006. ISSN 0730-0301. doi: 10.1145/1141911.1141931. URL http://doi.acm.org/10.1145/1141911.1141931.

[35] M. Pauly, N. J. Mitra, J. Wallner, H. Pottmann, and L. Guibas. Discovering structural regularity in 3D geometry. *ACM Transactions on Graphics*, 27(3):#43, 1–11, 2008.

[36] S.C. Pei and L.G. Liou. Automatic symmetry determination and normalization for rotationally symmetric 2D shapes and 3D solid objects. *Pattern Recognition*, 27, 1994.

[37] J. Podolak, A. Golovinskiy, and S. Rusinkiewicz. Symmetry-enhanced remeshing of surfaces. In *In Proc. of Symp. of Geometry Processing*, 2007.

[38] V.S.N. Prasad and L.S. Davis. Detecting rotational symmetries. In *ICCV*, volume 2, pages 954–961, 2005.

[39] D. Raviv, A.M. Bronstein, M.M. Bronstein, and R. Kimmel. Symmetries of non-rigid shapes. In *ICCV*, 2007.

[40] Richard Roberts, Sudipta N. Sinha, Richard Szeliski, and Drew Steedly. Structure from motion for scenes with large duplicate structures. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2011)*, June 2011.

[41] Radu Bogdan Rusu. *Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments*. PhD thesis, Technische Universität München, 2009.

[42] F. Schaffalitzky and A. Zisserman. Geometric grouping of repeated elements within images. *Shape, Contour, and Grouping in Computer Vision, D.A. Forsyth, V. Di Gesu, J.L. Mundy, and R. Cipolla, eds., Springer-Verlag*, 1999.

[43] N. Snavely, S.M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3D. In *SIGGRAPH*, pages 835–846, 2006.

[44] C. Sun and J. Sherrah. 3d symmetry detection using the extended gaussian image. *PAMI*, 19(2), 1997.

[45] C.M. Sun. Fast recovery of rotational symmetry parameters using gradient orientation. *Optical Eng.*, 46(4), 1997.

[46] T. Tuytelaars, A. Turina, and L.V. Gool. Non-combinatorial detection of regular repetitions under perspective skew. *PAMI*, 25(4), 2003.

[47] G. Tzimiropoulos, N. Mitianoudis, and T. Stathaki. A unifying approach to moment-based shape orientation and symmetry classification. *Trans. Img. Proc.*, 18(1), 2009.

[48] Changchang Wu, Jan-Michael Frahm, and Marc Pollefeys. Detecting large repetitive structures with salient boundaries. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, *ECCV (2)*, volume 6312 of *Lecture Notes in Computer Science*, pages 142–155. Springer, 2010. ISBN 978-3-642-15551-2. URL http://dblp.uni-trier.de/db/conf/eccv/eccv2010-2.html#WuFP10.

[49] Fuzhang Wu, Dong-Ming Yan, Weiming Dong, Xiaopeng Zhang, and Peter Wonka. Inverse procedural modeling of facade layouts. *ACM Trans. Graph.*, 33(4):121:1–121:10, July 2014. ISSN 0730-0301. doi: 10.1145/2601097.2601162. URL http://doi.acm.org/10.1145/2601097.2601162.

[50] R.K.K. Yip, W.C.Y. Lam, P.K.S. Tam, and D.N.K. Leung. Hough transform technique for the detection of rotational symmetry. *Pattern Recognition Letters*, 15(9), 1994.

[51] C. Zach, M. Klopschitz, and M. Pollefeys. Disambiguating visual relations using loop constraints. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1426–1433, June 2010. doi: 10.1109/CVPR.2010.5539801.