Probably Approximately Symmetric: Fast Rigid Symmetry Detection with Global Guarantees

Simon Korman*, Roee Litman*, Shai Avidan and Alex Bronstein

School of Electrical Engineering, Tel Aviv University, Israel
* Equal contributors

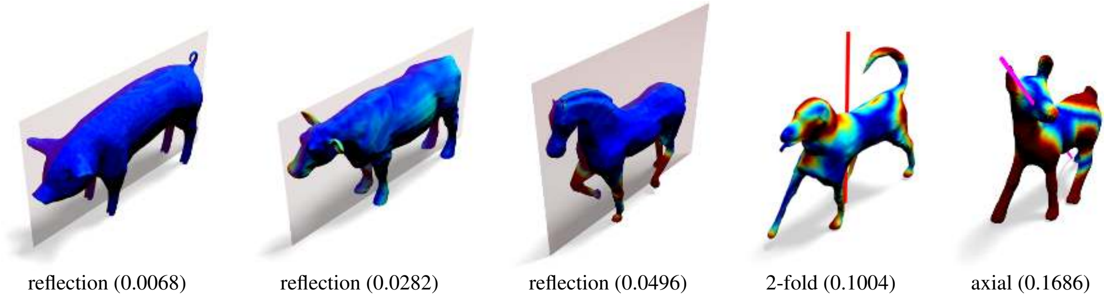


Figure 1: ***Examples from the 'Fourlegs' category of COSEG [WAvK*12], ordered by increasing distortion (symmetry error).*** *The single best detected symmetry is shown for each shape; symmetry type and distortion are reported below each image. The color of each point is the absolute difference between a point and its match in the transformed shape (increasing from blue to red). In the horse, for example, the body is symmetric but the legs are not.*

Abstract

We present a fast algorithm for global rigid symmetry detection with approximation guarantees. The algorithm is guaranteed to find the best approximate symmetry of a given shape, to within a user-specified threshold, with very high probability. Our method uses a carefully designed sampling of the transformation space, where each transformation is efficiently evaluated using a sub-linear algorithm. We prove that the density of the sampling depends on the total variation of the shape, allowing us to derive formal bounds on the algorithm's complexity and approximation quality. We further investigate different volumetric shape representations (in the form of truncated distance transforms), and in such a way control the total variation of the shape and hence the sampling density and the runtime of the algorithm. A comprehensive set of experiments assesses the proposed method, including an evaluation on the eight categories of the COSEG data-set. This is the first large-scale evaluation of any symmetry detection technique that we are aware of.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—

1. Introduction

Symmetry is "a distinction without a difference" in the words of the renowned physicist and Nobel laureate Frank Wilczek. Doubtlessly, symmetry along with the related concepts of self-similarity and invariance is an all-pervasive property of Nature and man-made art. Engineering, architectural, and artistic designs characterized by symmetry usually enjoy structural robustness and efficiency, which also explains why evolution led many biological constructions to assume symmetric properties. Symmetry also plays an important role in our visual perception, in particular that of beauty, and according to modern physical theories, is incorporated deeply into the laws of the universe itself.

Automatic detection of symmetries of a 3D geometric shape has received significant attention in computational geometry, computer graphics, and vision literature. However, despite the steady progress in the field, the task remains computationally challenging. Existing approaches to symmetry detection interpret symmetry as invariance under a certain class of transformations and they can be categorized according to several key features. The taxonomy we present here is by no means complete, and the reader is referred to [MPWC12, HOK10] for a comprehensive survey of symmetry detection in 3D shapes and images.

First and foremost, symmetry is characterized by a group of admissible transformations. While it is customary to tacitly assume the Euclidean group (defining *rigid* symmetries or *congruences* further categorized into reflections or involutions, rotations, and improper rotations or roto-reflections including the former two), more elaborate types of transformations involving uniform scaling (*similarities*), affine and projective transformations, and even *intrinsic* symmetries have been studied in the literature. Our main focus will be restricted to rigid symmetries, though the proposed algorithm and analysis can be extended to practically any group with a finite (and reasonably small) number of parameters, such as the affine group.

Second, symmetries can be classified as *global*, *partial*, and *local*. Global symmetry is defined by a transformation that maps the whole shape onto itself. A shape not possessing a global symmetry can still have partial symmetries in the form of self-similar parts. Local symmetry usually refers to regular spatial arrangements of a structural element into tilings and ornaments. Here, we focus on global symmetries.

Finally, *exact* (perfect) and *approximate* (imperfect) symmetries can be distinguished: the former map the shape exactly to itself, whereas in case of the latter the mapping leads to a distortion smaller than a pre-defined threshold. Depending on the application, exact partial symmetries can also be regarded as approximate global ones. This work focuses on approximate symmetries.

1.1. Prior work

Exact symmetry. Efficient algorithms exist for exact symmetry detection. For example, in the case of a collection of n points in the plane, Atallah [Ata85] describes an algorithm for enumerating all axes of symmetry under reflection of a planar shape. Wolter *et al.* [WWV85] give exact algorithms, based on string matching, for the detection of symmetries of point clouds, polygons, and polyhedra. These algorithms are often impractical due to their sensitivity to noise, because they are restricted to exact symmetries.

Non-parametric symmetry. Several algorithms exist for the detection of non-parametric intrinsic symmetries of deformable shapes (as opposed to the rigid *extrinsic* counterparts). Raviv *et al.* [RBBK10] use a branch-and-bound technique to find global intrinsic symmetries with a prescribed distortion of pairwise geodesic distances. Ovsjanikov *et al.* [OSG08] detects the global *intrinsic* symmetry of shapes using the spectral properties of the Laplace-Beltrami operator. Lipman *et al.* [LCDF10] relieve the assumption of a known transformation group by introducing a symmetry-factored embedding, which enables detecting approximate, as well as partial symmetries of a point cloud, also using spectral methods.

Approximate symmetry. The detection of approximate symmetries has also been addressed in the literature, and can be roughly divided into two approaches:

The first approach defines approximate symmetry by an infimum of a continuous distance function quantifying how similar is a shape to its transformed version. Zabrodsky *et al.* [ZPA95] proposed such a symmetry distance, which has been largely adopted and extended by following works, including our approach. One way to detect an approximate infimum is through an exhaustive evaluation of the transformation space on a grid with a high-enough density. This task can be done naïvely in $\mathcal{O}(n^6)$ for a shape discretized by an $n \times n \times n$ grid, and reaching high accuracy in this approach requires bigger n and an increase in computation times. A more efficient algorithm by Kazhdan *et al.* [KFR04] performs the task in $\mathcal{O}(n^4)$ using an FFT-like approach, but does not provide guarantees on the distance from the optimal possible distortion (See Section 4.4 for a more detailed comparison to our method).

The second approach alleviates the computational complexity by translating the search into a proxy domain, realizing that the set of admissible symmetries is sparse in the transformation space. One of the earliest examples is [SS97], which uses the gaussian image as the proxy domain. Later work by Martinet *et al.* [MSHS06] examines extrema and spherical harmonic coefficients of generalized even moments. Mitra *et al.* [MGP06] cluster Hough-like votes for transformations that align boundaries with similar local shape descriptors. Podolak *et al.* [PSG*06] detect reflection symmetries using a monte-carlo algorithm that selects a pair of surface points and votes for the plane between them. Searching a proxy space provides a set of candidate transformations, that have to be validated directly using some symmetry measure (e.g., [ZPA95]). These candidate transformation are usually further refined using e.g. *Iterative Closest Point* (ICP). Consequently, there is no guarantee on how far is the symmetry measure of the detected symmetries from that of the optimal one.

Our algorithm follows the first approach in that it samples the transformation space, but does so in an efficient manner that guarantees a known approximation error. This enables the use of a branch-and-bound scheme that allows to match the performance of the second approach while maintaining approximation guarantees.

Alignment methods A method related to ours is the surface registration algorithm of [AMCO08] that uses a clever sampling of transformation space. However, it is not easily applicable to the problem of detecting all approximate symmetries of a shape.

Our work follows that of [KRTA13], who proposed a fast method for 2D affine template matching in images with global guarantees. As opposed to their sampling density which depends on a generic image assumption (e.g., image smoothness), ours is determined adaptively according to the specific shape 'complexity'. Additionally, we manipulate the shape representation to control the sampling density, and hence the algorithm's runtime. Finally, while template matching focuses on finding a single best transformation, our goal is to detect all such transformations.

1.2. Contributions

We detect global rigid symmetries in volumetric representations of 3D shapes, and introduce an algorithm that is guaranteed, with high probability, to detect the best symmetry within a given degree of approximation. This is inspired by the classical "probably approximately correct" (PAC) framework [Val84] in learning theory (hence the title of the present paper). To the best of our knowledge, this is the first symmetry detection algorithm coming with such a guarantee. An example output of the algorithm, ranking shapes by their best symmetry, can be seen in Figure 1.

We provide a bound on the required sampling density of transformation space, which is the basis of our algorithm. This bound depends on the desired approximation level as well as, surprisingly, on the 'complexity' of the specific shape, which is manifested through the total variation of its volumetric representation. We further show how to construct shape representations with reduced total variation leading to reduced complexity, and discuss the tradeoff between complexity and sensitivity to noise.

A comprehensive experimental evaluation validates that our approach is capable of detecting approximate symmetries in a large data-set, as well as detecting all symmetries in complicated shapes, all within state-of-the-art execution times.

2. Approximate rigid symmetries

We start by defining our level-set based shape representation and approximate symmetry. We then bound the sample density required to detect approximate symmetries with a user specified precision parameter δ. Generally speaking, although we keep all derivations in the continuous setting, they are straightforwardly amenable to any reasonable discretization of the volume, including hierarchical subdivisions.

2.1. Shape representation

Let $\mathcal{S}$ be a three-dimensional rigid shape with the centroid aligned at the origin. We represent the shape by the $\frac{1}{2}$-sublevel set of a level set function $s : \mathbb{R}^3 \mapsto [0,1]$. The simplest of such representations is the binary indicator function of $\mathcal{S}$ (equalling 1 in the interior); other representations such as truncated distance maps will be discussed in Section 3.1. We will freely interchange between $\mathcal{S}$ and s referring to a shape. We will take the *radius* of the shape to be the smallest scalar r such that the function s is invariant to rotations outside the Euclidean ball $B_r(0)$ of radius r centered at the origin,

$$r = \inf_r \{ s(\mathbf{Rx}) = s(\mathbf{x}) : \mathbf{x} \in B_r(0), \mathbf{R} \in \mathrm{SO}(3) \}. \quad (1)$$

The ball $B_r(0)$ defines the effective support of s, which might be larger than the shape $\mathcal{S}$.

We associate with the shape the *total variation* of s,

$$\mathrm{V}_{\mathcal{S}} = \frac{1}{\mathrm{Vol}\, B_r} \int_{B_r} \|\nabla s(\mathbf{x})\| d\mathrm{vol}(\mathbf{x}), \quad (2)$$

where $d$vol denotes the standard volume element. When s is not differentiable, total variation can be defined using the weak derivative. In particular, for the case of the indicator function $\mathrm{V}_{\mathcal{S}}$ is equal to the ratio between the area of the boundary ∂S and the volume of the bounding ball B_r. Geometrically, total variation can be related to the total curvature of the shape and the amount of "features" it contains. Note that for the case of $\mathrm{O}(3)$, one could have considered derivation and integration only tangent to concentric spheres.

2.2. Rigid symmetries

Let $T \in \mathrm{E}(3)$ be a Euclidean transformation (a combination of translation, rotation, and reflection). The transformed shape $T\mathcal{S}$ will be represented by the indicator function $s(T\mathbf{x})$. T is said to be an *exact global symmetry* of $\mathcal{S}$ if $s(T\mathbf{x}) = s(\mathbf{x})$. The collection of all symmetries of $\mathcal{S}$ forms a group under function composition, which we refer to as the *symmetry group* of $\mathcal{S}$, denoted by $\mathrm{Sym}\,\mathcal{S}$. Each symmetry $T \in \mathrm{Sym}\,\mathcal{S}$ defines a collection of stationary points, $\{\mathbf{x} = T\mathbf{x}\}$, which is known to be either a line or a plane. Such a line or plane is called a *symmetry axis* (or *plane*) of the shape. The set of symmetry axes and planes fully defines the symmetry group of a shape. Since translations have no stationary points, for compactly supported shapes, $\mathrm{Sym}\,\mathcal{S}$ is necessarily a subgroup of the orthogonal group $\mathrm{O}(3)$ containing rotations and reflections around the shape's centroid. For this reason, we will henceforth denote the admissible transformations as 3×3 rotation or reflection matrices, $\mathbf{R}$.

Exact symmetries are a mathematical idealization rarely achieved in practice due to acquisition and representation inaccuracies. In order to account for such imperfections, we define the distortion

$$\begin{aligned} \mathrm{dis}_s \mathbf{R} &= \frac{1}{\mathrm{Vol}\, B_r} \| s - \mathbf{R}^{-1}(s) \|_1 \quad (3) \\ &= \frac{1}{\mathrm{Vol}\, B_r} \int_{B_r} |s(\mathbf{x}) - s(\mathbf{Rx})| d\mathrm{vol}(\mathbf{x}), \end{aligned}$$

where $\mathbf{R}^{-1}(s)$ is a short-hand notation for $(s \circ \mathbf{R})(x) =$

$s(\mathbf{Rx})$. Note that $\mathrm{dis}_s\mathbf{R}$ is bounded to the interval $[0,1]$, and equals zero in the case of perfect symmetry. For solid shapes represented by indicator functions, the distortion can be interpreted as the total amount of mismatched volume and it coincides with the common symmetry measure of [ZPA95]. Note that such an L_1 formulation is more robust to outliers compared to, e.g., the worst-case Hausdorff distance.

We say that $\mathbf{R} \in \mathrm{O}(3)$ defines an ε-symmetry of $\mathcal{S}$ if $\mathrm{dis}_s\mathbf{R} \leq \varepsilon$, and denote by $\mathrm{Sym}_\varepsilon\mathcal{S}$ the collection of all ε-symmetries of $\mathcal{S}$. Note that unlike their exact counterparts, approximate symmetries do not necessarily form a group.

2.3. Sampling of the orthogonal group

In order to practically detect symmetries, one necessarily has to work with a finite sample of the transformation space (i.e. the orthogonal group). The main ingredient of our approach is an upper bound on the sampling density controlled by the maximum allowed distortion of an approximate symmetry.

We begin by defining a metric between any two transformations in the space, which will be used later to define a net of transformations. The metric measures how far apart any point in the ball B_r may be mapped by two different transformations, formally:

$$D(\mathbf{R}_1,\mathbf{R}_2) = \max_{\|\mathbf{x}\|\leq r} \|\mathbf{R}_1\mathbf{x} - \mathbf{R}_2\mathbf{x}\|. \tag{4}$$

Note that this distance does not depend on the shape, but rather only on its support radius r.

A key observation is that the difference in the distortion of two transformations is upper bounded by the product of the shape total-variation $\mathrm{V}_\mathcal{S}$ and the distance D between the transformations. This is formalized in the following proposition, with the accompanying illustration in Figure 2.

Proposition 2.1 $|\mathrm{dis}_s\mathbf{R}_1 - \mathrm{dis}_s\mathbf{R}_2| \leq \mathrm{V}_\mathcal{S} \cdot D(\mathbf{R}_1,\mathbf{R}_2)$ for any $\mathbf{R}_1,\mathbf{R}_2 \in \mathrm{O}(3)$.

Proof First, observe that invoking the triangle inequality and using the group properties,

$$\begin{aligned} |\mathrm{dis}_s\mathbf{R}_1 \;&-\; \mathrm{dis}_s\mathbf{R}_2| \quad \ldots \\ &= \frac{1}{\mathrm{Vol}\,B_r}\cdot \big|\|s-\mathbf{R}_1^{-1}(s)\|_1 - \|s-\mathbf{R}_2^{-1}(s)\|_1\big| \\ &\leq \frac{1}{\mathrm{Vol}\,B_r}\cdot \|\mathbf{R}_1^{-1}(s)-\mathbf{R}_2^{-1}(s)\|_1 \\ &= \frac{1}{\mathrm{Vol}\,B_r}\cdot \|q-\mathbf{R}_1^{-1}(q)\|_1 = \mathrm{dis}_q\mathbf{R} \end{aligned} \tag{5}$$

with $q = \mathbf{R}^{-1}(s)$ and $\mathbf{R} = \mathbf{R}_2\mathbf{R}_1^{-1}$. We can therefore define a new shape $\mathcal{Q} = \mathbf{R}_1\mathcal{S}$ with the corresponding function q, and operate with $\mathrm{dis}_s\mathbf{R}$. Using the group properties, it is also straightforward that $D(\mathbf{R}_1,\mathbf{R}_2) = D(\mathbf{I},\mathbf{R})$, with $\mathbf{I}$ being the identity transformation.

We define the flow $\Phi_\mathbf{R} : (\mathbf{x},t) \to \mathbf{R}^t\mathbf{x}$, $t \in [0,1]$, inducing the orbits $C(\mathbf{x}) = \{\mathbf{R}^t\mathbf{x} : \mathbf{x} \in B_r, t \in [0,1]\}$, whose length is upper-bounded by $D(\mathbf{I},\mathbf{R})$. Using the triangle inequality,

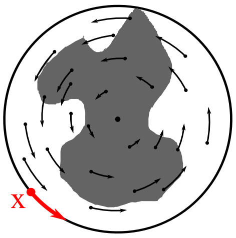


Figure 2: ***A 2D illustration of Proposition 2.1.*** *A planar shape (gray region) represented by a function s undergoes a rotation* $\mathbf{R}$*. The change in value at every point is bounded by the accumulation of* $\|\nabla s\|$ *along the orbit it travels (marked by black arrows). The longest path* $D(\mathbf{I},\mathbf{R})$ *is traveled by the points farthest from the rotation axis, e.g. the one marked in red. Integrating these changes over the entire ball gives* $dis_s\mathbf{R}$*, which is bounded by* $\mathrm{V}_\mathcal{S}\cdot D(\mathbf{I},\mathbf{R})$*.*

$$\begin{aligned} |q(\mathbf{x}) - q(\mathbf{Rx})| \;&=\; |q\circ\Phi_\mathbf{R}(\mathbf{x},0) - q\circ\Phi_\mathbf{R}(\mathbf{x},1)| \\ &\leq\; \int_0^1 \|\nabla(q\circ\Phi_\mathbf{R}(\mathbf{x},t))\|\,\|\dot\Phi_\mathbf{R}(\mathbf{x},t)\|dt \end{aligned} \tag{6}$$

where $\dot\Phi_\mathbf{R}(\mathbf{x},t) = \frac{\partial}{\partial t}\Phi_\mathbf{R}(\mathbf{x},t)$. We can now derive that

$$\begin{aligned} &\mathrm{Vol}\,B_r\cdot\mathrm{dis}_q\mathbf{R} = \int_{B_r} |q(\mathbf{x}) - q(\mathbf{Rx})|d\mathrm{vol}(\mathbf{x}) \\ &\leq \int_{B_r}\int_0^1 \|\nabla(q\circ\Phi_\mathbf{R}(\mathbf{x},t))\|\,\|\dot\Phi_\mathbf{R}(\mathbf{x},t)\|dt\,d\mathrm{vol}(\mathbf{x}) \\ &= \int_0^1\int_{B_r} \|\nabla(q\circ\Phi_\mathbf{R}(\mathbf{x},t))\|\,\|\dot\Phi_\mathbf{R}(\mathbf{x},t)\|d\mathrm{vol}(\Phi_\mathbf{R}(\mathbf{x},t))\,dt \\ &\leq D(\mathbf{I},\mathbf{R})\cdot\int_{B_r} \|\nabla(q\circ\Phi_\mathbf{R}(\mathbf{x},t))\|d\mathrm{vol}(\Phi_\mathbf{R}(\mathbf{x},t)) \\ &= D(\mathbf{I},\mathbf{R})\cdot\int_{B_r}\|\nabla q\|d\mathrm{vol}(\mathbf{x}) = D(\mathbf{I},\mathbf{R})\cdot\mathrm{Vol}\,B_r\mathrm{V}_\mathcal{S}\ , \end{aligned} \tag{7}$$

where the first inequality follows from (6), the following equality follows from the volume preservation under rotations and reflections, and the final equality from the definition of the total variation $\mathrm{V}_\mathcal{S}$.

Finally, the proposition follows by combining inequalities (5) and (7) and using the fact that $D(\mathbf{R}_1,\mathbf{R}_2) = D(\mathbf{I},\mathbf{R})$. □

Proposition 2.1 bounds the change in the distortion under a bounded displacement in the transformation space (the orthogonal group O(3)). We can now turn to defining a finite sampling of O(3). Let us fix a precision parameter $\delta > 0$, set a sampling radius $\rho = \delta/\mathrm{V}_\mathcal{S}$, and construct a discrete set of transformations, $\mathcal{N}_\rho$, forming a ρ-net in O(3) with respect to the distance D, namely that any point in O(3) has a sample in $\mathcal{N}_\rho$ at a distance of at most ρ.

The rationale behind defining such a net of transformations is as follows: Let $\mathbf{R}^* \in \mathrm{O}(3)$ be an ε-symmetry of $\mathcal{S}$. While $\mathbf{R}^*$ will not necessarily be contained in the net $\mathcal{N}_\rho$, there will exist some other $\mathbf{R} \in \mathcal{N}_\rho$ with $\mathrm{dis}_s\mathbf{R} \leq \varepsilon + \delta$. In other words, evaluating the distortion of all the transformations in such a net guarantees the detection of symmetries within a predefined distortion. In the following proposition, we describe an efficient construction of such a net and give a bound on its size.

Proposition 2.2 Let $\mathcal{N}_\rho$ be a ρ-net in $\mathrm{O}(3)$ with respect to the distance D. Then, $|\mathcal{N}_\rho| \le n_\rho = 4\pi\left(\frac{\rho}{r} - \sin\frac{\rho}{r}\right) \sim \mathcal{O}\left(\left(\frac{r}{\rho}\right)^3\right)$.

Proof First, from $D \le r \cdot d$ (d being the standard geodesic distance on $\mathrm{O}(3)$), we conclude that a ρ'-net (for $\rho' = \rho/r$) is a ρ-net in the metric D. We can therefore use the more convenient d, proceeding with the standard packing number argument: The total volume of $\mathrm{O}(3)$ is given by twice the area of the three-dimensional hypersphere, $2\mathrm{Vol}\,\mathbb{S}^3$. Since $\mathcal{N}_\rho$ is ρ'-separated, the balls $B_{\rho'/2}$ in $\mathbb{S}^3$ form a disjoint collection, whose volume is smaller than the total volume of $\mathrm{O}(3)$. The bound is obtained by demanding $n_\rho \mathrm{Vol}\, B_{\rho'/2} = 2\mathrm{Vol}\,\mathbb{S}^3$. Substituting closed form expressions for the volumes on the sphere yields $n_\rho \pi(\rho' - \sin\rho') = 4\pi^2$, from where n_ρ is obtained. Finally, using the Taylor expansion $\sin\rho' = \rho' - \frac{\rho'^3}{6} + \mathcal{O}(\rho'^5)$ yields $\rho' - \sin\rho' \sim \mathcal{O}(\rho'^3)$. □

Our sampling of the orthogonal group is summarized in the following corollary, which is a direct consequence of the combination of Propositions 2.1 and 2.2, with the choice of $\rho = \delta/\mathrm{V}_{\mathcal{S}}$.

Corollary 2.3 For a given precision parameter $\delta > 0$, there exists a sampling of the orthogonal group of size $\mathcal{O}((\frac{r \cdot \mathrm{V}_{\mathcal{S}}}{\delta})^3)$, such that for any given transformation T in $\mathrm{O}(3)$, the sample contains a transformation, whose distortion is bounded away by δ from the distortion of T.

Such a sampling of $\mathrm{O}(3)$ can be achieved, using e.g. farther-point sampling [ELPZ97] or fixed-step strategies. The choice of $\rho = \delta/\mathrm{V}_{\mathcal{S}}$ translates into an angular density of about $\frac{2\pi\cdot\delta}{r\cdot\mathrm{V}_{\mathcal{S}}}$ in the standard axis-angle parametrization.

2.4. Fast evaluation of the distortion

A naïve symmetry detection algorithm consists of testing whether $\mathrm{dis}_s\mathbf{R} \le \delta$ for each $\mathbf{R}$ in the net $\mathcal{N}_\rho$. However, such a test requires the computation of the integral (3) for each sample, which results in a non-trivial complexity. To alleviate this burden, we use a faster randomized sub-linear sampling procedure, which gives approximately the same result with overwhelmingly high probability.

Let $\mathbf{x}_1, \cdots, \mathbf{x}_m$ be points randomly drawn from the uniform distribution on B_r. We define the approximate distortion as

$$\widetilde{\mathrm{dis}}_s\mathbf{R} = \frac{1}{m}\sum_{i=1}^{m} |s(\mathbf{x}_i) - s(\mathbf{R}\mathbf{x}_i)|, \tag{8}$$

where each of the summands is bounded on $[0,1]$. Since $\mathbb{E}\{\widetilde{\mathrm{dis}}_s\mathbf{R}\} = \mathrm{dis}_s\mathbf{R}$, we can use the Chernoff-Hoeffding inequality to bound the probability $P(|\widetilde{\mathrm{dis}}_s\mathbf{R} - \mathrm{dis}_s\mathbf{R}| > \varepsilon)$, leading to the following

```
input  : Shape $\mathcal{S}$ represented by $s$; precision parameter
         $\delta > 0$; error probability $p$
output : Approximate symmetry $\mathbf{R} \in \mathrm{O}(3)$; approximate
         distortion $d$
Construct a $\rho = \frac{\delta}{2\mathrm{V}_{\mathcal{S}}}$-net $\mathcal{N}_\rho$ on $\mathrm{O}(3)$
foreach $\mathbf{R} \in \mathcal{N}_\rho$ do
    Sample $m_{\delta/2} = \frac{2}{\delta^2}\log\frac{2}{p}$ random points from $B_r$
    Compute $\widetilde{\mathrm{dis}}_s\mathbf{R} = \frac{1}{m}\sum_{i=1}^{m} |s(\mathbf{x}_i) - s(\mathbf{R}\mathbf{x}_i)|$
end
Return $\mathbf{R}$ with the minimal $d = \widetilde{\mathrm{dis}}_s\mathbf{R}$
```

Algorithm 1: Best approximate symmetry detection.

Proposition 2.4 For $m_\varepsilon = \frac{1}{2\varepsilon^2}\log\frac{2}{p} = \mathcal{O}(\varepsilon^{-2}\log\frac{1}{p})$, $|\mathrm{dis}_s\mathbf{R} - \widetilde{\mathrm{dis}}_s\mathbf{R}| \le \varepsilon$ with probability higher than $1-p$.

3. Symmetry detection algorithm

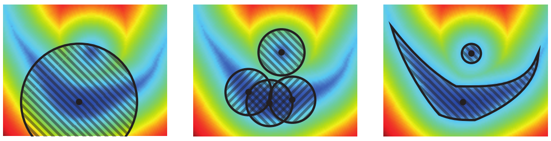
Putting the pieces together, we summarize in Algorithm 1 the proposed method for detecting the best approximate symmetry. Combining the previous results, we state the following

Theorem 3.1 The runtime complexity of Algorithm 1 is $\mathcal{O}((r \cdot \mathrm{V}_{\mathcal{S}})^3\delta^{-5}\log\frac{1}{p})$ and with probability $1-p$, it holds that:

1. if $d \le 0.5 \cdot \delta$, then $\mathbf{R}$ is a δ-symmetry of $\mathcal{S}$
2. if $d > 1.5 \cdot \delta$, then $\mathcal{S}$ has no δ-symmetries.

Observe that unless some elements of $\mathcal{N}_\rho$ are removed, the second condition will never happen, as the algorithm will return a symmetry δ-close to the identity transformation.

Algorithm 1 detects a single approximate symmetry of $\mathcal{S}$. In order to detect the entire $\mathrm{Sym}_\delta\mathcal{S}$, we run the algorithm sequentially, each time removing a neighborhood of the detected transformation $\mathbf{R}$ from $\mathcal{N}_\rho$. The neighborhood can be naturally defined as the δ-component of $\mathbf{R}$, computed by ap-



too large radius too small radius flood-fill

Figure 3: ***Illustration of minima neighbourhood removal.*** *A 2D function, visualized using a heat-map. In our implementation, we use a fixed-sized removal radius. Using a too large radius may remove other minima, while a too small one leaves areas that might be detected in the next iteration. These phenomena may be avoided by applying a flood-fill procedure.*

```
input  : Shape $\mathcal{S}$ represented by $s$; precision parameter
         $\delta > 0$; error probability $p$
output: Collection of $\delta$-symmetries $S = \mathrm{Sym}_{\delta}\mathcal{S}$
Initialize $S = \emptyset$
Construct a $\rho = \frac{\delta}{2\mathrm{V}_{\mathcal{S}}}$-net $\mathcal{N}_\rho$ on $\mathrm{O}(3)$
while not all symmetries have been detected do
    Run Algorithm 1 on $\mathcal{N}_\rho$ to detect $\mathbf{R}$ with
    approximate distortion $d$
    if $d > 1.5 \cdot \delta$ then stop;
    if $\mathbf{R}$ is a rotation (and not a reflection) then
        Let $\mathcal{X}$ be the set of $n$-fold symmetries along its
        axis, with distortion $\leq \delta$    $(n = 2, \cdots, N)$
    else
        Let $\mathcal{X} = \{\mathbf{R}\}$
    end
    Add $\mathcal{X}$ to $S$ and remove from $\mathcal{N}_\rho$ a fixed-sized
    neighborhood of each symmetry in $\mathcal{X}$
end
Return all detected transformations $\mathbf{R}$
```

Algorithm 2: Detection of all approximate symmetries.

plying a flood-fill procedure to $\mathcal{N}_\rho$. Alternatively, the neighborhood can be defined as a ball of a fixed radius with respect to the standard geodesic distance on $\mathrm{O}(3)$. The latter approach was adopted in our experiments due to its simplicity, despite the problems that may arise when using a too small or too big radius (see Figure 3 for an illustration).

Algorithm 2 summarizes the described procedure for the detection of all approximate symmetries. When a rotation symmetry is detected, we further investigate its axis to find its n-fold symmetries (up to some integer N). We report on an n-fold rotation if the distortion of all its n members is below δ. A continuous (axial) rotational symmetry is reported when the distortions of all members of all n-fold rotations, $n = 2, \ldots, N$, are below δ.

3.1. Manipulating the shape complexity

For a fixed precision δ, the complexity of our symmetry detection algorithm is governed by the term $(r \cdot \mathrm{V}_{\mathcal{S}})^3$, which is the cube of the shape *complexity factor* $\mathrm{C} = r \cdot \mathrm{V}_{\mathcal{S}}$, a unitless quantity that resembles the isoperimetric quotient and describes the geometric complexity of the function s representing the shape.

Through the total variation of s, C depends on the function representing $\mathcal{S}$ and not directly on $\mathcal{S}$ itself. This leads to the important issue of designing representation functions for shapes that minimize the computation complexity.

To this end, we suggest controlling the shape complexity using *truncated signed distance function* (TSDF) representations. The advantages of doing so are two-fold: First, the TSDFs produce smoother shape representations, which lead to faster running time. Additionally, the resulting representation values have lower variance (as a result of increased smoothness), allowing use of tighter bounds than the one stated in Proposition 2.4, which only assumes that the summands are bounded but does not consider their variance.

Nevertheless, these advantages come at the cost of the enhancement of thin structures (possibly amplifying the effect of shot-noise) as well as the possible loss of discriminativity, due to smoothing of fine details. The effects (noise amplification and loss of discriminativity) are studied below. Also, as part of our large-scale experiment in Section 4.3, we analyze the influence of these choices on discriminativity.

For a given (truncation) constant K, we define the Euclidean TSDF by

$$d_K(\mathbf{x}, \partial\mathcal{S}) \;=\; \min\{K, \max\{-K, d(\mathbf{x}, \partial\mathcal{S})\}\} \qquad (9)$$

where $d(\mathbf{x}, \partial\mathcal{S})$ is the signed distance map from the boundary $\partial\mathcal{S}$ of the shape S. Since $d(\mathbf{x}, \partial\mathcal{S})$ satisfies the eikonal equation $\|\nabla d(\mathbf{x}, \partial\mathcal{S})\| = 1$ almost everywhere, the total variation of $d_K(\mathbf{x}, \partial\mathcal{S})$ is bounded by

$$\mathrm{V}_{\mathcal{S}}(d_K) \;\leq\; \frac{c \cdot K \cdot \mathrm{Area}\,\partial\mathcal{S}}{\mathrm{Vol}\,B_{r+K}}, \qquad (10)$$

where c is a constant.

We construct a family of functions

$$s_K(\mathbf{x}) \;=\; \frac{1}{2K} d_K(\mathbf{x}, \partial\mathcal{S}) + \frac{1}{2} \qquad (11)$$

with the image in $[0,1]$, whose $\frac{1}{2}$-sub-level set is $\mathcal{S}$. For $K = 0$, s_0 is simply the binary indicator function. Denoting $\mathrm{V}_{\mathcal{S}K} = \mathrm{V}_{\mathcal{S}}(s_K)$ and $\mathrm{C}_K = \mathrm{C}(s_K)$, we observe that

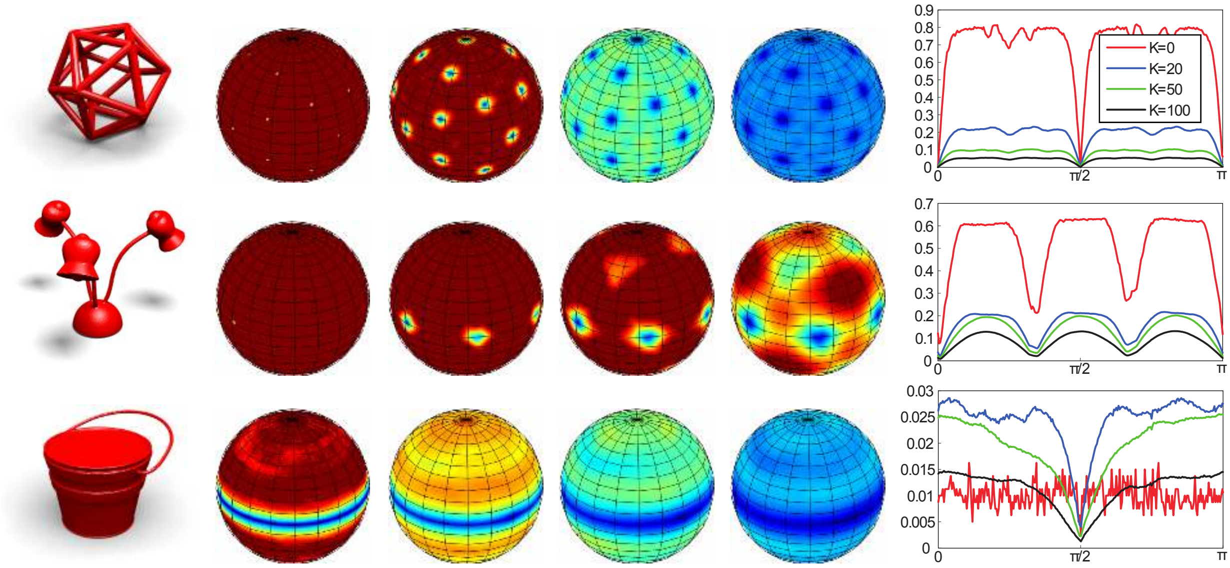
$$\begin{aligned} \mathrm{C}_K \;&\leq\; (r+K) \cdot \mathrm{V}_{\mathcal{S}K} \leq (r+K) \frac{\mathrm{V}_{\mathcal{S}0} \cdot \mathrm{Vol}\,B_r}{\mathrm{Vol}\,B_{r+K}} \\ &=\; \mathrm{C}_0 \left(\frac{r}{r+K}\right)^2, \end{aligned} \qquad (12)$$

which for $K \gg r$ becomes $\mathrm{C}_K \sim \mathcal{O}(K^{-2})$. Therefore, from the point of view of the complexity factor alone, it is advantageous to increase K without limits.

However, a large K has a negative impact on the noise resilience of the symmetry detection algorithm. To visualize this, assume that the shape is almost perfectly symmetric, such that under a transformation $\mathbf{R} \in \mathrm{O}(3)$, s_0 and $s_0\mathbf{R}$ match except for on a small ball B_ε resulting from noise. Therefore, using s_0, $\mathbf{R}$ has the distortion of $\delta_0 = \mathrm{dis}_s \mathbf{R} = \mathrm{Vol}\,B_\varepsilon / \mathrm{Vol}\,B_r$. Increasing K yields

$$\delta_K \;=\; \frac{\mathrm{Vol}\,B_{\varepsilon+K}}{\mathrm{Vol}\,B_{r+K}} = \left(\frac{\varepsilon+K}{r+K}\right)^3, \qquad (13)$$

which for $K \gg r$ becomes $\delta_K \sim 1$, amplifying the noise to unreasonable proportions.



(a) shape (b) K=0 (binary) (c) K=20% (d) K=50% (e) K=100% (f) 1D profile along equator

Figure 4: ***Distortion maps for different truncation levels.*** *We consider only reflection symmetries in this illustration. Each row shows one example in the following format: (a) the shape (b)-(e): distortion levels of the function s_K for different truncation values K. Color-coding ranges from 0 (blue) to a clipped value of 0.2 (red) at each location and represents the respective distortion of the planar reflection symmetry, whose normal passes through the point. (f) a section of the distortion along the equator of the sphere. See text for interpretation.*

The same effect also decreased the sensitivity to fine features. Suppose two approximate symmetries R_1 and R_2 maintain the shape invariant except for on small balls B_ε and $B_{2\varepsilon}$, respectively. The corresponding distortions when using s_0 are

$$\delta^1 = \frac{\mathrm{Vol}\, B_\varepsilon}{\mathrm{Vol}\, B_r} = \left(\frac{\varepsilon}{r}\right)^3 , \quad \delta^2 = \frac{\mathrm{Vol}\, B_{2\varepsilon}}{\mathrm{Vol}\, B_r} = \left(\frac{2\varepsilon}{r}\right)^3 = 8\delta^1 .$$

That is, R_2 is an order of magnitude "worse" than R_1, which is typically an easily detectable situation.

When increasing K ,the discriminativity, viewed as the ratio

$$\frac{\delta^2_K}{\delta^1_K} = \left(\frac{2\varepsilon+K}{r+K}\right)^3 \Big/ \left(\frac{\varepsilon+K}{r+K}\right)^3 = \left(\frac{2\varepsilon+K}{\varepsilon+K}\right)^3 \qquad (14)$$

approaches 1 for $K >> \varepsilon$, meaning that features of size K are smoothed out.

4. Experiments and applications

The code used to generate the reported results can be downloaded from the project webpage [KLAB].

Data-sets The main data-set we work with is the COSEG data-set [WAvK*12], which includes (among other) 190 shapes belonging to eight categories, that were originally purported for the evaluation of segmentation algorithms. While the shapes were created using CAD tools, they are not completely synthetic in their nature. Specifically, while most of the shapes have at least one kind of symmetry, in the vast majority of the cases the symmetry is far from being perfect, which makes its detection challenging. We first rasterize a randomly rotated version of each shape into a cartesian voxelized volume, where the maximal dimension is taken to be 160 voxels. Since all COSEG shapes are vertically aligned, a random rotation disables the advantage of any specific sampling location. Then, we center the shape around its centroid and measure its support. Finally, we pad and crop the volume to a cube, with the side length twice the shape support radius r. This guarantees that the shape remains within the volume under arbitrary rotations and reflections. The final volume dimensions are around 200^3.

We also created a small data-set of shapes that have complex symmetry groups. These include the icosahedron and the dodecahedron (See Figure 10 for an illustration). Finding *all* the symmetries of such shapes is computationally challenging. The third type of data we use is a volumetric scalar MRI image taken from [CKK*97].

Symmetries and their notations We seek to find approximate symmetries of different kinds. The first kind are *planar reflections*, around a generally oriented plane which passes through the centroid. We denote such a symmetry by **REFL** and visualize it as a transparent plane. The second kind are t-fold rotations around an axis that passes through the centroid (where we search for t between 2 and 20). Such a symmetry (or a set of symmetries) includes rotation symmetries of the

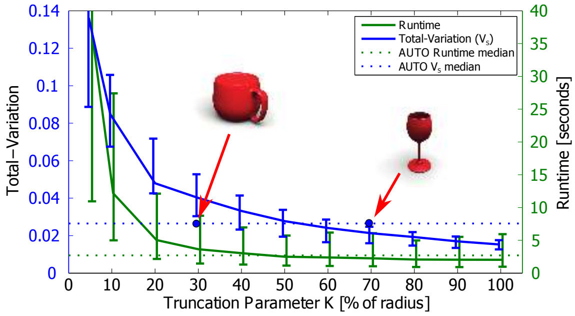


Figure 5: ***The empirical dependence of $\mathrm{V}_{\mathcal{S}}$ and run time on K.*** *Large K leads to lower total-variation $\mathrm{V}_{\mathcal{S}}$. Plotted are the median of the total-variation (and of the runtime in seconds), over the entire COSEG data-set. Error bars are 10^{th} and 90^{th} percentiles. See Section 4.1 for details.*

set of angles $\{2\pi i/t\}$ for $i = 1,...,t-1$. We denote such a symmetry by **t-fold-ROT** and visualize its rotation axis in red. The third kind, axial-symmetries, are fully-continuous rotation symmetries around some axis. We denote these by **CONT** and visualize them using a magenta colored axis.

Algorithm settings and implementation details We represent a shape s_K by applying a TSDF, where the truncation parameter K is chosen adaptively such that the total variation $\mathrm{V}_{\mathcal{S}}$ of the shape is approximately $3/r$, as detailed in Section 4.2. When running the main algorithm (Algorithm 2), we aim for high precision, which translates into invoking the (single symmetry detection) Algorithm 1 with low values of the precision parameter δ. For efficiency, we run Algorithm 1 in a branch-and-bound manner, which begins with an initial (coarse) net defined by $\delta = 0.25$ and iteratively increases resolution only in "promising" regions of the transformation space, finally reaching the desired resolution. Note that this can be done as in [KRTA13], based on our net construction, while keeping the theoretical guarantees. In addition, after the detection of each symmetry, we carve out all transformations whose symmetry axis is less than 10° from that of the detected axis and then repeat the search for additional symmetries. All experiments were run on a 2.70GHz machine, with 8GB RAM. Our timings throughout (excluding Table 1 and 2) do not include a 0.8 seconds pre-processing time for the TSDF computation, per shape.

4.1. The influence of truncation

Figure 4 shows intermediate results of running our algorithm when a shape is represented with the binary indicator function, s_0, as well as with s_K with various truncation values K, on a variety of shapes. Each row depicts a different shape, followed by color-maps of the different distortion levels on a hemisphere (The hemisphere has the same orientation of the shape). For simplicity we focus only on planar reflective symmetries. The color-coding at each location on the hemisphere represents the distortion of the planar reflection symmetry, whose normal passes through the point. Each column of (b)-(e) shows such a sphere, for K taken to be 0%, 20%, 50%, and 100% of the shape radius. The last column (f) shows a 1D profile of the map around the equator of the sphere. As can be seen, when $K = 0$ (the binary indicator case) the distortion is relatively volatile. Increasing K decreases the total variation (and shape complexity factor, as in Eq. (12)) and therefore makes the distortion map much smoother (see Proposition 2.1). As a result, the sampling rate required by the algorithm can be decreased. Notice that in the case of the lamp (second row) the distortion of the binary indicator function (i.e. $K = 0$) is high even around approximate symmetries, because the shape is not perfectly symmetric. An increase in K is necessary is such cases (see discussion in Section 4.3). Finally, the binary indicator function is not affected much by the handle of the bucket (last row) because this is a very thin structure. The detection of the exact symmetry would require extremely fine sampling, while increasing K increases the sensitivity to fine features.

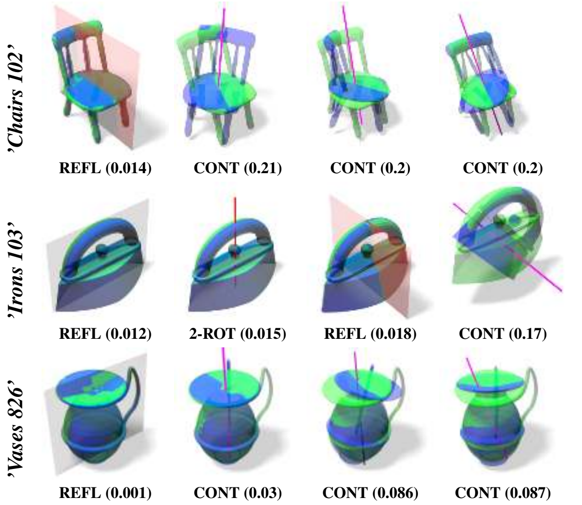


Figure 6: ***Four best symmetries of COSEG shapes.*** *Three typical examples from the data set with different numbers of symmetries of different quality. The original shape is shown in green and the shape mapped to by the detected transformation is shown in blue. The chair shape has one clear reflection; three other transformations have higher distortion (reported in parentheses). The iron has three good symmetries (two reflections and a 2-fold rotation), while the vase has one nearly perfect reflection symmetry, and another rotation symmetry that is inexact because of the handle.*

4.2. Automatic selection of the truncation parameter K

It is desirable to make the runtime of Algorithm 1 depend only on the precision parameter δ, and not on the properties of the shape itself. Recall that the number of samples (net size) depends on the total variation $\mathrm{V}_{\mathcal{S}}$, which in turn depends on the shape representation. Therefore, we would like

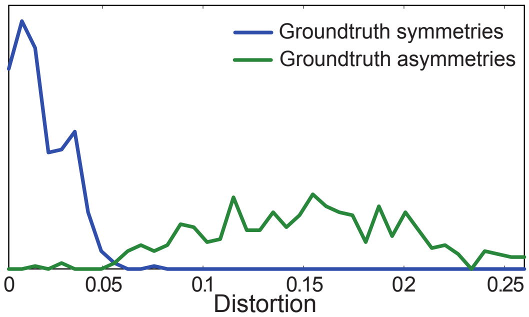


Figure 7: ***Distortion distribution of manually labeled results.*** *We manually labeled as "approximate symmetry" or "non-symmetry" the best four detections of the 190 COSEG shapes, and plotted the distortion distributions over these two sample sets. It turns out that these sets can be separated nicely by thresholding the distortion. This empirically established threshold is used as the stopping criterion in subsequent experiments.*

to automatically choose a value of K for each shape, that will produce $V_{\mathcal{S}}$ proportional to $1/r$. This will make the shape complexity factor C relatively constant (since $C = r \cdot V_{\mathcal{S}}$) and therefore ensures constant runtime for a wide range of values of δ. To this end, we measured the empirical dependence of the total variation on the truncation level K, over the entire COSEG data-set. Figure 5 shows the median, 10th and 90th percentiles of $V_{\mathcal{S}}$ (in blue) and runtimes (in green, for a single detection per shape). Observe that increasing K reduces both runtime and the total variation of the shape, in accordance with the bound stated in Section 3.1.

The automatic choice of K for an unseen shape, is determined by initially calculating its $V_{\mathcal{S}}$ for an arbitrary value of K and then by improving the choice using a binary search, according to the above empirical distribution. We empirically chose the goal of $C = 3$, which implies $V_{\mathcal{S}} = 3/r$. Figure 5 also shows two specific shapes. For the shape '*goblets 32*' (right) whose shape complexity C is large, we use a rather high truncation of $K = 0.7r$ in order to reduce its $V_{\mathcal{S}}$ drastically to the order of $3/r$. On the other hand, the shape '*vases 801*' (left), which is more solid and has a lower shape complexity requires a much lower truncation of $K = 0.3r$. The described automatic calculation of K, which we term 'AUTO', was used throughout the following experiments.

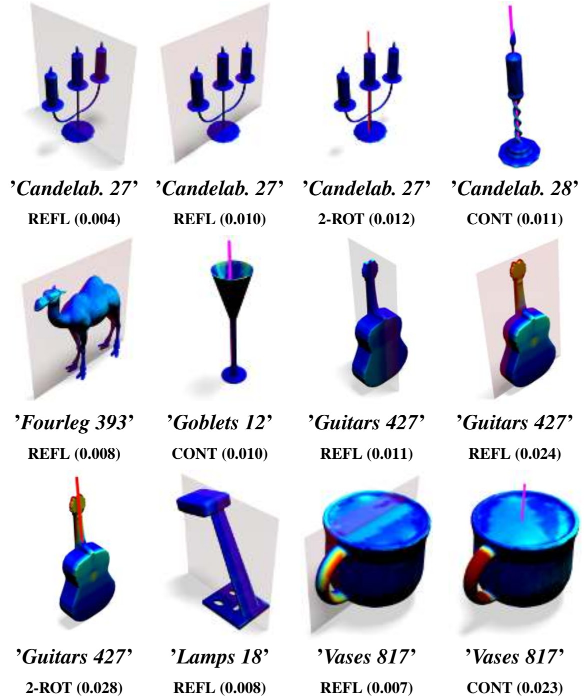
We empirically evaluated the influence of the truncation level K on the discriminativity of the detection algorithm, as part of the large-scale experiment in the following section.

4.3. Large-scale evaluation

After establishing the accuracy and runtime complexity of our algorithm, we tested it on the complete COSEG dataset. To the best of our knowledge, we are the first to report symmetry detection results on a data-set of this scale. At the first stage, we ran the algorithm to detect the four best symmetries per shape giving a total of $760 = 190 \cdot 4$ symmetries. We did so (even though each shape might have less or more than 4 approximate symmetries) in order to investigate the option of automatically detecting when the returned symmetry is indeed an approximate symmetry. See Figure 6 for the four first detected symmetries (along with their distortion levels) for three such example shapes. As an example, the 'Chairs 102' shape has only one approximate symmetry, a planar reflection, which we detect first and which has low distortion of 0.014. The three remaining detected symmetries, which can not be considered approximate symmetries, have distortions levels of over 0.2.

We manually labeled all the 'potential' symmetries returned by our algorithm as either "approximate symmetry" or "non-symmetry" and plotted a histogram of the respective distortions. As can be seen in Figure 7, our distortion measure is fairly invariant to the shape and a global threshold of $dis = 0.05$ can be used to determine if the detected symmetry is indeed a meaningful symmetry. We then ran the algorithm again, with the stopping criterion defined by this threshold. The algorithm found a total of 463 symmetries in the 190 shapes. Please refer to the supplementary material (availiable also on [KLAB]) for the complete set of detections. In Figure 8, we show the approximate symmetries that were detected for eight representative shapes. Notice, that while some of the shapes have almost perfect symmetries (camel, lamp, candelabras), some others have (possibly, in addition to a perfect symmetry) some symmetries that are only approximate (e.g. cup and guitar). In the latter cases, our visualization shows how well each part of the shape undergoes the symmetry (see cup handle and guitar neck).

Discriminativity analysis The use of TSDF representations significantly improves the runtime complexity of the algorithm, as was shown in Figure 5. We now turn to test the qualitative consequences of manipulating the truncation parameter K (as well as of the automatic choice of K). While the large-scale experiment was done using the automatic selection of K (described in Section 4.2), here we experiment also using a fixed set of truncation levels K. The results are summarized in Figure 9, where we count for each representation the number of detected approximate-symmetries as well as the number of false detections. As expected, the increase in K comes at a certain loss of discriminativity, as the ratio between true and false detections slightly deteriorates. The increase in the number of detected approximate symmetries, up to a certain level of K, is due to the rejection of approximate symmetries by the less smooth representations (i.e. lower K's). Most noticeably, our automatic selection mode outperforms any fixed selection of K in terms of the number of true detections and the ratio of false detections as well as in terms of runtime (see Figure 5).



’Candelab. 27’ **REFL (0.004)**
’Candelab. 27’ **REFL (0.010)**
’Candelab. 27’ **2-ROT (0.012)**
’Candelab. 28’ **CONT (0.011)**
’Fourleg 393’ **REFL (0.008)**
’Goblets 12’ **CONT (0.010)**
’Guitars 427’ **REFL (0.011)**
’Guitars 427’ **REFL (0.024)**
’Guitars 427’ **2-ROT (0.028)**
’Lamps 18’ **REFL (0.008)**
’Vases 817’ **REFL (0.007)**
’Vases 817’ **CONT (0.023)**

Figure 8: ***Detected approximate symmetries on representative COSEG shapes.*** *Several examples - 8 shapes and the 12 symmetries we detected for them, using the threshold from Figure 7 as the largest admissible distortion. See the supplementary material for all 463 symmetries detected on the entire dataset.*

4.4. Comparison with Kazhdan *et al.* [KFR04]

As mentioned in Section 1.1, the work of Kazhdan *et al* [KFR04] bares some resemblance to the present work, mainly because it also directly evaluates many transformations using the measure of [ZPA95] and therefore some of our bounds may apply to it. In spite of the similarities, there are some major differences which make the comparison difficult. The methods mainly differ in the choice of the set of transformations, as well as in the way by which they are evaluated.

The FFT-like approach in [KFR04] requires a regular $n \times n$ grid sampling of the latitude-longitude space of transformations, where n is preferably a complete power of 2. For this setting, the complexity of [KFR04] is $O(n^4)$, which is dominated by an auto-correlation computation of order $O(n^4)$, followed by a computation of order $O(kn^2)$ for detecting k-fold symmetries for each k. In comparison, our method's complexity, $O(n^3 \cdot \varepsilon^{-2} \log \frac{1}{p})$, scales more gracefully with n.

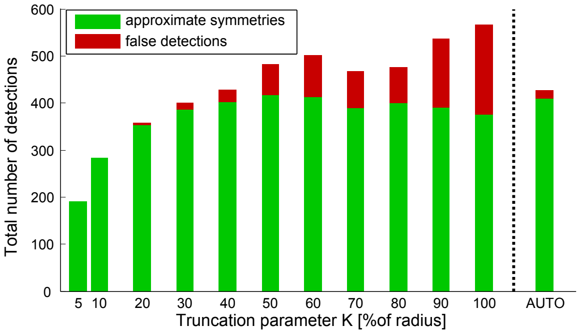


Figure 9: ***The empirical discriminativity vs. K.*** *Low values of K lead to rejection of some approximate symmetries, while large ones reduce the discriminative power of our method. The automatic selection of K leads to a favorable tradeoff. See Section 4.3 for details.*

We first evaluated both methods on the task of detecting the best symmetry in $O(3)$ (including reflections and rotations of up to 8 folds) for the shapes in the COSEG data-set and compared several grid-size configurations of [KFR04] against our method. The publicly available implementation of [KFR04] does not allow access to the $O(n^4)$ auto-correlation result and requires to recalculate it for each k-fold detection and therefore, for fairness of comparison, we report the average time over all k-fold computations for [KFR04]. The results summarized in Table 1 show that our method reaches lower distortion values even when compared to a grid of 256^2, which takes much longer to evaluate. Note that all distortion results are calculated on the original shape representation, following [ZPA95].

algorithm	grid size	number of trans.	distortion [ZPA95]	runtime [sec]
Kazhdan *et al.* [KFR04]	32^2	1,024	0.160	**0.17**
	64^2	4,096	0.087	0.42
	128^2	16,384	0.056	3.42
	256^2	65,536	0.044	30.56
Proposed method	–	106,054	**0.040**	2.62

Table 1: ***Best symmetry detection.*** *The table summarizes performance of two algorithms for best symmetry detection on the COSEG data-set. Presented are median values for: (i) number of evaluated transformations (ii) symmetry measure [ZPA95] and (iii) run-time.*

The sets of transformations used by both methods are quite similar when limited to reflection symmetries, therefore we were able to perform a more delicate comparison in this setting. We made several modifications to our method so it would be directly comparable to [KFR04]. First, we set our net sizes to be equivalent to the grid sizes in the code of [KFR04], rather than following Proposition 2.2 by using V_S and δ. For the same reason, we disabled the branch-and-bound procedure, which allows reaching fine resolutions even with an initial coarse net. These modifications have a negative impact on our algorithm. Note however that this comparison does not show the full effectiveness of [KFR04], as the $O(n^4)$ auto-correlation preprocess (which is mandatory) is used only for the limited case of reflections.

We ran our algorithm in two configurations: one with the original binary shape ($K=0$), and another with automatic selection of K (AUTO), as can be seen in Table 2. Note that our net covers the transformation space uniformly, while the sampling used in [KFR04] (a product of equidistant samples in the azimuth-elevation space) is denser around the 'poles' and less around the 'equator'. This explains the slightly better distortions we obtain, for each fixed grid size.

algorithm	grid size	number of trans.	distortion [ZPA95]	runtime [sec]
Kazhdan *et al.* [KFR04]	32^2	$1,024$	0.162	**0.19**
	64^2	$4,096$	0.085	0.32
	128^2	$16,384$	0.057	2.64
	256^2	$65,536$	0.044	27.65
Proposed method ($K=0$)	$\sim 32^2$	$1,031$	**0.076**	**0.19**
	$\sim 64^2$	$4,094$	0.055	**0.22**
	$\sim 128^2$	$16,370$	0.045	**0.29**
	$\sim 256^2$	$65,301$	0.044	**0.50**
Proposed method (AUTO K)	$\sim 32^2$	$1,031$	**0.076**	0.96*
	$\sim 64^2$	$4,094$	**0.052**	1.02*
	$\sim 128^2$	$16,370$	**0.041**	1.12*
	$\sim 256^2$	$65,301$	**0.035**	1.40*

Table 2: ***Best reflection symmetry detection.*** *The table summarizes performance of three algorithms for best reflection symmetry detection on the COSEG data-set. See Table 1 and text for more details. (*) Runtimes for 'AUTO K' include a pre-processing of* $\sim$ 0.8 *seconds for the TSDF calculation.*

A benefit of using the TSDF representation is evident when comparing the distortions achieved in the $K=0$ and the AUTO runs. As stated in Section 3.1, applying the TSDF lowers the variance of the shape representation, allowing use of tighter bounds than the one mentioned in proposition 2.4 (which only assumes the individual summands are bounded and does not take advantage of their variance). As a result, better estimation of the distortion is achieved.

4.5. Complex symmetry groups

We tested our algorithm on shapes with known complex symmetries. Cumulative runtimes for finding the entire set of symmetries are reported in Figure 10. In both cases (Dodecahedron and Icosahedron), the algorithm correctly detected exactly all 46 symmetries. The average symmetry detection time of the algorithm *decreases* as the number of symmetries of a shape increases because after each symmetry is detected we carve out its neighborhood.

Martinet *et al.* [MSHS06] evaluate their algorithm on the icosahedron as well. The runtime we report here (45 seconds) is more than an order of magnitude smaller compared to the 50 minutes for detecting all symmetries of the icosahedron, reported in [MSHS06], when treating the icosahedron as a single complex shape. The authors of [MSHS06] also report a runtime of 1 minute and 57 seconds when applying their 'constructive' method, which assumes that the icosahedron is given as a segmented set of 30 tiles.

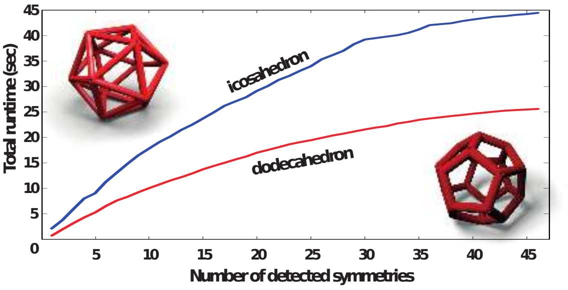


Figure 10: ***Runtimes for the 'icosahedron' and 'dodecahedron'.*** *The plot shows the runtime accumulating while detecting the symmetries. The algorithm found exactly the 46 symmetries of these shapes: fifteen reflections, ten 3-fold rotations, six 5-fold rotations and fifteen 2-fold rotations. Note that less effort is needed the further the algorithm progresses. This is due to the fact that regions around the previously detected symmetries are being 'carved-out' from the search space. Our runtimes are competitive with those reported by Martinet* et al. *[MSHS06]. See text for details.*

4.6. Determining how symmetric a shape is

Some applications only require to determine if a shape is symmetric or not. For example, in the case of quadrupeds, their shape will be symmetric if they are in the natural pose. To demonstrate this we took the 'Fourlegs' class from the COSEG dataset [WAvK*12], which includes 20 shapes of various quadrupeds. We ran our algorithm and found the best symmetry for each of the shapes. Figure 1 shows some of the shapes in increasing distortion order.

Several observations can be made. First, observe that the algorithm indeed found the most prominent symmetry in each case. The color code in Figure 1 encodes the difference between the corresponding points, measured as the absolute difference between a point on the original shape and its corresponding point on the transformed shape. As shown in the first example (pig), the shape is almost perfectly symmetric. In the next example (cow), the head is slightly tilted and indeed this is correctly detected by our algorithm. Note that this distortion does not affect the quality of the recovered plane of symmetry. The third example (horse) shows that the algorithm properly detects the symmetry plane despite the fact that the legs are not symmetric. The last two examples show shapes that are not symmetric and indeed the animals are not in their natural pose and the distortion of the best symmetry found by the algorithm is high.

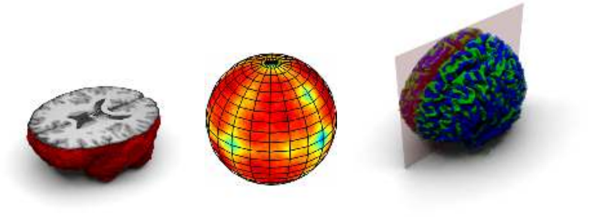


Figure 11: ***Symmetry in an MRI scalar volume. Left:*** *an illustration of the bottom half of a volumetric MRI of a brain.* ***Center****: visualization of the reflection distortion of the volume.* ***Right:*** *The detected reflection symmetry plane, along with the original image (green) and its reflected version (blue), visualized as iso-surfaces.*

4.7. Symmetry in an MRI scalar volume

So far we assumed that the input originates from solid 3D shapes. However, our method can handle general scalar volumes, which are common in 3D medical imaging. We used a simulated volume of a normal brain from [CKK*97], in the T1 MRI modality, slice thickness of 1mm, 3% noise and 20% non-uniformity. The best symmetry detected by our method discovered the bilateral symmetry of the left and right brain hemispheres (see Figure 11), although the model is far from being perfectly symmetric.

5. Summary

We presented a fast algorithm for global approximate 3D symmetry detection that is guaranteed to find all approximate symmetries of a volumetric representation of a shape within a user specified accuracy. The algorithm is robust to noise and is fast in practice, taking about two seconds to detect a symmetry.

A key contribution of our work is a proof that the density of the net depends on the total variation of the shape. Therefore, the best transformation on the net is within an approximation constant from the optimal transformation. We further show the use of TSDF representations to control the shape total variation, and hence the sampling density. The algorithm is further accelerated using sub-linear sampling that randomly examines only a small number of points, which makes the algorithm find symmetry with overwhelmingly high probability.

Several experiments asses the performance of the algorithm, including very complicated shapes with tens of symmetries. Unlike previous work, we include an experiment on a large set of shapes and show that the method scales well.

The proposed algorithm can be modified and generalized in the following manners. First, it could handle richer transformation groups like Euclidean transformations E(3), enabling rigid registration of shapes. Second, it could be applied to a given part of a shape (as was done in [KRTA13]), and act as a component in a partial symmetry detector.

References

[AMCO08] Aiger D., Mitra N. J., Cohen-Or D.: 4-points congruent sets for robust pairwise surface registration. In *ACM Transactions on Graphics (TOG)* (2008), vol. 27, ACM, p. 85. 2

[Ata85] Atallah M. J.: On symmetry detection. *Computers, IEEE Transactions on C-34*, 7 (1985), 663–666. 2

[CKK*97] Cocosco C. A., Kollokian V., Kwan R. K.-S., Pike G. B., Evans A. C.: Brainweb: Online interface to a 3d mri simulated brain database. *NeuroImage 5* (1997), 425. 7, 12

[ELPZ97] Eldar Y., Lindenbaum M., Porat M., Zeevi Y. Y.: The farthest point strategy for progressive image sampling. *Trans. on Imag. Proc. 6*, 9 (1997), 1305–1315. 5

[HOK10] Hel-Or H., Kaplan C. S.: *Computational symmetry in computer vision and computer graphics*. Now publishers Inc, 2010. 2

[KFR04] Kazhdan M., Funkhouser T., Rusinkiewicz S.: Symmetry descriptors and 3d shape matching. In *SGP* (2004), ACM, pp. 115–123. 2, 10, 11

[KLAB] Korman S., Litman R., Avidan S., Bronstein A.: Probably approximately symmetric webpage. www.eng.tau.ac.il/~simonk/ShapeSymmetry/index.html. 7, 9

[KRTA13] Korman S., Reichman D., Tsur G., Avidan S.: Fast-match: Fast affine template matching. In *CVPR* (2013), IEEE, pp. 2331–2338. 3, 8, 12

[LCDF10] Lipman Y., Chen X., Daubechies I., Funkhouser T. A.: Symmetry factored embedding and distance. *ACM Trans. Graph. 29*, 4 (2010). 2

[MGP06] Mitra N. J., Guibas L. J., Pauly M.: Partial and approximate symmetry detection for 3d geometry. *ACM Trans. Graph. 25*, 3 (2006), 560–568. 2

[MPWC12] Mitra N. J., Pauly M., Wand M., Ceylan D.: Symmetry in 3d geometry: Extraction and applications. In *EUROGRAPHICS State-of-the-art Report* (2012). 2

[MSHS06] Martinet A., Soler C., Holzschuch N., Sillion F.: Accurate detection of symmetries in 3d shapes. *ACM Transactions on Graphics (TOG) 25*, 2 (2006), 439–464. 2, 11

[OSG08] Ovsjanikov M., Sun J., Guibas L.: Global intrinsic symmetries of shapes. In *Proceedings of the Symposium on Geometry Processing* (2008), SGP '08, pp. 1341–1348. 2

[PSG*06] Podolak J., Shilane P., Golovinskiy A., Rusinkiewicz S., Funkhouser T.: A planar-reflective symmetry transform for 3d shapes. In *ACM Transactions on Graphics (TOG)* (2006), vol. 25, ACM, pp. 549–559. 2

[RBBK10] Raviv, Bronstein A. M., Bronstein M. M., Kimmel R.: Full and partial symmetries of non-rigid shapes. *Intl. Journal of Computer Vision (IJCV) 89*, 1 (2010), 189–39. 2

[SS97] Sun C., Sherrah J.: 3d symmetry detection using the extended gaussian image. *IEEE Trans. Pattern Anal. Mach. Intell. 19*, 2 (1997), 164–168. 2

[Val84] Valiant L. G.: A theory of the learnable. *Communications of the ACM 27*, 11 (1984), 1134–1142. 3

[WAvK*12] Wang Y., Asafi S., van Kaick O., Zhang H., Cohen-Or D., Chen B.: Active co-analysis of a set of shapes. *ACM Transactions on Graphics (TOG) 31*, 6 (2012). 1, 7, 11

[WWV85] Wolter J., Woo T., Volz R.: Optimal algorithms for symmetry detection in two and three dimensions. *The Visual Computer 1*, 1 (1985), 37–48. 2

[ZPA95] Zabrodsky H., Peleg S., Avnir D.: Symmetry as a continuous feature. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 17*, 12 (1995), 1154–1166. 2, 4, 10, 11