Alignment of 3D models

Mohamed Chaouch, Anne Verroust-Blondet

Alignment of 3D models

Mohamed Chaouch *, Anne Verroust-Blondet

INRIA, Paris - Rocquencourt
78153 Le Chesnay, FRANCE

Abstract

In this paper we present a new method for alignment of 3D models. This approach is based on two types of symmetries of the models: the reflective symmetry and the local translational symmetry along a direction. Finding all reflective symmetries of a shape is much more difficult than simply checking whether a given set of symmetries exists. Inspired by the work on the Principal Component analysis (PCA), we select the best optimal alignment axes within the PCA axes, the plane reflection symmetry being used as a selection criterion. This pre-processing transforms the alignment problem into an indexing scheme based on the number of the retained PCA-axes. In order to capture the local translational symmetry of a shape along a direction, we introduce a new measure we call the local translational invariance cost (*LTIC*). The mirror planes of a model are also used to reduce the number of candidate coordinate frames when looking for the one which corresponds to the user's perception. Experimental results show that the proposed method finds the rotation that best aligns a 3D mesh.

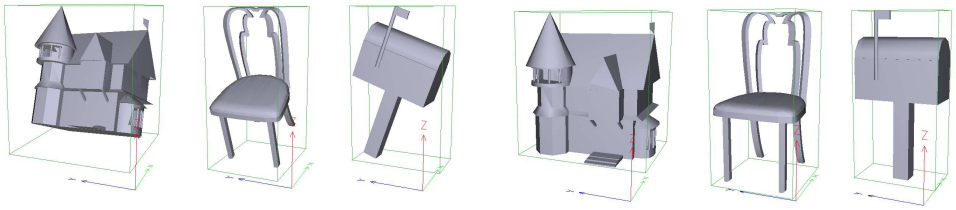
Key words: 3D alignment; Principal Component Analysis; Symmetry detection; 3D shape retrieval

1 Introduction

Normalization of 3D models is a common pre-processing stage in many applications in computer graphics, such as, visualization, 3D object recognition, 3D shape matching and retrieval [2,19,22,26]. 3D models are generally given in arbitrary scale, position and orientation in 3D-space. Most of the methods do not satisfy geometrical invariance; hence it is important to normalize the models into a canonical coordinate frame before any processing. The normalization consists of two steps:

* Corresponding author.
Email addresses: mohamed.chaouch@inria.fr (Mohamed Chaouch), anne.verroust@inria.fr (Anne Verroust-Blondet).

the alignment to determine the pose-invariant and the scaling to make the scale-invariant. The alignment is the most difficult point in the normalization process.
To perform an alignment, a concatenation of isometries in 3D-space (translation, rotation and reflection) must be selected to determine the canonical coordinate system. In most of the methods, the center of gravity of the model is chosen as the origin to secure the translation invariance. However, the choice of a suitable rotation is still a well discussed topic [2,5,11,16,19,24,26]. Note that the alignment problem addressed in this paper is different from the alignment approaches of [5,11], where the purpose is to find the best alignment between two given 3D models. Here, we want to compute an intrinsic global coordinate system for each 3D object.



Alignments of 3D models with CPCA. Alignments of 3D models with our method.

Fig. 1. Comparing CPCA based alignment and our alignment. In each case, green, blue and red arrows represent the principal axes (CPCA) and the suitable axes (our method).

When looking at a 3D model, one can say whether it is well aligned or not and knows, in most of the cases, how to find its good alignment. When the 3D model has symmetries, it is aligned with particular axes or symmetry planes. This is confirmed by Ferguson [9] who noticed that symmetry detection is a key part of human perception and this fact has guided Podolak et al. [17] when introducing principal symmetry axes. Our goal is to find a method that best aligns any 3D model (an alignment similar to what a human would select - see left part of Figure 1) and will consequently align two similar 3D models in the same way. In this paper, we show that by detecting *the planar reflection symmetries* we can select a set of good alignment axes. However, this method is guaranteed to give the correct alignment for only some cases and keeping only this type of symmetry is insufficient for computing the best alignment for any 3D model. An alternative method is to detect also *the local translational symmetry* that has an interesting semantic meaning: the object has the same geometrical properties in different parts along a given direction.
To build our general alignment algorithm, we first focus on discrete detection of plane reflection symmetries and classify a model in terms of its symmetry group and the number of its mirror planes. This classification is used to select the good alignment axes among those found by the principal components analysis (PCA). Then we introduce local translational invariance cost ($LTIC$) that measures the invariance of a model with respect to local translation along a given direction. This measure is used to compute the remaining alignment axes when the model has at most one good alignment axis given by the PCA. This paper is an extension of [4]: it gives more details on our alignment algorithm and discuss on methods computing reference frames from our alignment axes.
We first review related work on alignment and symmetry detection for 3D models

in section 2. Then we present our selection of the best alignment axes within the PCA-eigenvectors by analyzing the plane reflection symmetries (Section 3), and we describe our alignment method (Section 4). Experimental results evaluating our method are presented in section 5. Finally, we discuss on the ordering and the orientation of the alignment axes in section 6 and we conclude in section 7.

2 Related Work

The most well-known approach computing the alignment of 3D objects is the principle component analysis method (PCA) [2,16,19,24,26], which is based on the computation of moments of 3D models. After a translation of the center of mass to the origin of the coordinate system, three principal axes computed with PCA are used to determine the orientation. Experiences show that PCA-alignment has two disadvantages: (i) It is often imprecise and can produce poor alignments; (ii) The principal axes are not always good at aligning orientations of different models within the same semantic class (as noticed by Chen et al. [6] on the mug example). Podolak et al. [17] introduce a planar reflective symmetry transform (PRST) that computes a measure of the reflectional symmetry of a 3D shape with respect to all possible planes. They use it to define two new concepts for the global coordinate system, *the center of symmetry* and *the principal symmetry axes*. The principal symmetry axes are the normals of the orthogonal set of planes with maximal symmetry, and the center of symmetry is the intersection of those three planes. This approach has been improved by Rustamov with the augmented symmetry transform [18].
Other methods finding symmetries in 3D models have been presented. These include Minovic et al. [14], who compute symmetries of a 3D object represented by an octree. Their method is based on the computation of a *principal octree* aligned with the principal axes. Then they compute a measure of symmetry, the *symmetry degree*, reasoning with the number of distinct eigenvalues associated to the principal axes. Furthermore, Sun and Sherrah [21] convert the symmetry detection problem to the correlation of the Gaussian image. Then rotational and reflectional symmetry directions are determined using the statistics of the orientation histogram. Finally, Martinet et al. [13] use generalized moments to detect perfect symmetries in 3D shapes and Mitra et al. [15] and Simari et al. [20] compute partial and approximate symmetries in 3D objects.
Our goal is to align 3D models using their planar symmetry properties. Our method must be such that similar objects (i.e., objects belonging to the same semantic class) have similar alignments. As noticed in [14], any plane of symmetry of a body is perpendicular to a principal axis. As a result, for models that have plane reflection symmetries, some PCA-coordinate planes coincide with some mirror planes. Therefore, we have chosen to use the PCA, not for global alignment, but for selection of robust partial alignment features of a model (i.e., only the principal axes that we consider good for a perfect alignment).

Given a 3D model, the first key idea is to test the reflection symmetry of the PCA-coordinate planes. According to the result of this test, we select a set of principal axes and use them in our alignment method. When the model has at least two orthogonal mirror symmetries, the PCA gives the good alignment. In the other cases we use the local translational invariance cost along a direction to compute the good alignment axes.
Before describing our alignment procedure, let us classify the 3D polygonal models with respect to their plane reflection symmetry and select classes of objects where PCA gives a good alignment.

3 Symmetry & 3D Objects

In what follows $\mathscr{M}$ denotes a 3D polygonal model represented by its surface $\mathfrak{S}$ composed of a set of triangular facets $\mathfrak{T} = \{T_1, ..., T_{N_T}\}$. We study the reflection planes in the symmetry groups [8], and use them to discriminate different classes of mirror symmetry. Then, we discuss for each class when the PCA alignment has good properties w.r.t. the planar reflective symmetry.

3.1 Plane Reflection Symmetry Analysis

A plane reflection symmetry is defined by a mirror plane π that can be parameterized by its unit normal $\mathbf{n}$ and its scalar distance δ from the origin. This symmetry associates to each point $\mathbf{p}$ of $\mathfrak{S}$ a mirror reflection point $\mathbf{q}$ on $\mathfrak{S}$ defined by: $\mathbf{q} = \mathbf{p} - 2\,(\mathbf{n}^T \cdot \mathbf{p} - \delta)\,\mathbf{n}$.

According to Dubrovin et al. [8], studying the plane reflection symmetries of a 3D polyhedral model and the types of symmetry groups, we can distinguish five classes of 3D polyhedral models (see examples in Figure 2):

(1) $\mathbf{G_C}$: 3D models that have cyclic symmetry. They have n mirror planes $(n > 1)$ that pass through a fixed axis, such as a regular n-pyramid, a simple rectangular table ($n = 2$) and a simple square table ($n = 4$). $\mathbf{G_C}$ is split into two subclasses, $\mathbf{G_C^{odd}}$ and $\mathbf{G_C^{even}}$, according to the parity of n.
(2) $\mathbf{G_D}$: 3D models that have dihedral symmetry. They have n mirror planes $(n > 1)$ that pass through a particular axis with one mirror plane perpendicular to the axis, such as a regular n-prism or regular n-bipyramid. $\mathbf{G_D}$ is split into two subclasses, $\mathbf{G_D^{odd}}$ and $\mathbf{G_D^{even}}$, according to the parity of n.
(3) $\mathbf{G_R}$: 3D models that have rotation symmetry such as the five convex regular polyhedra called platonic solids. It contains three sub-groups: $\mathbf{G_T}$ of tetrahedral symmetry (6 mirror planes), $\mathbf{G_O}$ of octahedral symmetry (9 mirror planes) and $\mathbf{G_I}$ of icosahedron symmetry (15 mirror planes).

(4) $\mathbf{G_U}$: 3D models that have only one plane reflection symmetry. This is the case for many natural and man-made objects such as airplanes, animals, humans, chairs, cars, etc.
(5) $\mathbf{G_Z}$: 3D models that don't have any plane reflection symmetry, such as plants and trees.

This classification is valid for perfect plane reflection symmetries. We will extend it to approximate mirror reflections (see section 4.1).

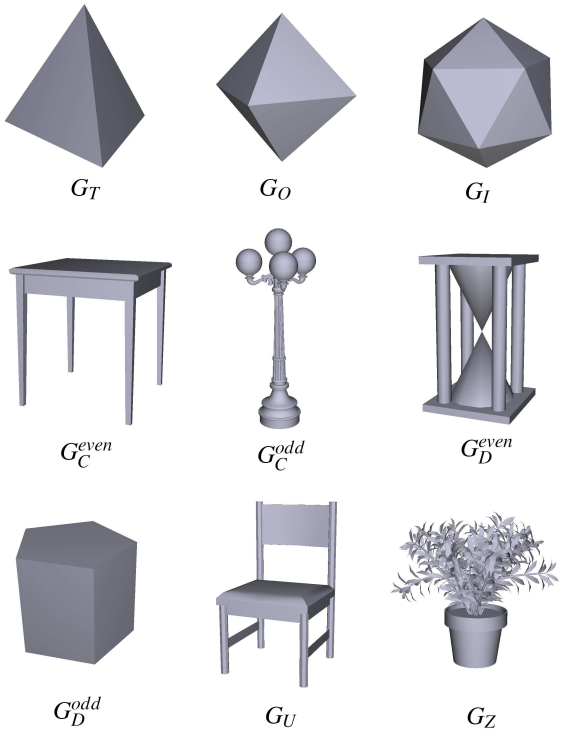


Fig. 2. Models belonging to different classes of mirror symmetry: Tetrahedron $\in G_T$, Octahedron $\in G_O$, Icosahedron $\in G_I$, Table $\in G_C^{even}$, Street-light lamp $\in G_C^{odd}$, Hourglass $\in G_D^{even}$, 5-prism $\in G_D^{odd}$, Chair $\in G_U$, Plant $\in G_Z$.

3.2 Principal Components & Plane Reflection Symmetry Analysis

In this section, we explore the relation between the principal components analysis (PCA) and the plane reflection symmetry analysis. In our proofs, we have retained the "Continuous principal components analysis" (CPCA) [24] because it appears to be more complete and the most stable of all the PCA-approaches we have studied. CPCA computes three orthogonal eigenvectors of the covariance matrix C.
As noticed in [14], when π is a mirror plane of $\mathfrak{S}$ and $\mathbf{n}$ is the unit normal of π, then π passes through the center of gravity of $\mathfrak{S}$ and $\mathbf{n}$ is an eigenvector of the covariance matrix C that is a principal component axis of $\mathfrak{S}$ (see in appendix a proof for the continuous case). If $\mathfrak{S}$ has n mirror planes that pass through a fixed axis (as

in the cases $\mathbf{G_C}, \mathbf{G_D}, \mathbf{G_R}$ of section 3.1), then n different eigenvectors are associated to the same eigenvalue: in this case, $\mathfrak{S}$ has a discrete rotational symmetry of order n $(n > 1)$ with respect to the same axis.
Besides that, we note that if $\mathfrak{S}$ has a set of dual orthogonal reflection planes, the CPCA detects at least two orthogonal normals associated to one dual orthogonal mirror plane of this set. In what follows, for each class described in section 3.1 we discuss the position of these vectors with the reflection symmetries:
- If $\mathscr{M} \in \mathbf{G_C}$ and n is even ($\mathscr{M} \in \mathbf{G_C^{even}}$), then the CPCA detects two orthogonal normals associated to two orthogonal reflection planes and the axis of the axial symmetry (the intersection of the mirror planes). When $\mathscr{M} \in \mathbf{G_C}$ and n is odd ($\mathscr{M} \in \mathbf{G_C^{odd}}$), the CPCA gives only one normal associated to one mirror plane.
- If $\mathscr{M} \in \mathbf{G_D}$, then the CPCA gives at least two orthogonal normals; the first is associated to one of the n mirror planes and the second is the axis of the axial symmetry. If, furthermore, n is even, then the CPCA detects the third axis associated to the mirror plane that is orthogonal to the first given mirror.
- If $\mathscr{M} \in \mathbf{G_O}$, then the CPCA detects three orthogonal normals associated to three orthogonal reflection planes, contrarily to the cases of $\mathbf{G_T}$ and $\mathbf{G_I}$, where the CPCA gives only one normal associated to one mirror plane.
- If $\mathscr{M} \in \mathbf{G_U}$, then the CPCA gives only one normal associated to its mirror plane.

Symmetry	Class	Number of mirror planes	Number of CPCA axes retained
Cyclic	$\mathbf{G_C^{even}}$	n with $n > 1$	2
symmetry	$\mathbf{G_C^{odd}}$	n with $n > 1$	1
Dihedral	$\mathbf{G_D^{even}}$	$n+1$ with $n > 1$	3
symmetry	$\mathbf{G_C^{odd}}$	$n+1$ with $n > 1$	2
Rotational	$\mathbf{G_T}$	6	1
symmetry	$\mathbf{G_O}$	9	3
	$\mathbf{G_I}$	15	1
One plane reflection	$\mathbf{G_U}$	1	1
No plane reflection	$\mathbf{G_Z}$	0	0

Table 1
Plane reflection symmetry types and Principal Components Analysis

Thus, when $\mathscr{M} \in \mathbf{G_C^{even}} \bigcup \mathbf{G_D} \bigcup \mathbf{G_O}$, the CPCA detects at least two good alignment axes and when $\mathscr{M} \in \mathbf{G_C^{odd}} \bigcup \mathbf{G_T} \bigcup \mathbf{G_I} \bigcup \mathbf{G_U}$, the CPCA gives only one good alignment axis. Finally, when $\mathscr{M} \in \mathbf{G_Z}$, the CPCA doesn't detect any good alignment axis. We summarize our discussion using the function $N_{GA}(\mathscr{M})$, which accounts the number of the good alignment axis computed by the CPCA. Given the symmetry class of the model $\mathscr{M}$, $N_{GA}(\mathscr{M})$ is defined as follows:

$$N_{GA}(\mathscr{M}) = \begin{cases} 2^+ & if\ \mathscr{M} \in \mathbf{G_C^{even}} \bigcup \mathbf{G_D} \bigcup \mathbf{G_O} \\ 1 & if\ \mathscr{M} \in \mathbf{G_C^{odd}} \bigcup \mathbf{G_T} \bigcup \mathbf{G_I} \bigcup \mathbf{G_U} \\ 0 & if\ \mathscr{M} \in \mathbf{G_Z} \end{cases}$$

$N_{GA}(\mathscr{M})$ will guide the computation inside our alignment algorithm.

4 Alignment of 3D Objects

Given a 3D model $\mathscr{M}$, we aim to develop a general algorithm that computes $N_{GA}(\mathscr{M})$ and selects the set of good alignment axes given by the CPCA, and, if necessary, computes the rest of alignment axes in order to complete the pose coordinate system.

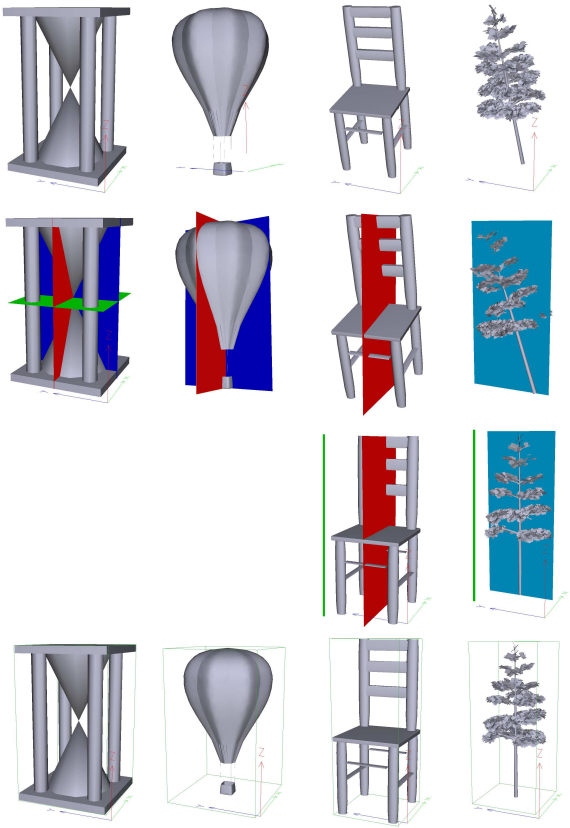


Fig. 3. Alignments of models with different N_{GA} using our method: Hot air balloon and Hourglass models $N_{GA} = 2^{+}$, Chair model $N_{GA} = 1$, Tree model $N_{GA} = 0$. ***Row 1***: CPCA Alignments, ***Row 2***: Testing the reflection symmetry for CPCA-coordinate planes, ***Row 3***: Computing the direction with maximal local translational invariance, ***Row 4***: Our Alignment results.

We describe here the main steps of our alignment algorithm and detail them in the next subsections (see on Figure 3 an illustration of the algorithm applied to four models of different N_{GA}-values).

Algorithm: Compute good alignment axes

(1) Translate the input 3D model $\mathscr{M}$ from its center of gravity to the origin of world coordinate system.
(2) Compute the three CPCA eigenvectors $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$ of the covariance matrix C and rotate the translated model in the new CPCA-coordinate system $R\,(\mathbf{v}_1;\mathbf{v}_2;\mathbf{v}_3)$

which has the eigenvectors as rows.

(3) Test the reflection symmetry for each coordinate plane normal to a CPCA-axis, (xy-, yz-, zx-coordinate plane) and deduce $N_{GA}(\mathscr{M})$. We elaborate on this step in section 4.1.1.

(4) Select the good alignment axis/axes according to the value of $N_{GA}(\mathscr{M})$:
$(if\ N_{GA}(\mathscr{M}) = 2^{+})$ Return the three good alignment axes $R_{ga}\ (\mathbf{n}_1;\mathbf{n}_2;\mathbf{n}_3) = (\mathbf{v}_1;\mathbf{v}_2;\mathbf{v}_3)$.
$(if\ N_{GA}(\mathscr{M}) = 1)$ Return $\mathbf{n}_1 \in \{\mathbf{v}_1,\mathbf{v}_2,\mathbf{v}_3\}$ the normal of the unique mirror plane as the first good alignment axis and rotate the 3D model in the new coordinate system $R'\ (\mathbf{n}_1;\mathbf{v}_2;\mathbf{v}_3)$ if $\mathbf{n}_1 = \mathbf{v}_1$, in $R'\ (\mathbf{n}_1;\mathbf{v}_3;\mathbf{v}_1)$ if $\mathbf{n}_1 = \mathbf{v}_2$, in $R'\ (\mathbf{n}_1;\mathbf{v}_1;\mathbf{v}_2)$ if $\mathbf{n}_1 = \mathbf{v}_3$.
$(if\ N_{GA}(\mathscr{M}) = 0)$ Return $\mathbf{n}_1$ the normal of the plane with maximal reflection symmetry (see section 4.1.2) as the first good alignment axis and rotate the 3D model in a new coordinate system $R'\ (\mathbf{n}_1;\mathbf{v'}_2;\mathbf{v'}_3)$.

(5) If $N_{GA}(\mathscr{M}) \in \{0,1\}$, Compute the direction vector with maximal local translational invariance cost as will be shown in the algorithm of section 4.2.3 and return the three good alignment axes $R_{ga}\ (\mathbf{n}_1;\mathbf{n}_2;\mathbf{n}_3)$.

Remark: without loss of generality, we suppose that, when the CPCA computes exactly two normals associated to two mirror planes (i.e. $N_{GA}(\mathscr{M}) = 2$), the two corresponding axes are n_1 and n_2.

4.1 Plane Reflection Symmetry

There are two approaches for measuring imperfect symmetry:
-*The symmetry distance* of a shape with respect to a given symmetry is the minimum mean squared distance from the given shape to its perfectly symmetric shape. This measure estimates the symmetry in 3D surface points. While this distance is precise and robust for measuring symmetry, it is expensive for large models.
- *The symmetry descriptor similarity* of a shape with respect to a given symmetry is the distance between a shape descriptor of the given shape and that of its perfectly symmetric shape. This measure has been proven useful in order to approximate the symmetry distance. The efficiency of the symmetry description in 3D space enables a fast comparison of the amount of reflection symmetries with respect to several planes.

4.1.1 Continuous Symmetry Distance

Let $\mathfrak{S}_\gamma$ be the reflective surface of $\mathfrak{S}$ with respect to a plane γ. It has N_T triangular facets as $\mathfrak{S}$. Following previous works on distance estimation between 3D surfaces [1,7] and on symmetry distance [25], we define the continuous symmetry distance

CSD_γ of $\mathfrak{S}$ with respect to a plane reflection γ as:

$$CSD_\gamma(\mathfrak{S}) = \frac{1}{\mathscr{A}} \iint_{\mathbf{p}\in\mathfrak{S}} d(\mathbf{p}, \mathfrak{S}_\gamma)\, ds, \tag{1}$$

where $\mathscr{A}$ denotes the area of $\mathfrak{S}$ and d is the distance between a point $\mathbf{p}$ of $\mathfrak{S}$ and $\mathfrak{S}_\gamma$, such that $d(\mathbf{p}, \mathfrak{S}_\gamma) = \min_{\mathbf{p}'\in\mathfrak{S}_\gamma} \|\mathbf{p}-\mathbf{p}'\|_2$, $\|.\|_2$ being the usual Euclidean norm. The integral of the symmetry error over $\mathfrak{S}$ is computed by summing the contributions of the triangular facets of $\mathfrak{S}$. We obtain a more precise result by taking into account all points of $\mathfrak{S}$. The computation of these integrals is slightly more expensive than the discrete case as stated by Zabrodsky et al. [25]. However, in order to obtain correct point-$\mathfrak{S}_\gamma$ distances, each triangular facet is sampled uniformly and $\mathfrak{S}$ is represented by N_S sampling points. The integral over each triangular facet T_i of $\mathfrak{S}$ is then approximately done with sums of integrals over triangles obtained by sampling T_i. In fact, for each vertex $\mathbf{v}$ of each sample triangle T of T_i, it is necessary to calculate the distance from $\mathbf{v}$ to all facets of $\mathfrak{S}_\gamma$ in order to find the minimum distance. This leads to a complexity $\mathscr{O}(N_T N_S)$, which is expensive for large models. This complexity has been reduced in [7] by using a local search processing in order to decrease the number of point-triangle distance evaluations. The idea is to partition the bounding box into cubic cells and use them in an indexing scheme for the fast search of the nearest triangle of $\mathfrak{S}_\gamma$ to the sampling point.
If γ is a perfect mirror plane of $\mathfrak{S}$, then $CSD_\gamma(\mathfrak{S})$ is null. As we want to retain the quasi-perfect mirror planes, we will approximate this definition. Then we say that γ is a mirror plane of $\mathfrak{S}$ when $SD_\gamma(\mathfrak{S}) < \varepsilon$ ($\varepsilon \simeq 0$). This test will be used in step 3 of the algorithm described in section 4 in order to select the mirror planes among the coordinate planes normal to a CPCA-axis.

4.1.2 Symmetry Descriptors

The symmetry descriptor represents the symmetries of a given model with respect to several planes in 3D space. It is generally associated to a given shape descriptor that represents a model with a spherical function or a 3D function that rotates with the model. Kazhdan et al. [12] define a symmetry descriptor using the planes through its center of gravity. Podolak et al. [17] extend this work by considering symmetries with respect to all possible planes through a model's bounding volume. Following Kazhdan et al. [12] and using the fact that mirror planes are orthogonal to CPCA axes when they exist, we consider a symmetry descriptor that represents the symmetries of a 3D model with respect to planes passing through its center of gravity and in the angular neighborhood of the planes normal to the CPCA-axes.
Measuring imperfect symmetry is used in step 4 ($N_{GA}(\mathscr{M}) = 0$) of the algorithm described in section 4. Specifically, given the symmetry descriptor values, we select the good axis by finding the plane with maximal symmetry.

Traditionally, in geometry, the translational symmetry is the invariance of an infinite object with respect to a particular translation. We extend this definition to a finite object, in particular, to a 3D model. We define here the local translational symmetry that will be used in this section. This symmetry implies that a 3D model has the same geometrical properties in different parts along a given direction. Appropriate shape descriptions of the model will be introduced to evaluate the invariance of geometrical properties.
Finding the direction that maximizes the local translational invariance is the last step in our general alignment scheme (when $N_{GA}(\mathscr{M}) \in \{0,1\}$). More precisely, we look for local translational symmetries with respect to all directions perpendicular to the first good alignment axis $\mathbf{n}_1$ computed in step 4 of the algorithm of section 4 (see Figure 4). For that purpose, we need to compute a shape descriptor f defined over a one-dimensional interval that represents a 3D model along a given direction and to define a measure of symmetry for f with respect to local translation along this direction. First of all, we describe a method selecting the direction with maximal translational invariance.

4.2.1 Shape Description over 1-D Space

Let $\mathbf{d} \in \mathbb{R}^3$ be a unit direction vector and $\pi_{\mathbf{d}}(\rho)$, $\rho \in \mathbb{R}$, be a family of planes perpendicular to $\mathbf{d}$ and at the signed distance ρ from the center of the coordinate system. We represent $\mathfrak{S}$ as follows, $I_{\mathbf{d}}$ being the interval defined by the limits of the 3D surface $\mathfrak{S}$ in the direction $\mathbf{d}$:

$$\mathfrak{S} = \bigcup_{\rho \in I_{\mathbf{d}}} \mathscr{S}_{\mathbf{d}}(\rho),$$

where $\mathscr{S}_{\mathbf{d}}(\rho)$ is the 3D sub-shape of $\mathfrak{S}$ limited by the planes $\pi_{\mathbf{d}}(\rho - \delta)$ and $\pi_{\mathbf{d}}(\rho + \delta)$, $\delta \in \mathbb{R}$. In this representation, $\mathfrak{S}$ is viewed as the union of bands of surface of width 2δ and limited by planar curves belonging to planes perpendicular to $\mathbf{d}$.
In what follows, $f_{\mathbf{d}}$ denotes a function defined on the interval $I_{\mathbf{d}}$ and having values on a scalar, or vector space, such that $f_{\mathbf{d}}(\rho)$ is a shape descriptor of $\mathscr{S}_{\mathbf{d}}(\rho)$ for any $\rho \in I_{\mathbf{d}}$. Methods computing the shape descriptor $f_{\mathbf{d}}$ of $\mathfrak{S}$ are given in section 4.2.4.

4.2.2 Local Translational Invariance Cost

Definition 1. *Given a shape descriptor $f_{\mathbf{d}}$ of $\mathfrak{S}$ defined over an interval $I_{\mathbf{d}}$ and a unit vector $\mathbf{d}$, we say that $f_{\mathbf{d}}$ has* local translational invariance along $\mathbf{d}$ in an interval $I \subset I_{\mathbf{d}}$ *if for all* $\rho, \rho' \in I,\ f_{\mathbf{d}}(\rho) = f_{\mathbf{d}}(\rho')$.

In order to measure the local translational symmetry of a shape descriptor $f_{\mathbf{d}}$, we

detect the maximal sub-intervals I_i of I_d such that $f_{\mathbf{d}}$ has local translational invariance along $\mathbf{d}$ in I_i. The cost of this symmetry is defined as follows:

Definition 2. *Given a shape descriptor $f_{\mathbf{d}}$ defined over $I_{\mathbf{d}}$, the* local translational invariance cost *(LTIC)* of $f_{\mathbf{d}}$ along $\mathbf{d}$ *is the sum of the lengths of the maximal intervals I_i of I_d where $f_{\mathbf{d}}$ has local translational invariance along $\mathbf{d}$ in I_i:*

$$LTIC(f_{\mathbf{d}}) = \sum_{I_i \in \mathscr{I}} \mathscr{L}(I_i) \ , \tag{2}$$

where $\mathscr{L}(I_i)$ is the length of I_i and $\mathscr{I} = \{I_i \subset I_d | I_i$ maximum; $f_{\mathbf{d}}$ has local translational invariance along $\mathbf{d}$ in $I_i\}$.

4.2.3 LTIC for Alignment

In this section, we investigate the use of the LTIC in 3D to compute a good alignment axis with respect to translational symmetry. More precisely, we want to select the second alignment axis by finding the direction with maximal local translational invariance cost among the directions perpendicular to the first axis $\mathbf{n}_1$ computed in step 4 of the algorithm of section 4.

In order to evaluate the LTIC, K direction vectors perpendicular to the first good

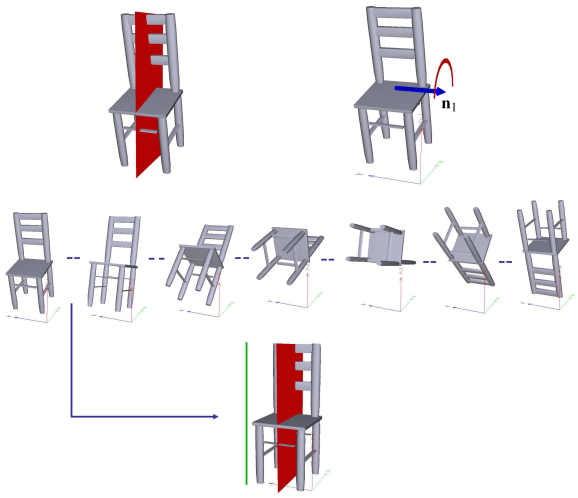


Fig. 4. Computing the good alignment axes (given the first one $\mathbf{n}_1$) for the chair model $N_{GA}(\mathscr{M}) = 1$. ***Row 1***: Return $\mathbf{n}_1$ the normal of the unique mirror plane as the first alignment axis, ***Row 2***: Rotate the model about $\mathbf{n}_1$ in order to find the rotation with maximal local translational invariance, ***Row 3***: Align chair model in the coordinate system R_{ga}.

alignment axis $\mathbf{n}_1$, are obtained by rotating the coordinate system about $\mathbf{n}_1$ as illustrated in Figure 4.

Let $\mathscr{R}^K_{\mathbf{n}}$ be the set generated by the transformation R_k which is the rotation about

$\mathbf{n}_1$ by the angle $\theta_k = \frac{\pi k}{K}$ where $0 \leq k < K$:

$$R_k = R^{\theta_k}_{(1,0,0)} \cdot R', \text{ with } \qquad R^{\theta_k}_{(1,0,0)} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\theta_k) & -sin(\theta_k) \\ 0 & sin(\theta_k) & cos(\theta_k) \end{bmatrix}$$

and R' (see section 4) is the matrix that contains $\mathbf{n}_1$ in the first row.
In what follows, we associate to each $R_k \in \mathscr{R}^K_{\mathbf{n}}$ one unit direction vector $\mathbf{d}_k$ equal to the second row of R_k. For each $\mathbf{d}_k$, a shape descriptor f_k is computed. Now, the problem of computing the good alignment axis is to find the direction $\mathbf{d}_{ga}$ or its associated rotation R_{ga}, that maximizes the $LTIC(f_k)$:

$$R_{ga} = \underset{R_k \in \mathscr{R}^K_{\mathbf{n}}}{\text{argmax}} \; LTIC(f_k). \tag{3}$$

Our algorithm for computing the good alignment axes (given the first one) can be summarized as follows:

Algorithm: Compute direction with maximal $LTIC$

(1) Translate the input 3D model $\mathscr{M}$ from its center of gravity to the origin, and scale the translated model such that the average distance of a point on the surface to the new coordinate origin is 1.
(2) Given a matrix R', for each θ_k,
   (a) Compute the transformation R_k and the associated direction vector $\mathbf{d}_k$.
   (b) Rotate the transformed model (step 1) in the coordinate system R_k in order to obtain $\mathfrak{S}_k$.
   (c) Compute the interval I_k of length $\mathscr{L}_{I_k}$ and the shape descriptor f_k defined over I_k.
   (d) Measure the $LTIC(f_k)$.
(3) Return R_{ga} associated to f_{ga} with maximal $LTIC$.

Given a matrix R', this algorithm finds the direction vector with maximal local translational invariance cost. The second good alignment axis $\mathbf{n_2}$ is the direction vector $\mathbf{d}_{ga}$ that is the second row of R_{ga} and is perpendicular to the first axis $\mathbf{n_1}$. The third good alignment axis $\mathbf{n_3}$ is naturally the third row of R_{ga}.

4.2.4 Three shape descriptor models for f_k

Suppose the surface $\mathfrak{S}_k$ is positioned in the coordinate system defined by ($\mathbf{n_1}$, $\mathbf{d}_k$, $\mathbf{n}_1 \wedge \mathbf{d}_k$), and $\mathscr{S}_k(\rho)$ and I_k are defined as in section 4.2.1 with $\mathbf{d} = \mathbf{d_k}$. Three shape descriptors models G_k, E_k and F_k (see Figure 5) are introduced to represent $\mathfrak{S}_k$. They use only one coordinate (along the axis $\mathbf{n}_1 \wedge \mathbf{d}_k$) as the axis $\mathbf{n}_1$ is already selected in the good coordinate system and $\mathbf{d}_k$-coordinate is fixed in $\mathfrak{S}_k(\rho)$.

- *Global average description G_k.*
It can be viewed as a curve given by the projections of the centers of gravity of the sub-shapes $\mathscr{S}_k(\rho)$ along the direction $\mathbf{n}_1 \wedge \mathbf{d}_k)^T$.

$$G_k(\rho) = \frac{1}{\mathscr{A}_k(\rho)} \iint_{\mathbf{p} \in \mathscr{S}_k(\rho)} (\mathbf{n}_1 \wedge \mathbf{d}_k)^T \cdot \mathbf{p} \, ds,$$

where $\mathscr{A}_k(\rho) = \iint_{\mathbf{p} \in \mathscr{S}_k(\rho)} ds$ denotes the area of $\mathscr{S}_k(\rho)$.
Using this descriptor, we measure the local invariance of the gravity along the direction $\mathbf{n}_1 \wedge \mathbf{d}_k$.

- *Global extremum description E_k.*
It represents two independent curves given respectively by the minimal and the maximal projections of the $\mathbf{n}_1 \wedge \mathbf{d}_k$ coordinates of the sub-shapes $\mathscr{S}_k(\rho)$.

$$E_k(\rho) = \begin{pmatrix} \min_{\mathbf{p} \in \mathscr{S}_k(\rho)}((\mathbf{n}_1 \wedge \mathbf{d}_k)^T \cdot \mathbf{p}) \\ \max_{\mathbf{p} \in \mathscr{S}_k(\rho)}((\mathbf{n}_1 \wedge \mathbf{d}_k)^T \cdot \mathbf{p}) \end{pmatrix}$$

Here, we measure the local invariance of the couple formed by the minimal and maximal coordinates of the surface along the direction $\mathbf{n}_1 \wedge \mathbf{d}_k$.

- *Vector shape description F_k:*
This descriptor evaluates more precisely the local invariance of the shape along the direction $\mathbf{n}_1 \wedge \mathbf{d}_k$.
Let $J_\mathbf{k}$ be the interval defi ned by the limits of the 3D surface $\mathfrak{S}_k$ in the direction $\mathbf{n}_1 \wedge \mathbf{d}_k$. The bounding box of mesh $\mathfrak{S}_k$ is partitioned into M_k cells along the direction $\mathbf{n}_1 \wedge \mathbf{d}_k$ according to a regular sampling of J_k. Then, $\mathscr{S}_k^j(\rho)$ is the intersection of the shape $\mathscr{S}_k(\rho)$ and the j^{th} cell.

$$\mathscr{S}_k(\rho) = \bigcup_{j=1}^{M_k} \mathscr{S}_k^j(\rho).$$

This descriptor represents $\mathscr{S}_k(\rho)$ with a collection of areas and averages associated to the shapes $\mathscr{S}_k^j(\rho)$, $1 \leq j \leq M_k$:

$$F_k(\rho) = \left(a_k^j(\rho), g_k^j(\rho)\right)_{M_k} \text{, where}$$

$$a_k^j(\rho) = \begin{cases} \iint_{\mathbf{p} \in \mathscr{S}_k^j(\rho)} ds & \text{if } \mathscr{S}_k^j(\rho) \neq \emptyset, \\ \emptyset & \text{otherwise.} \end{cases}$$

$$g_k^j(\rho) = \begin{cases} \frac{1}{a_k^j(\rho)} \iint_{\mathbf{p} \in \mathscr{S}_k^j(\rho)} (\mathbf{n}_1 \wedge \mathbf{d}_k)^T \cdot \mathbf{p} \, ds \text{ if } \mathscr{S}_k^j(\rho) \neq \emptyset, \\ \emptyset \qquad \text{otherwise.} \end{cases}$$

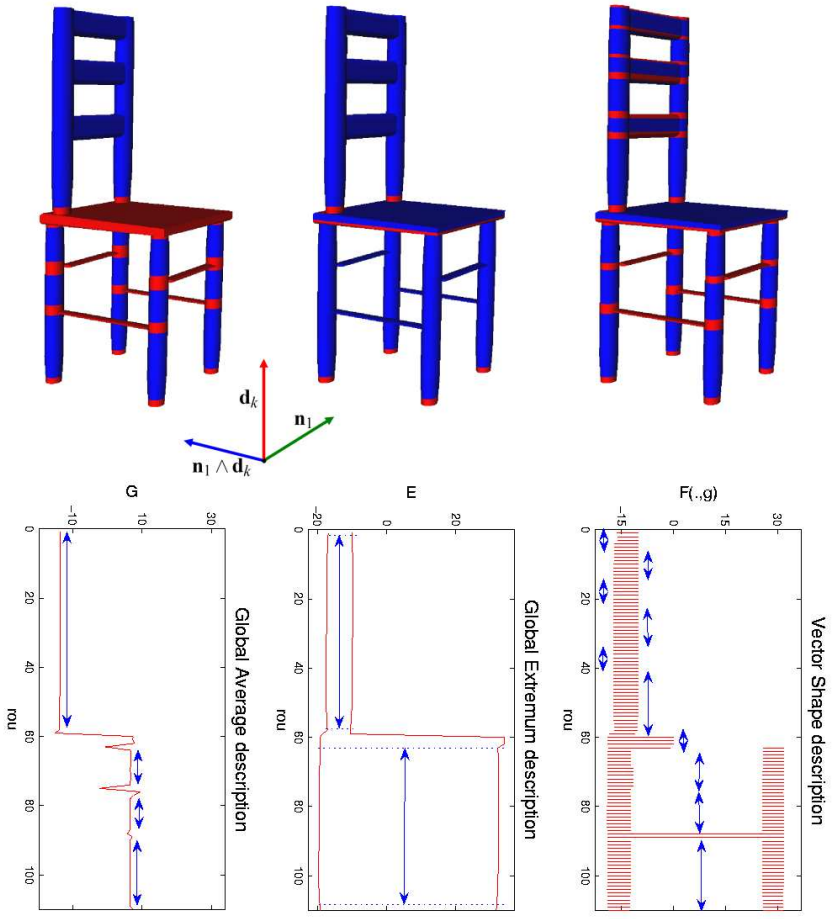


Fig. 5. The three shape descriptors G, E and F corresponding to the chair model are computed along the vertical direction $\mathbf{d}_k$. For this case, the three descriptors lead to the same direction with maximal $LTIC$. Surface positions associated to intervals of local translational invariance are colored in blue while the others are colored in red.

Discrete computation

With the introduced definitions, we deduce a discrete version of the function f_k represented on N_k points regularly sampled on I_k. To define f_k at the same scale in any direction d_k, the number of samples N_k is such that the interval $\frac{\mathscr{L}(I_k)}{N_k}$ has a fixed length 2δ (see section 4.2.1) for any orientation k. A unit of measurement $N = \frac{scale}{2\delta}$ should be fixed for all 3D models. In our case, $N = \frac{1}{2\delta}$ $(N = 32, 64, 128)$ because $\mathscr{M}$ is scaled (see the first step of the algorithm in section 4.2.3).

$$N_k = \lfloor N\, \mathscr{L}(I_k) \rfloor. \tag{4}$$

Similarly, we take $M_k = \lfloor N\, \mathscr{L}(J_k) \rfloor$ when computing F_k.
Finally, for each shape descriptor proposed here, we use a distance $dist(f_k(i), f_k(i'))$ (where $f_k(i)$, $1 \leq i \leq N_k$) in order to apply definition 1, choose a normalized error $\varepsilon_N \simeq 0$ fixed for all 3D models and consider that $f_k(i) = f_k(i')$ if $dist(f_k(i), f_k(i')) < \varepsilon_N$. In our implementation, $dist(,)$ is the usual Euclidean norm $\|.\|_1$ for the global

average description and the global extremum description. It is defined as follows for the vector shape description:

$$dist(F_k(i),F_k(i')) = \frac{d_{F_k}(i,i') + d_{F_k}(i',i)}{\mathscr{A}_k(i) + \mathscr{A}_k(i')}, \text{ where}$$

$$d_{F_k}(i,i') = \sum_{1\le j\le M_k}^{g_k^j(i)\neq\emptyset} a_k^j(i) \min_{1\le j'\le M_k}^{g_k^{j'}(i')\neq\emptyset} \|g_k^j(i) - g_k^{j'}(i')\|_1.$$

The distance $d_{F_k}(i,i')$ leads to a complexity $\mathscr{O}(M_kM_k)$. In order to reduce the computing time, we reduce the number of distance evaluations $\|g_k^j(i) - g_k^{j'}(i')\|_1$, $1 \le j, j' \le M_k$. We make an a priori coherence assumption: we suppose that the index $j_m = \text{argmin}_{1\le j'\le M_k} \|g_k^j(i) - g_k^{j'}(i')\|_1$ is not far from the index j. Thus, we first test the sparsity of $g_k^j(i')$ and we stop if $g_k^j(i') \neq \emptyset$. Otherwise, the adjacent points are processed, in order of increasing distance from $g_k^j(i)$ and we stop when we find $g_k^{j_m}(i') \neq \emptyset$. We stress that all non-tested points are farther than the found point.

5 Experimental Results

In order to evaluate our alignment algorithm, we ran the experiments with the Test Princeton 3D Shape Benchmark database [19] consisted of 907 polygonal models categorized into 92 distinct classes. Our alignment method computes three lines of support which are arbitrarily oriented and given in arbitrary order. Two alignment frames are said similar when their sets of three lines are the same.

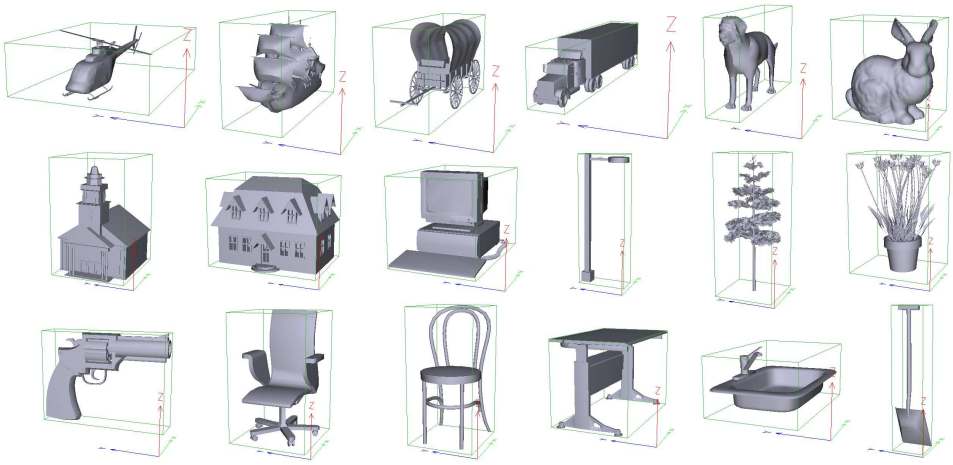


Fig. 6. Alignments of models from different classes using our method.

We found that our approach produces coordinate frames that are robust and semantically correct for most of the models. Figure 6 shows a number of models from

different classes aligned by our method. Moreover, our approach gives more precise results than CPCA, as illustrated in Figure 7, and provides similar alignments for models belonging to the same class, see for example the alignments of the mailbox class in Figure 8.

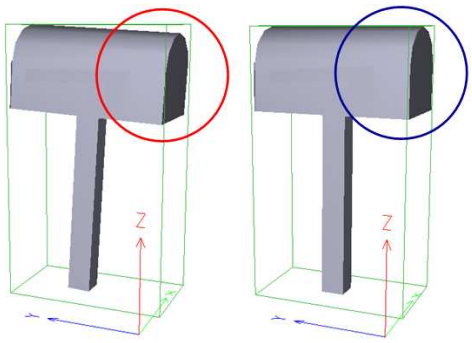


Fig. 7. Comparing the precision of CPCA (left) and our alignment of mailbox model.

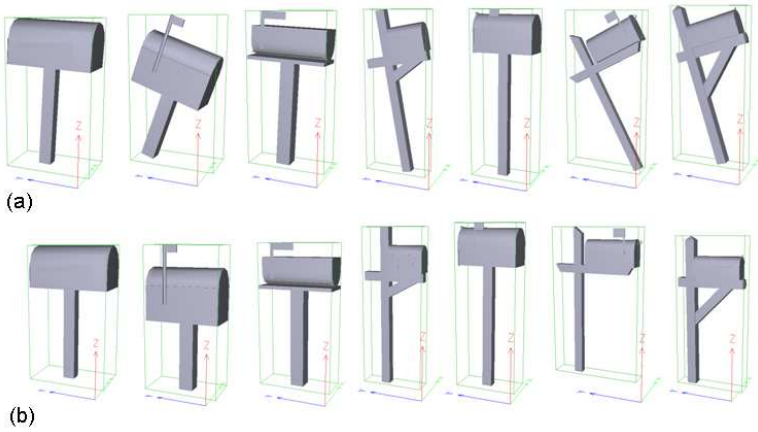


Fig. 8. Alignments of the mailbox class using CPCA (a) and our method (b).

We measured the performance of our alignment method by generating a test set of 33 distinct classes which generally are not well aligned by CPCA. Table 2 gives, for each class, the percentage of perfect alignment (i.e., accurately similar to what a human would select) and compares the results of the CPCA method to our method using the shape descriptors G, E and F introduced in section 4.2.4. To compute the percentages, we asked three individual, expert in 3D modeling, to group the models of each class into two sub-classes “well aligned” and “poorly aligned” objects. For example, all the models shown in Figure 6 (belonging to this test set) have been considered “well aligned”. The percentages appearing in the Table 2 indicate the average ratio of “well aligned” models inside each class.
By examining Table 2, we note that for the three descriptors (G, E and F), our general scheme provides better alignment performance, with perfect-alignment percentages that are generally close to 100%. When using the shape descriptor F, our method provides more accurate alignment results than using G and E.

Class	Nbr ($\mathscr{M}$)	CPCA	Our Method		
			G	E	F
Helicopter aircraft	18	77.7%	94.4%	100%	100%
Enterprise spaceship	11	36.4%	100%	100%	100%
Dog quadruped	7	00.0%	14.3%	28.6%	85.7%
Horse quadruped	6	16.7%	66.7%	66.7%	83.3%
Rabbit quadruped	4	00.0%	25.0%	75.0%	75.0%
Head body part	16	62.5%	56.2%	81.2%	100%
Skull body part	6	00.0%	16.7%	16.7%	100%
Barn building	5	40.0%	80.0%	80.0%	80.0%
Church building	4	00.0%	100%	100%	75.0%
One story building	14	35.7%	85.7%	92.9%	92.9%
Two story building	10	10.0%	80.0%	100%	100%
Chess set	9	66.7%	100%	100%	100%
Desktop computer	11	00.0%	63.6%	81.8%	81.8%
Computer monitor	13	00.0%	92.3%	92.3%	100%
Fireplace	6	00.0%	83.3%	83.3%	83.3%
Cabinet furniture	9	66.7%	100%	100%	100%
School desk furniture	4	00.0%	100%	100%	100%
Bench seat	11	00.0%	100%	100%	100%
Dining chair	11	00.0%	100%	100%	100%
Desk chair seat	15	00.0%	100%	100%	100%
Rectangular table	25	72.0%	100%	100%	100%
Handgun gun	10	00.0%	80%	90%	100%
Ladder	4	50.0%	100%	100%	100%
Streetlight lamp	8	75.0%	100%	100%	100%
Mailbox	7	14.3%	100%	100%	100%
Potted plant	26	53.8%	92.3%	88.5%	100%
Conical tree	10	70.0%	90.0%	80.0%	90.0%
Large sail boat	6	00.0%	50.0%	100%	100%
Sink	4	25.0%	75.0%	100%	100%
Slot machine	4	25.0%	75.0%	50.0%	100%
Covered wagon vehicle	5	00.0%	60.0%	60.0%	100%
Semi vehicle	7	14.3%	57.1%	100%	100%
Train car	5	40.0%	100%	100%	100%

Table 2
Perfect alignment percentages for some classes (311 models). Comparing the Accuracy of CPCA and our method with the shape descriptors G, E and F.

To evaluate the effectiveness of our alignment algorithm in shape retrieval tasks, we applied it as a normalization step in a general retrieval process. As 3D retrieval approaches based on 2D projections (2D/3D approaches) are very sensitive to the 3D model orientation, we have chosen to test our alignment on one of these meth-

ods. We used the shape descriptor DLA [3] that represents each model by a set of depth lines transformed into sequences and the dynamic programming distance DPD that measures the similarity between the depth line descriptors. To compare objectively the retrieval effectiveness, for both types of alignment methods, we computed Precision-Recall diagrams commonly used in information search (the query is not counted in the answer as in [23]) and four quantitative measures, Nearest Neighbor (NN), First Tier (FT), Second Tier (ST), and Discounted Cumulative Gain (DCG), for evaluating query results (see [19] for a description of this measures).

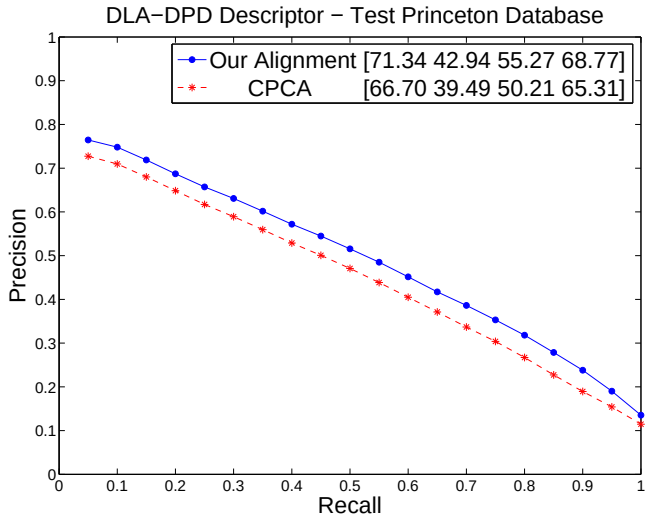


Fig. 9. Average Precision-recall curves using the CPCA and our alignment followed by the depth line-based approach DLA (with dynamic programming distance DPD, 6 depth images of size 32x32). The mean NN, FT, ST and DCG values are given in the legends.

Comparing the curves as well as the NN, FT, ST and DCG values in Figure 9, we conclude that our alignment method clearly outperforms the CPCA. In particular, these results confirm that our approach is better than CPCA for aligning similar models in the same way.

Efficiency:

The $\mathcal{O}(N_T)$ complexity of the CPCA algorithm makes our approach clearly faster than the existing alignment approaches based on symmetry in 3D rotation space. As one can see in Table 3, the CPCA provides, in our general algorithm applied to the Test PSB database, a quick alignment for 28.5% of the models $(N_{GA} = 2^+)$ that have at least two good alignment axes. The most time-consuming stage is the symmetry descriptor algorithm for finding the plane with maximal symmetry. This descriptor was computed on 20% of the models $(N_{GA} = 0)$ that don't have any good alignment axis within CPCA-axes.

N_{GA}	0	1	2^+
Number of 3D models	181	467	259
Distribution (N_{GA})	20.0%	51.5%	28.5%

Table 3
Repartition of 3D models of the Test Princeton Shape Benchmark database with respect to N_{GA}

6 Ordering and orientation of the axes of the alignment frame

As we noticed before, the three alignment axes $\mathbf{n_1}, \mathbf{n_2}, \mathbf{n_3}$ of R_{ga} computed by the algorithm of section 4 are given in an arbitrary orientation and order. Thus, 48 coordinate systems can be built by performing permutations and inversions of the alignment axes.
In our tests, we used the method based on moments, which is described in section 6.1, to compute a coordinate system. This method is guaranteed to give the same order and orientations of the alignment axes for similar models in most of the cases. However, the solution is generally different from the natural pose that a human would select.
In section 6.2 we introduce the perspective of this work: an alternative method where the symmetry properties of the model are used to position the model according to the human perception. A unique coordinate system cannot be computed in all the cases but a set of coordinate system candidates can be proposed.

6.1 Method based on moments

In this section, we follow Vranic's approach [23] used to order the CPCA axes and to fi x their orientation. Let $\mathfrak{S}_{ga}$ be the surface of the model in the frame R_{ga}. We calculate the average projections of the points of $\mathfrak{S}_{ga}$ in the new coordinate planes through the center of gravity of the model.

$$c_x = \frac{1}{\mathscr{A}} \iint_{\mathbf{p}\in\mathfrak{S}_{ga}} p_x.p_x \, ds, \;\; c_y = \frac{1}{\mathscr{A}} \iint_{\mathbf{p}\in\mathfrak{S}_{ga}} p_y.p_y \, ds, \;\; c_z = \frac{1}{\mathscr{A}} \iint_{\mathbf{p}\in\mathfrak{S}_{ga}} p_z.p_z \, ds,$$

with $\mathbf{p} = (p_x, p_y, p_z)$.

We sort the values of c_x, c_y and c_z in decreasing order and form the rotation matrix $\mathbf{A}$ which has the three corresponding unit vectors as rows. We rotate the vertices of $\mathfrak{S}_{ga}$ using $\mathbf{A}$ and obtain a new surface $\mathfrak{S}'_{ga}$.

To ensure the reflection invariance, vertices of $\mathfrak{S}'_{ga}$ are multiplied by a diagonal

matrix $\mathbf{F} = diag(sign(f_x), sign(f_y), sign(f_z))$, with

$$f_x = \frac{1}{\mathscr{A}} \iint_{\mathbf{p} \in \mathfrak{S}'_{ga}} sign(p_x)|p_x|^2 ds\ ,\ \mathbf{p} = (p_x, p_y, p_z) \quad \text{and } f_y, f_z \text{ similar.}$$

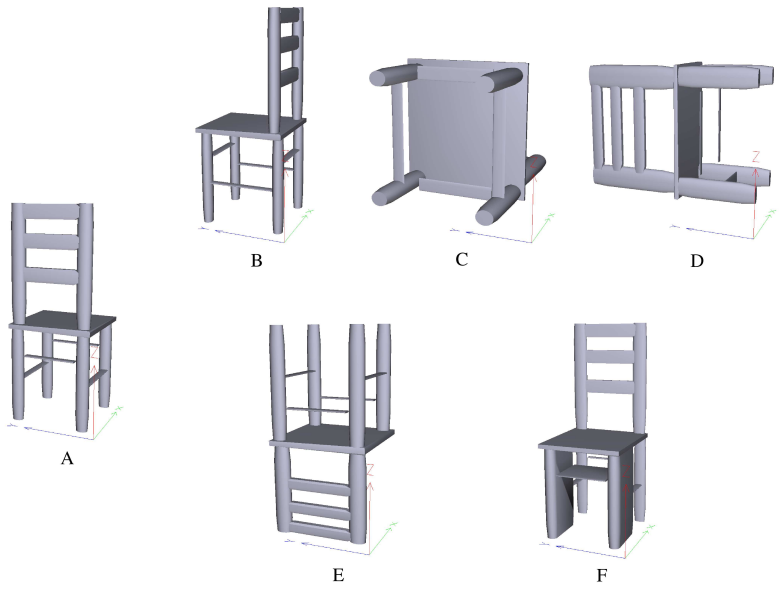


Fig. 10. The different coordinate frames obtained with the method based on moments after deformations of a chair.

This method is not robust w.r.t. small deformations of a model. Let us consider the chairs of Figure 10. The chair A has been transformed in chairs B, C, D, E and F. The chair B has been scaled in depth by a factor 1.1, the chair D has been scaled in width by a factor 1.9 and the chair C has been scaled in width and depth by a factor 2. The method based on moments computes four different reference frames for these four chairs. Moreover, by changing the proportions of the back of the chair, as in case E and by enlarging the front legs of the chair as in case F, the orientations of the axes are different from the original ones.

6.2 Alternative method

In this section, we discuss the positions of the alignment axes with respect to the natural pose (a vertical direction and two horizontal ones that a human perceives). To do this, we use the value of N_{GA} and the type of symmetry group of a 3D model detailed in section 3.1. Note that the order and the orientations cannot be found in all the cases but this approach will reduce human interaction. Our work will be guided by the following observation: when considering human perception, Fergu-

son [9] observed that the best known orientation effect is the preference for vertical symmetry. It is confirmed when examining the 3D models of the Princeton Shape Benchmark: for example, the models of Figure 6 having a planar symmetry have a mirror plane in vertical position.
Thus two assumptions are used when computing the reference frames from the alignment axes:
- When the 3D model has a cyclic or a dihedral symmetry, the axis of cyclic or dihedral symmetry is vertical.
- When the object has a reflection plane and when its associated normal **n** is not the axis of a dihedral symmetry, **n** is horizontal.
In the following, we need a process that detect and compute the order of a cyclic symmetry along a given axis $\mathbf{n_i}$ for a 3D model. This can be carried out by using the symmetry descriptor of Podolak et al. [17] on a distribution of planes rotated around $\mathbf{n_i}$.

Considering the number of principal axes retained by our method, we now discuss the ordering and orientation of the alignment directions. Let $\mathscr{M}$ be a 3D model with N_{GA} good alignment axes detected by the CPCA and $R_{ga} = (\mathbf{n_1};\mathbf{n_2};\mathbf{n_3})$ its reference frame computed by the algorithm of section 4. We want to compute the reference frames $R'_{ga} = (\mathbf{w_1};\mathbf{w_2};\mathbf{w_3})$, where $\mathbf{w_3}$ is in the vertical direction and $\mathbf{w_1} = \pm\mathbf{n_i}$, $\mathbf{w_2} = \pm\mathbf{n_j}$ and $\mathbf{w_3} = \pm\mathbf{n_k}$,with $\{i,j,k\} = \{1,2,3\}$. According to sections 3.1 and 3.2, we have:

- If $N_{GA} = 3$, then $\mathscr{M} \in \mathbf{G_D^{even}} \bigcup \mathbf{G_O}$
  As we don't know whether $\mathscr{M}$ is in $\mathbf{G_D^{even}}$ or in $\mathbf{G_O}$, we study first the case $\mathscr{M} \in \mathbf{G_D^{even}}$ and explain how to proceed in all the cases.
  When $\mathscr{M} \in \mathbf{G_D^{even}}$, $\mathscr{M}$ has an even number n of mirror planes and has another mirror plane P_3 which is perpendicular to the previous ones. In this case, $\mathscr{M}$ has a dihedral symmetry around the intersection of the n mirror planes. When $n > 2$, this axis has to be the vertical axis w_3 of our reference frame. To differentiate the three cases ($\mathbf{G_D^{even}}$ with $n = 2$, $\mathbf{G_D^{even}}$ with $n > 2$ and $\mathbf{G_O}$), we test the cyclic symmetry around each principal axis:
  - If the order of the cyclic symmetry around each principal axis is 2, then $\mathscr{M} \in \mathbf{G_D^{even}}$ with $n = 2$ and six reference frames can be computed from the principal axes.
  - If the order of the cyclic symmetries around two principal axes is 2 and if the order n of cyclic symmetry around the third axis is greater than 2, then $\mathscr{M} \in \mathbf{G_D^{even}}$ with $n > 2$. Moreover, if n is a multiple of 4, then the two horizontal axes play the same role and we have one coordinate system (case (a) of Figure 11), otherwise we obtain two different coordinate systems.
  - If the order of the cyclic symmetry around each principal axis is 4, then $\mathscr{M} \in \mathbf{G_O}$ and any ordering and orientation of the three axes computed by the CPCA will lead to the same coordinate system for $\mathscr{M}$.
  This is summarized in table 4.

N_{GA}	Order of the cyclic symmetry around			Class	4 divides n	Vertical axis	Coordinate systems proposed to the user
	$\mathbf{n_1}$	$\mathbf{n_2}$	$\mathbf{n_3}$				
3	2	2	2	$\mathbf{G_D^{even}}$			$(\mathbf{n_1};\mathbf{n_2};\mathbf{n_3})$ $(\mathbf{n_2};\mathbf{n_1};\mathbf{n_3})$ $(\mathbf{n_1};\mathbf{n_3};\mathbf{n_2})$ $(\mathbf{n_3};\mathbf{n_1};\mathbf{n_2})$ $(\mathbf{n_2};\mathbf{n_3};\mathbf{n_1})$ $(\mathbf{n_3};\mathbf{n_2};\mathbf{n_1})$
3	2	2	$n>2$	$\mathbf{G_D^{even}}$	no	$\mathbf{n_3}$	$(\mathbf{n_1};\mathbf{n_2};\mathbf{n_3})$ $(\mathbf{n_2};\mathbf{n_1};\mathbf{n_3})$
3	2	2	$n>2$	$\mathbf{G_D^{even}}$	yes	$\mathbf{n_3}$	$(\mathbf{n_1};\mathbf{n_2};\mathbf{n_3})$
3	$n>2$	2	2	$\mathbf{G_D^{even}}$	no	$\mathbf{n_1}$	$(\mathbf{n_2};\mathbf{n_3};\mathbf{n_1})$ $(\mathbf{n_3};\mathbf{n_2};\mathbf{n_1})$
3	$n>2$	2	2	$\mathbf{G_D^{even}}$	yes	$\mathbf{n_1}$	$(\mathbf{n_2};\mathbf{n_3};\mathbf{n_1})$
3	2	$n>2$	2	$\mathbf{G_D^{even}}$	no	$\mathbf{n_2}$	$(\mathbf{n_3};\mathbf{n_1};\mathbf{n_2})$ $(\mathbf{n_1};\mathbf{n_3};\mathbf{n_2})$
3	2	$n>2$	2	$\mathbf{G_D^{even}}$	yes	$\mathbf{n_2}$	$(\mathbf{n_3};\mathbf{n_1};\mathbf{n_2})$
3	4	4	4	$\mathbf{G_0}$		$\mathbf{n_3}$	$(\mathbf{n_1};\mathbf{n_2};\mathbf{n_3})$
2	$n>2$	-	2	$\mathbf{G_D^{odd}}$	no	$\mathbf{n_1}$	$(\mathbf{n_2};\mathbf{n_3};\mathbf{n_1})$ $(\mathbf{n_2};-\mathbf{n_3};\mathbf{n_1})$ $(\mathbf{n_3};\mathbf{n_2};\mathbf{n_1})$ $(-\mathbf{n_3};\mathbf{n_2};\mathbf{n_1})$
2	-	$n>2$	2	$\mathbf{G_D^{odd}}$	no	$\mathbf{n_2}$	$(\mathbf{n_1};\mathbf{n_3};\mathbf{n_2})$ $(\mathbf{n_1};-\mathbf{n_3};\mathbf{n_2})$ $(\mathbf{n_3};\mathbf{n_1};\mathbf{n_2})$ $(-\mathbf{n_3};\mathbf{n_1};\mathbf{n_2})$
2	-	-	$n\geq 2$	$\mathbf{G_C^{even}}$	no	$\mathbf{n_3}$	$(\mathbf{n_1};\mathbf{n_2};\mathbf{n_3})$ $(\mathbf{n_1};\mathbf{n_2};-\mathbf{n_3})$ $(\mathbf{n_2};\mathbf{n_1};\mathbf{n_3})$ $(\mathbf{n_2};\mathbf{n_1};-\mathbf{n_3})$
2	-	-	$n\geq 2$	$\mathbf{G_C^{even}}$	yes	$\mathbf{n_3}$	$(\mathbf{n_1};\mathbf{n_2};\mathbf{n_3})$ $(\mathbf{n_1};\mathbf{n_2};-\mathbf{n_3})$

Table 4
Coordinate system candidates when $N_{GA} > 1$.

N_{GA}	Class	Horizontal axis	Vertical axis	Coordinate systems proposed to the user
1	$\mathbf{G_U}$	$\mathbf{n_1}$	$\mathbf{n_2}$ or $\mathbf{n_3}$	$(\mathbf{n_1};\pm\mathbf{n_2};\pm\mathbf{n_3})$ $(\mathbf{n_1};\pm\mathbf{n_3};\pm\mathbf{n_2})$ $(\pm\mathbf{n_3};\mathbf{n_1};\pm\mathbf{n_2})$ $(\pm\mathbf{n_2};\mathbf{n_1};\pm\mathbf{n_3})$
0	$\mathbf{G_Z}$	$\mathbf{n_1}$	$\mathbf{n_2}$ or $\mathbf{n_3}$	$(\pm\mathbf{n_1};\pm\mathbf{n_2};\pm\mathbf{n_3})$ $(\pm\mathbf{n_1};\pm\mathbf{n_3};\pm\mathbf{n_2})$ $(\pm\mathbf{n_3};\pm\mathbf{n_1};\pm\mathbf{n_2})$ $(\pm\mathbf{n_2};\pm\mathbf{n_1};\pm\mathbf{n_3})$

Table 5
Coordinate system candidates when $N_{GA} \leq 1$.

- If $N_{GA} = 2$, then $\mathscr{M} \in \mathbf{G_C^{even}} \bigcup \mathbf{G_D^{odd}}$.
  Following the remark of section 4, the two mirror planes detected by the CPCA are given by $\mathbf{n_1}$ and $\mathbf{n_2}$. We test the cyclic symmetry around these axes. If one of these axes has a cyclic symmetry, then $\mathscr{M} \in \mathbf{G_D^{odd}}$ and the axis is vertical (case (b) of Figure 11) and four coordinate systems are computed. In the other case, $\mathscr{M} \in \mathbf{G_C^{even}}$ and $\mathbf{n_3}$, the axis of cyclic symmetry of order n, is vertical. Moreover, if n is a multiple of 4, then $\mathbf{n_1}$ and $\mathbf{n_2}$ play the same role and we obtain two different coordinate systems, as in the case (c) of Figure 11. Otherwise, we obtain four coordinate systems, as resumed in table 4 and illustrated in case (d) of Figure 11.
- If $N_{GA} = 1$, then $\mathscr{M} \in \mathbf{G_C^{odd}} \bigcup \mathbf{G_T} \bigcup \mathbf{G_I} \bigcup \mathbf{G_U}$
  The eigenvector $\mathbf{n_1}$ computed by the CPCA will be horizontal and the user has to choose among 16 coordinate systems given in table 5.
- If $N_{GA} = 0$, then $\mathscr{M} \in \mathbf{G_Z}$.
  The normal $\mathbf{n_1}$ of the plane with maximal reflection symmetry computed by the algorithm of section 4 will be horizontal and 32 coordinate systems are proposed to the user, resumed in table 5.

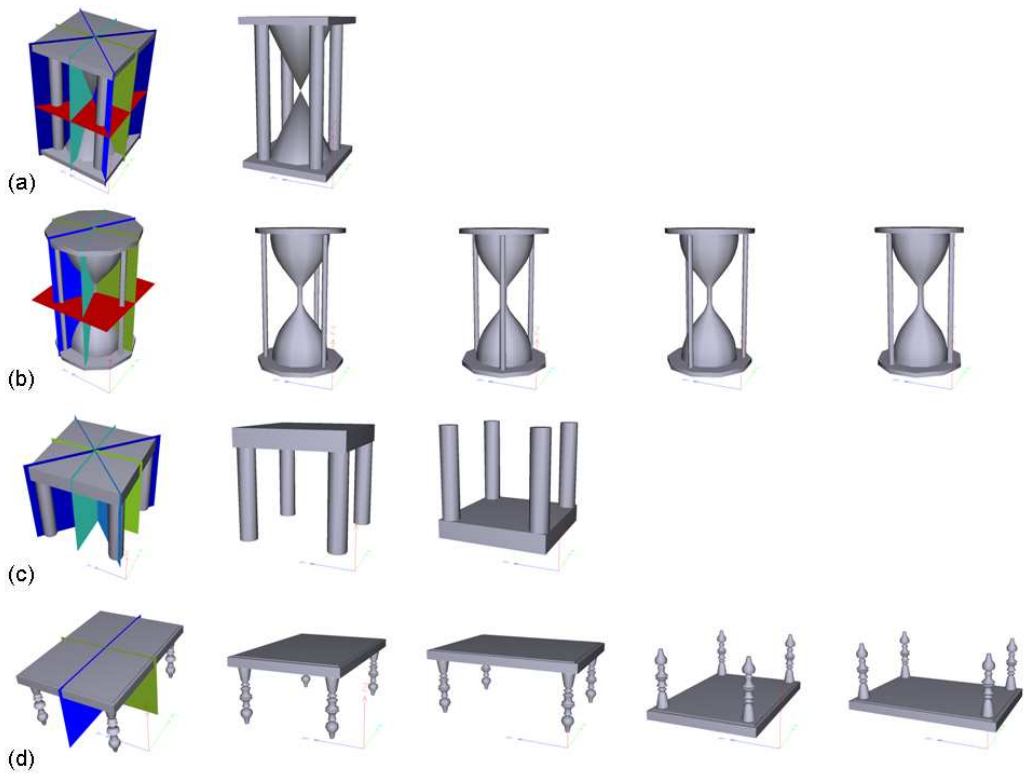


Fig. 11. The coordinate systems corresponding to the following cases: (a) $N_{GA} = 3$, $\mathscr{M} \in \mathbf{G}_{\mathbf{D}}^{\mathbf{even}}$ with $n = 4$, (b) $N_{GA} = 2$, $\mathscr{M} \in \mathbf{G}_{\mathbf{C}}^{\mathbf{odd}}$, with $n = 3$, (c) $N_{GA} = 2$, $\mathscr{M} \in \mathbf{G}_{\mathbf{C}}^{\mathbf{even}}$ with $n = 4$, (d) $N_{GA} = 2$, $\mathscr{M} \in \mathbf{G}_{\mathbf{C}}^{\mathbf{even}}$ with $n = 2$.

We are currently developing an interactive tool that proposes to a user a set of coordinate system candidates based on this discussion. It can also be used to reduce the number of cases to consider when computing the similarity between two 3D models in shape retrieval methods.

Reference frame and upright orientation

Fu et al. [10] deal with a close but slightly different problem: given a training set of models with their prescribed upright orientations, they propose a method computing the upright orientation of new objects but they don't compute the object's reference frames. They focus on "standing man-made models", that is models which usually stand on flat surfaces and which "have well-defined upright orientations". For that purpose, a supervised learning algorithm selects the best orientation from a small set of orientation candidates extracted by analyzing the object's convex hull. A set of attributes are associated to each candidate base. Their values are computed evaluating geometric properties such as static stability, symmetry, parallelism and visibility of the 3D objects w.r.t. the base.
Our alignment method could replace the convex hull process to find candidate upright orientations: it would lead to at most 6 possible upright orientations associated to the alignment axes. Moreover, following the process summarized in table 4, we

would compute upright orientations of models belonging to $\mathbf{G_D^{even}}$ (except when there are exactly three mirror planes, which corresponds to the first line of table 4), $\mathbf{G_D^{odd}}$ and $\mathbf{G_O}$, cases where the coordinate systems proposed to the user gives only one vertical axis with the same orientation.
Conversely, if we use a property such as static stability when looking for a reference frame, we can reduce the number of coordinate system candidates proposed to the user but we may eliminate the user's solution as for the case of the gun and of the street light in Figure 8. Parallelism and visibility properties may also be fragile if we consider that the user's choice is primordial.
Anyway, the symmetry properties seem to be the most important ones for both upright orientations and reference frames: in fact, the examples of correct upright orientations given in Figure 6 of [10] have, for most of them, one or more vertical reflective planes.

7 Conclusion

We have presented a new alignment method for 3D models. It retains the principal axes of the CPCA with respect to the approximate reflection plane symmetry. We have introduced a new notion of cost (*LTIC*) that measures the invariance of a model with respect to local translation along a given direction. This measure is used to compute the remaining alignment axes. To obtain the ordering and orientation of the resulting alignment axes, we have proposed a method that reduces the number of coordinate systems to a set of candidates containing the optimal solution.
Our experiments show that our approach consistently aligns the 3D objects: we obtain 100% in 24 classes among the 33 classes tested and the others never exceed less than 75% of correct alignment. Moreover, our alignment method provides more accurate results than the CPCA when it is used as a normalization step in a 3D shape retrieval method.

References

[1] N. Aspert, D. Santa-Cruz, T. Ebrahimi, MESH: Measuring errors between surfaces using the Hausdorff distance, in: IEEE International Conference on Multimedia and Expo (ICME 02), Lausanne, Switzerland, 2002, pp. 705–708.

[2] B. Bustos, D. A. Keim, T. Schreck, D. Vranic, An experimental comparison of feature-based 3D retrieval methods, in: 2nd International Symposium on 3D Data Processing, Visualization and Transmission, (3DPVT'04), Thessaloniki, Greece, 2004, pp. 215–222.

[3] M. Chaouch, A. Verroust-Blondet, 3D model retrieval based on depth line descriptor, in: IEEE International Conference on Multimedia & Expo (ICME'07), Beijing, China,

2007, pp. 599–602.

[4] M. Chaouch, A. Verroust-Blondet, A novel method for alignment of 3D models, in: IEEE International Conference on Shape Modeling and Applications, (SMI 2008), 2008, pp. 187–195.

[5] D. Chen, M. Ouhyoung, A 3D model alignment and retrieval system, in: International Computer Symposium, Workshop on Multimedia Technologies, vol. 2, Hualien, Taiwan, 2002, pp. 1436–1443.

[6] D. Chen, X. Tian, Y. Shen, M. Ouhyoung, On visual similarity based 3D model retrieval, Computer Graphics Forum 22 (3) (2003) 223–232.

[7] P. Cignoni, C. Rocchini, R. Scopigno, Metro: Measuring error on simplified surfaces, Computer Graphics Forum 17 (2) (1998) 167–174.

[8] B. A. Dubrovin, A. T. Fomenko, S. P. Novikov, Modern geometry, methods and applications. Part I. , The Geometry od surfaces, transformation groups, and fields, Springer-Verlag, 1992.

[9] R. W. Ferguson, Modeling orientation effects in symmetry detection: The role of visual structure, in: 22nd Conference of the Cognitive Science Society, New Jersey, 2000, pp. 125–130.

[10] H. Fu, D. Cohen-Or, G. Dror, A. Sheffer, Upright orientation of man-made objects, ACM Trans. on Graphics 27 (3).

[11] M. Kazhdan, An approximate and efficient method for optimal rotation alignment of 3D models, IEEE Trans. Pattern Anal. Mach. Intell. 29 (7) (2007) 1221–1229.

[12] M. Kazhdan, T. Funkhouser, S. Rusinkiewicz, Symmetry descriptors and 3D shape matching, in: Symposium on Geometry Processing, 2004, pp. 117–126.

[13] A. Martinet, C. Soler, N. Holzschuch, F. X. Sillion, Accurate detection of symmetries in 3D shapes, ACM Trans. on Graphics 25 (2) (2006) 439–464.

[14] P. Minovic, S. Ishikawa, K. Kato, Symmetry identification of a 3-D object represented by octree, IEEE Trans. Pattern Anal. Mach. Intell. 15 (5) (1993) 507–514.

[15] N. J. Mitra, L. Guibas, M. Pauly, Partial and approximate symmetry detection for 3D geometry, ACM Trans. on Graphics 25 (3) (2006) 560–568.

[16] E. Paquet, M. Rioux, A. Murching, T. Naveen, A. Tabatabai, Description of shape information for 2-D and 3-D objects, Signal Processing : Image Communication 16 (2000) 103–122.

[17] J. Podolak, P. Shilane, A. Golovinskiy, S. Rusinkiewicz, T. Funkhouser, A planar-reflective symmetry transform for 3D shapes, ACM Trans. on Graphics 25 (3) (2006) 549–559.

[18] R. Rustamov, Augmented symmetry transforms, in: IEEE International Conference on Shape Modeling and Applications (SMI'07), Lyon, France, 2007, pp. 13–20.

[19] P. Shilane, P. Min, M. Kazhdan, T. Funkhouser, The Princeton shape benchmark, in: IEEE International Conference on Shape Modeling and Applications (SMI'04), Genova, Italy, 2004, pp. 167–178.

[20] P. Simari, E. Kalogerakis, K. Singh, Folding meshes: Hierarchical mesh segmentation based on planar symmetry, in: Fourth Eurographics Symposium on Geometry Processing, 2006, pp. 111–120.

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

[22] J. Tangelder, R. Veltkamp, A survey of content based 3D shape retrieval methods, Multimedia Tools and Applications 39 (3) (2008) 441–471.

[23] D. Vranic, 3D model retrieval, Ph.D. thesis, U. of Leipzig (2004).

[24] D. Vranic, D. Saupe, J. Richter, Tools for 3D-object retrieval: Karhunen-Loeve transform and spherical harmonics, in: 2001 Workshop Multimedia Signal Processing, Cannes, France, 2001, pp. 293–298.

[25] H. Zabrodsky, S. Peleg, D. Avnir, Symmetry as a continuous feature, IEEE Trans. Pattern Anal. Mach. Intell. 17 (12) (1995) 1154–1166.

[26] T. Zaharia, F. Prêteux, 3D versus 2D/3D shape descriptors: A comparative study, in: SPIE Conf. on Image Processing: Algorithms and Systems III - IS&T / SPIE Symposium on Electronic Imaging, Science and Technology '04, vol. 5298, San Jose, CA, USA, 2004, pp. 47–58.

Appendix

Lemma 3. *Let π be a mirror plane of $\mathfrak{S}$ and* **g** *be the center of gravity of $\mathfrak{S}$. Then* $\mathbf{g} \in \pi$.

Lemma 4. *Let π be a mirror plane of $\mathfrak{S}$ and* **n** *be the unit normal of π. Then* **n** *is an eigenvector of $\mathfrak{S}$.*

Proof. The vector **n** is an eigenvector of the covariance matrix C of $\mathfrak{S}$, if $\exists\ \lambda \neq 0$ such that $C \cdot \mathbf{n} = \lambda\ \mathbf{n}$.
Let $\pi = \{\mathbf{u} \in \mathbb{R}^3 | \mathbf{n}^T \cdot \mathbf{u} = \delta\}$ be the mirror plane of $\mathfrak{S}$. Then, $\forall\, \mathbf{v} \in \mathfrak{S}$, $\exists\, (\mathbf{v}', \mathbf{v}_\pi, d_v) \in (\mathfrak{S}, \pi, \mathbb{R})$ such that $\mathbf{v} = \mathbf{v}_\pi + d_v\ \mathbf{n}$, $\mathbf{v}' = \mathbf{v}_\pi - d_v\ \mathbf{n}$.
Suppose **g** is the the center of gravity of $\mathfrak{S}$. We fi rst construct the covariance matrix of $\mathfrak{S}$.

$$
\begin{aligned}
C &= \frac{1}{S}\iint_{\mathbf{v}\in\mathfrak{S}} (\mathbf{v}-\mathbf{g})\cdot(\mathbf{v}-\mathbf{g})^T \, ds \\
&= \frac{1}{2S}\iint_{\mathbf{v}\in\mathfrak{S}} (\mathbf{v}-\mathbf{g})\cdot(\mathbf{v}-\mathbf{g})^T \, ds \\
&\quad + \frac{1}{2S}\iint_{\mathbf{v}'\in\mathfrak{S}} (\mathbf{v}'-\mathbf{g})\cdot(\mathbf{v}'-\mathbf{g})^T \, ds \\
&= \frac{1}{2S}\iint_{\mathbf{v}\in\mathfrak{S}} (\mathbf{v}_\pi-\mathbf{g}+d_v\,\mathbf{n})\cdot(\mathbf{v}_\pi-\mathbf{g}+d_v\,\mathbf{n})^T \, ds \\
&\quad + \frac{1}{2S}\iint_{\mathbf{v}\in\mathfrak{S}} (\mathbf{v}_\pi-\mathbf{g}-d_v\,\mathbf{n})\cdot(\mathbf{v}_\pi-\mathbf{g}-d_v\mathbf{n})^T \, ds \\
&= \frac{1}{S}\iint_{\mathbf{v}\in\mathfrak{S}} (\mathbf{v}_\pi-\mathbf{g})\cdot(\mathbf{v}_\pi-\mathbf{g})^T \, ds + \frac{1}{S}\iint_{\mathbf{v}\in\mathfrak{S}} d_v^2\,\mathbf{n}\cdot\mathbf{n}^T \, ds \\
C\cdot\mathbf{n} &= \frac{1}{S}\left[\iint_{\mathbf{v}\in\mathfrak{S}} (\mathbf{v}_\pi-\mathbf{g})\cdot(\mathbf{v}_\pi-\mathbf{g})^T \, ds\right]\cdot\mathbf{n} + \frac{1}{S}\left[\iint_{\mathbf{v}\in\mathfrak{S}} d_v^2\,\mathbf{n}\cdot\mathbf{n}^T \, ds\right]\cdot\mathbf{n} \\
&= \frac{1}{S}\iint_{\mathbf{v}\in\mathfrak{S}} (\mathbf{v}_\pi-\mathbf{g})\cdot(\mathbf{v}_\pi-\mathbf{g})^T\cdot\mathbf{n}\, ds + \frac{1}{S}\iint_{\mathbf{v}\in\mathfrak{S}} d_v^2\,\mathbf{n}\cdot\mathbf{n}^T\cdot\mathbf{n}\, ds
\end{aligned}
$$

By previous Lemma $\mathbf{g}\in\pi$ and by orthogonal projection for all $\mathbf{v}$ onto π i.e., $\mathbf{v}_\pi\in\pi$, we get

$$
\begin{aligned}
(\mathbf{v}_\pi-\mathbf{g})^T\cdot\mathbf{n} &= \mathbf{n}^T\cdot(\mathbf{v}_\pi-\mathbf{g}) \\
&= \mathbf{n}^T\cdot\mathbf{v}_\pi - \mathbf{n}^T\cdot\mathbf{g} \\
&= \delta-\delta \;=\; 0
\end{aligned}
$$

here $\mathbf{n}$ is unit vector, thus we have $\mathbf{n}^T\cdot\mathbf{n}=1$
Combining the three equations, we obtain

$$
C\cdot\mathbf{n} = \left[\frac{1}{S}\iint_{\mathbf{v}\in\mathfrak{S}} d_v^2\, ds\right]\mathbf{n} \;=\; \lambda\,\mathbf{n}
$$

Therefore, the unit normal $\mathbf{n}$ of the mirror plane π is the eigenvector of $\mathfrak{S}$ and $\frac{1}{S}\iint_{\mathbf{v}} d_v^2$ is the corresponding eigenvalue. □