



# A Novel Method for Alignment of 3D Models

Mohamed Chaouch, Anne Verroust-Blondet

## ► To cite this version:

Mohamed Chaouch, Anne Verroust-Blondet. A Novel Method for Alignment of 3D Models. [Research Report] RR-6408, 2008, pp.22. inria-00203336v3

**HAL Id: inria-00203336**

**<https://inria.hal.science/inria-00203336v3>**

Submitted on 11 Jan 2008 (v3), last revised 11 Jan 2008 (v4)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# *A Novel Method for Alignment of 3D Models*

Mohamed Chaouch — Anne Verroust-Blondet

N° ????

Janvier 2008

Thème COG

 *apport  
de recherche*



## A Novel Method for Alignment of 3D Models

Mohamed Chaouch<sup>\*</sup> , Anne Verroust-Blondet<sup>†</sup>

Thème COG — Systèmes cognitifs  
Équipes-Projets Imedia

Rapport de recherche n° 7777 — Janvier 2008 — 22 pages

**Abstract:** In this paper we present a new method for alignment of 3D models. This approach is based on symmetry properties, and uses the fact that PCA techniques have good properties with respect to the planar reflective symmetry. The fast search of the best optimal alignment axes within the PCA-eigenvectors is an essential first step in our alignment process. The plane reflection symmetry is used as a criterion for selection. This pre-processing transforms the alignment problem into an indexing scheme based on the number of the retained PCA-axes. We also introduce a local translational invariance cost (*LTIC*) that captures a measure of the local translational symmetries of a shape with respect to a given direction. Experimental results show that the proposed method finds the rotation that best aligns a 3D mesh.

**Key-words:** polyhedral models, 3D models, alignment, normalization, plane reflection symmetry, principal components analysis

This work has been supported in part by the DELOS NoE on Digital Libraries (EU IST NoE G038-507618).

<sup>\*</sup> Email: mohamed.chaouch@inria.fr

<sup>†</sup> Email: anne.verroust@inria.fr

## Une nouvelle méthode d'alignement de modèles tridimensionnels

**Résumé :** Une nouvelle méthode d'alignement de modèles 3D est proposée. Cette approche est basée sur des propriétés de symétrie et utilise le fait que les techniques d'analyse en composantes principales (ACP) ont de bonnes propriétés vis à vis des plans de symétrie d'un objet tridimensionnel. La recherche rapide d'axes d'alignement optimaux parmi les vecteurs propres de l'ACP est une première étape du processus d'alignement. Les plans de symétrie du modèle 3D sont utilisés comme critère de sélection. Quand l'ACP ne fournit pas tous les axes, nous introduisons un coût d'invariance local de translation (*LTIC*) qui calcule une mesure de symétrie locale de translation d'une forme le long d'un axe. Les résultats expérimentaux sur la Base de données de Princeton montrent que la méthode proposée calcule des alignements cohérents à l'intérieur de chaque classe de modèles 3D.

**Mots-clés :** modèles 3D, modèles polyédriques, alignement, normalisation, symétrie de réflexion plane, analyse en composantes principales

# 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 [SMKF04, ZP04, BKSV04]. 3D models are generally given in arbitrary scale, position and orientation in 3D-space and most of the methods do not satisfy geometrical invariance, then it is very important to normalize the models into a canonical coordinate frame before any processing. The normalization consists in two steps: the alignment to determine the pose-invariant and the scaling to make the scale-invariant. The alignment has always been 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 good rotation is still a well discussed topic [PRM<sup>+</sup>00, VSR01, CO02, SMKF04, ZP04, BKSV04, Kaz07]. The alignment problem addressed in this paper is different from the alignment approaches of [CO02, Kaz07], 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.

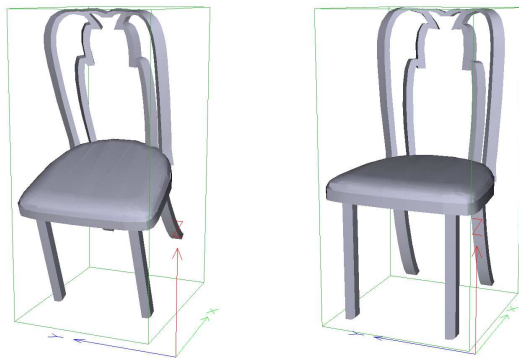


Figure 1: A model of chair. Left: before normalization. Right: well aligned model

When looking at a 3D model, we can say whether it is well aligned or not and we know, in most of the cases, how to find its good alignment (cf. Figure 1). When the 3D model has symmetries, the object is aligned with particular axes or symmetry planes. This is confirmed by Ferguson [Fer00] who noticed that symmetry detection is a key part of human perception and this fact has guided Podolak et al. [PSG<sup>+</sup>06] 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) 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. Thus 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 proceed as follows. 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 (*LTIC*) cost that measures the invariance of a model with respect to local translation about 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 structured as follows. Section 2 reviews related work on alignment and symmetry detection for 3D models. Section 3 presents our selection of the best alignment axes within the PCA-eigenvectors by analyzing the plane reflection symmetry, and section 4 describes our alignment method. Experimental results of are presented in section 5. Finally, we conclude and point out future work in section 6.

## 2 Related Work

The most well-known approach computing the alignment of 3D objects is the principle component analysis method (PCA) [PRM<sup>+</sup>00, VSR01, SMK<sup>+</sup>04, ZP04, BKS<sup>+</sup>04] 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. The 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.

Podolak et al. [PSG<sup>+</sup>06] 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 [Rus07].

Other methods finding symmetries in 3D models have been presented. Minovic et al. [MIK93] 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. Sun and Sherrah [SS97] 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. While Martinet et al. [MSHS06] use generalized moments to detect perfect symmetries in 3D shapes, Mitra et al. [MGP06] 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 a same semantic class) have similar alignments. As noticed in [MIK93], 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 the following  $\mathcal{M}$  will denote a 3D polygonal model represented by its surface  $\mathfrak{S}$  composed of a set of triangular facets  $\mathfrak{T} = \{T_1, \dots, T_{N_T}\}$ ,  $T_i \subset \mathbb{R}^3$ , given by a set of vertices  $\mathfrak{P} = \{\mathbf{p}_1, \dots, \mathbf{p}_{N_P}\}$ ,  $\mathbf{p}_i = (x_i, y_i, z_i) \in \mathbb{R}^3$ .

We study the reflection planes in the symmetry groups [DFN92], and use them to discriminate different classes of mirror symmetry. Then, we discuss for each class when the PCA alignment has good properties with respect to the planar reflective symmetry.

#### 3.1 Plane Reflection Symmetry Analysis

A plane reflection symmetry is defined by a mirror plane  $\pi$  that can be parameterized by its unit normal  $\mathbf{n}$  and its scalar distance  $\delta$  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 [DFN92], 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:

1. **G<sub>C</sub>**: 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 ( $n = 4$ ).
2. **G<sub>D</sub>**: 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.
3. **G<sub>R</sub>**: 3D models that have rotation symmetry such as the five convex regular polyhedra called platonic solids. It contains three sub-groups: **G<sub>T</sub>** of tetrahedral symmetry (6 mirror planes), **G<sub>O</sub>** of octahedral symmetry (9 mirror planes) and **G<sub>I</sub>** of icosahedron symmetry (15 mirror planes).
4. **G<sub>1</sub>**: 3D models that have only one plane reflection symmetry. This is the case for many natural objects such as the airplanes, the animals, the humans, the chairs, the cars, etc.
5. **G<sub>0</sub>**: 3D models that don't have any plane reflection symmetry, such as the plants and the trees.



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



Figure 2: Models from different classes of mirror symmetry: Tetrahedron  $\in G_T$ , Octahedron  $\in G_O$ , Icosahedron  $\in G_I$ , Table  $\in G_C^{even}$ , Streetlight lamp  $\in G_C^{odd}$ , Hourglass  $\in G_D^{even}$ , 5-prism  $\in G_D^{odd}$ , Chair  $\in G_1$ , Plant  $\in G_0$ .

### 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” PCA (CPCA) [VSR01] because it appears to be more complete and the most stable of all the approaches we have studied. CPCA computes three orthogonal eigenvectors of the covariance matrix  $C$ .

As noticed in [MIK93], when  $\pi$  is a mirror plane of  $\mathfrak{S}$  and  $\mathbf{n}$  is the unit normal of  $\pi$ , then  $\pi$  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  $\mathfrak{S}$  (cf. in appendix, the proof in the continuous case). If  $\mathfrak{S}$  has  $n$  mirror planes that pass through a fixed axis (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, 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  $\mathcal{M} \in \mathbf{G}_{\mathbf{C}}$  and  $n$  is even ( $\mathcal{M} \in \mathbf{G}_{\mathbf{C}}^{\text{even}}$ ), 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  $\mathcal{M} \in \mathbf{G}_{\mathbf{C}}$  and  $n$  is odd ( $\mathcal{M} \in \mathbf{G}_{\mathbf{C}}^{\text{odd}}$ ), the CPCA gives only one normal associated to one mirror plane.
- If  $\mathcal{M} \in \mathbf{G}_{\mathbf{D}}$ , 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  $\mathcal{M} \in \mathbf{G}_{\mathbf{O}}$ , then the CPCA detects three orthogonal normals associated to three orthogonal reflection planes, contrarily to the cases of  $\mathbf{G}_{\mathbf{T}}$  and  $\mathbf{G}_{\mathbf{I}}$ , where the CPCA gives only one normal associated to one mirror plane.
- If  $\mathcal{M} \in \mathbf{G}_{\mathbf{I}}$ , then the CPCA gives only one normal associated to its mirror plane.

Thus, when  $\mathcal{M} \in \mathbf{G}_{\mathbf{C}}^{\text{even}} \cup \mathbf{G}_{\mathbf{D}} \cup \mathbf{G}_{\mathbf{O}}$ , the CPCA detects at least two good alignment axes and when  $\mathcal{M} \in \mathbf{G}_{\mathbf{C}}^{\text{odd}} \cup \mathbf{G}_{\mathbf{T}} \cup \mathbf{G}_{\mathbf{I}} \cup \mathbf{G}_{\mathbf{I}}$ , the CPCA gives only one good alignment axis. Finally, when  $\mathcal{M} \in \mathbf{G}_{\mathbf{O}}$ , the CPCA doesn't detect any good alignment axis. We summarize our discussion using  $N_{GA}(\mathcal{M})$ , which accounts the number of the good alignment axis computed by the CPCA. Given the symmetry class of the model  $\mathcal{M}$ ,  $N_{GA}(\mathcal{M})$  is defined as follows:

$$N_{GA}(\mathcal{M}) = \begin{cases} 2^+ & \text{if } \mathcal{M} \in \mathbf{G}_{\mathbf{C}}^{\text{even}} \cup \mathbf{G}_{\mathbf{D}} \cup \mathbf{G}_{\mathbf{O}} \\ 1 & \text{if } \mathcal{M} \in \mathbf{G}_{\mathbf{C}}^{\text{odd}} \cup \mathbf{G}_{\mathbf{T}} \cup \mathbf{G}_{\mathbf{I}} \cup \mathbf{G}_{\mathbf{I}} \\ 0 & \text{if } \mathcal{M} \in \mathbf{G}_{\mathbf{O}} \end{cases}$$

## 4 Alignment of 3D Objects

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

**Algorithm:** Compute good alignment axes

1. Translate the input 3D model  $\mathcal{M}$  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  $N_{GA}(\mathcal{M})$ . This step is detailed in section 4.1.1.
4. Select the good alignment axis/axes according to the value of  $N_{GA}(\mathcal{M})$ :

(if  $N_{GA}(\mathcal{M}) = 2^+$ ) 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  $N_{GA}(\mathcal{M}) = 1$ ) Return  $\mathbf{n} \in \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$  the normal of the unique mirror plane as the first good alignment axis and rotate the 3D model in the new coordinate system  $R'(\mathbf{n}; \mathbf{v}_2; \mathbf{v}_3)$  if  $\mathbf{n} = \mathbf{v}_1$ , in  $R'(\mathbf{n}; \mathbf{v}_3; \mathbf{v}_1)$  if  $\mathbf{n} = \mathbf{v}_2$ , in  $R'(\mathbf{n}; \mathbf{v}_1; \mathbf{v}_2)$  if  $\mathbf{n} = \mathbf{v}_3$ .
- (if  $N_{GA}(\mathcal{M}) = 0$ ) Return  $\mathbf{n}$  the normal of the plane with maximal reflection symmetry (cf. section 4.1.2) as the first good alignment axis and rotate the 3D model in a new coordinate system  $R'(\mathbf{n}; \mathbf{v}'_2; \mathbf{v}'_3)$ .
5. If  $N_{GA}(\mathcal{M}) \in \{0, 1\}$ , 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  $R_{ga}(\mathbf{n}; \mathbf{n}_2; \mathbf{n}_3)$ .

## 4.1 Plane Reflection Symmetry

There have been 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 has the advantage to estimate 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 well 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 us first define  $\mathfrak{S}_\gamma$  as the reflective surface of  $\mathfrak{S}$  with respect to a plane  $\gamma$ . It is represented by a set of triangles  $\mathfrak{T}_\gamma = \{T'_1, \dots, T'_{N_T}\}$ , given by a set of vertices  $\mathfrak{P}_\gamma = \{\mathbf{p}'_1, \dots, \mathbf{p}'_{N_P}\}$ . Following previous works on distance estimation between 3D surfaces [CRS98, ASCE02] and on symmetry distance [ZPA95], we define the continuous symmetry distance  $CSD_\gamma$  of  $\mathfrak{S}$  with respect to a plane reflection  $\gamma$  as:

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

where  $\mathcal{A}$  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:

$$d(\mathbf{p}, \mathfrak{S}_\gamma) = \min_{\mathbf{p}' \in \mathfrak{S}_\gamma} \|\mathbf{p} - \mathbf{p}'\|_2,$$

with  $\|\cdot\|_2$  is the usual Euclidean norm.

The integral of the symmetry error over the whole surface is computed by summing the contributions of all the triangles in  $\mathfrak{T}$ . We obtain a more precise result by taking into account all points of  $\mathfrak{S}$ . The computation of these integrals is only slightly more expensive than the discrete case [ZPA95]. However, it is necessary to sample the surface  $\mathfrak{S}$  uniformly in order to obtain correct point- $\mathfrak{S}_\gamma$  distances: each triangular facet is sampled uniformly. The integral over each triangle  $T_i \subset \mathfrak{T}$  is then approximately done with sums of integrals over triangles

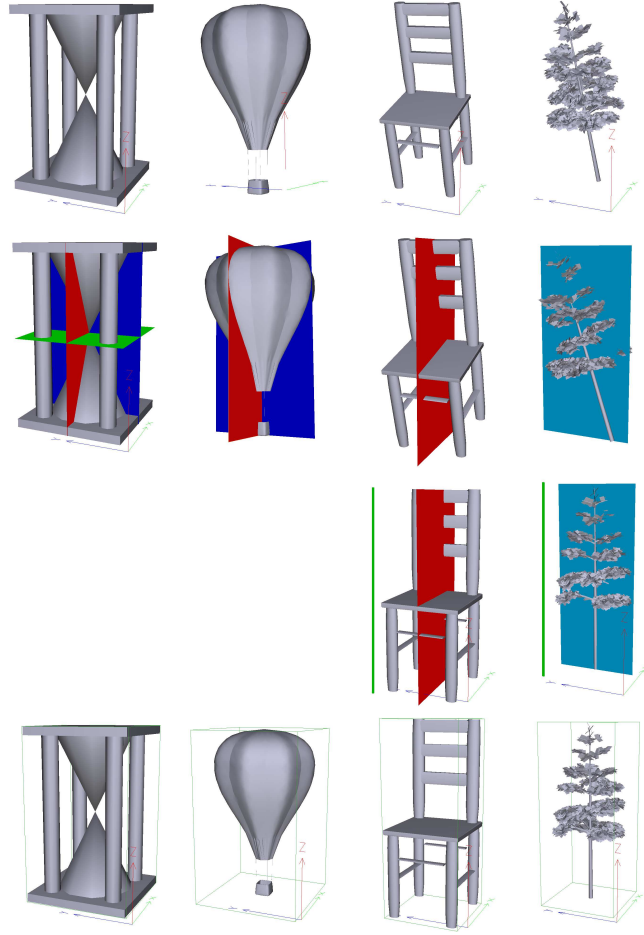


Figure 3: Alignments of models with different  $N_{GA}$  using our method: Hot air balloon and Hourglass models  $N_{GA} = 2^+$ , Chair model  $N_{GA} = 1$ , Tree model  $N_{GA} = 0$ . **Row 1**: CPCA Alignments, **Row 2**: Testing the reflection symmetry for CPCA-coordinate planes, **Row 3**: Computing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

obtained by sampling  $T_i$ .

In fact, for each vertices of each sample triangle  $T \subset T_i$  it would be necessary to calculate the distance to all triangles of  $\mathfrak{S}'$  in order to find the minimum distance. This leads to a complexity  $\mathcal{O}(N_T N_S)$  that is very expensive for large models. This complexity have been reduced in [CRS98] by using a local search processing in order to reduce 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  $\gamma$  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 so that we say that  $\gamma$  is a mirror plane of  $\mathfrak{S}$  when  $SD_\gamma(\mathfrak{S}) < \epsilon$  ( $\epsilon \simeq 0$ ). 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 plane 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. [KFR04] define a symmetry descriptor using the planes through its center of gravity. Podolak et al. [PSG<sup>+</sup>06] extended this work by considering symmetries with respect to all possible planes through a model's bounding volume.

Following [KFR04], we consider a symmetry descriptor that represents the symmetries of a 3D model with respect to only planes through its center of gravity and in the angular neighborhood to the planes normal to the CPCA-axes. We apply our symmetry descriptor to the spherical shape descriptor computed by the Gaussian Euclidean Distance Transform [KFR04] of the surface. Measuring imperfect symmetry is used in step 4 ( $N_{GA}(\mathcal{M}) = 0$ ) 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.

## 4.2 Local Translational Invariance

Traditionally, in geometry, the translational symmetry is the invariance of an infinite object after 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.

Finding the direction that maximizes the local translational invariance is the last step in our general alignment scheme. More precisely, we look for local translational symmetries with respect to all directions perpendicular to the first good alignment axis (cf. section 4). To do this we need to compute a shape descriptor  $f$  defined over a one-dimensional interval, that represents a 3D model along a given direction. We need also to define a measure of symmetry for  $f$  with respect to local translation along this direction. For this purpose, we describe a method which selects 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  $\rho$  from the center of the coordinate system. By taking  $I_{\mathbf{d}}$  the interval defined by the limits of the 3D surface  $\mathfrak{S}$  in the direction  $\mathbf{d}$ , we represent  $\mathfrak{S}$  as follows:

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

where  $\mathcal{S}_{\mathbf{d}}(\rho)$  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 what follows, we consider a function  $f_{\mathbf{d}}$  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  $\mathcal{S}_{\mathbf{d}}(\rho)$  for any  $\rho \in I_{\mathbf{d}}$ . Methods computing the shape descriptor  $f_{\mathbf{d}}$  are given in section 4.2.4.

#### 4.2.2 Local Translational Invariance Cost

**Definition 1.** Given a shape descriptor  $f_{\mathbf{d}}$  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  $I_{\mathbf{d}}$  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  $I_{\mathbf{d}}$  where  $f_{\mathbf{d}}$  has local translational invariance along  $\mathbf{d}$  in  $I_i$ :

$$LTIC(f_{\mathbf{d}}) = \sum_{I_i \in \mathcal{I}} \mathcal{L}(I_i) \quad (2)$$

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

#### 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  $\mathbf{n}$  computed in section 4.

In order to evaluate the LTIC, we generate a set of direction vectors perpendicular to the first good alignment axis  $\mathbf{n}$ , obtained by rotating the coordinate system about  $\mathbf{n}$ :

Let  $\mathcal{R}_{\mathbf{n}}^K$  be the set generated by the transformation  $R_k$  which is the rotation about  $\mathbf{n}$  by the angle  $\theta_k = \frac{\pi k}{K}$  where  $0 \leq k < K$ :

$$R_k = R_{(1,0,0)}^{\theta_k} \cdot R',$$

$$\text{with} \quad R_{(1,0,0)}^{\theta_k} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_k) & -\sin(\theta_k) \\ 0 & \sin(\theta_k) & \cos(\theta_k) \end{bmatrix}$$

and  $R'$  (cf. section 4) is the matrix that contains  $\mathbf{n}$  in the first row.

In what follows, we associate to each  $R_k \in \mathcal{R}_{\mathbf{n}}^K$  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 introduced. 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  $LTIC(f_k)$ :

$$R_{ga} = \operatorname{argmax}_{R_k \in \mathcal{R}_{\mathbf{n}}^K} LTIC(f_k). \quad (3)$$

The 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  $\mathcal{M}$  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  $\theta_k$ ,
  - (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  $\mathcal{L}_{I_k}$  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  $\mathbf{n}$ . 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  $(\mathbf{n}, \mathbf{d}_k, \mathbf{n} \wedge \mathbf{d}_k)$ , and  $\mathcal{S}_k(\rho)$  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  $F_k$  are introduced to represent  $\mathfrak{S}_k$ . They use only one coordinate (along the axis  $\mathbf{n} \wedge \mathbf{d}_k$ ) as the axis  $\mathbf{n}$  is already selected in the good coordinate system and  $\mathbf{d}_k$ -coordinate is fixed in  $\mathfrak{S}_k(\rho)$ .

- *Global average description  $G_k$ :*

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

where  $\mathcal{A}_k(\rho) = \iint_{\mathbf{p} \in \mathcal{S}_k(\rho)} ds$  denotes the area of  $\mathcal{S}_k(\rho)$ .

- *Global extremum description  $E_k$ :*

$$E_k(\rho) = \left( \min_{\mathbf{p} \in \mathcal{S}_k(\rho)} ((\mathbf{n} \wedge \mathbf{d}_k)^T \cdot \mathbf{p}), \max_{\mathbf{p} \in \mathcal{S}_k(\rho)} ((\mathbf{n} \wedge \mathbf{d}_k)^T \cdot \mathbf{p}) \right).$$

- *Vector shape description  $F_k$ :*

Let  $J_k$  be the interval defined by the limits of the 3D surface  $\mathfrak{S}_k$  in the direction  $\mathbf{n} \wedge \mathbf{d}_k$ . The bounding box of mesh  $\mathfrak{S}_k$  is partitioned into  $M_k$  cells along the direction  $\mathbf{n} \wedge \mathbf{d}_k$  following a regular sampling of  $J_k$ . Then,  $\mathcal{S}_k^j(\rho)$  is the intersection of the shape  $\mathcal{S}_k(\rho)$  and the  $j^{th}$  cell.

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

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

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

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

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

### 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  $\frac{\mathcal{L}(I_k)}{N_k}$  has a fixed length  $2\delta$  (cf. 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  $\mathcal{M}$  is scaled (see the first step of the algorithm in section 4.2.3).

$$N_k = \lfloor N \mathcal{L}(I_k) \rfloor. \quad (4)$$

Similarly, we take  $M_k = \lfloor N \mathcal{L}(J_k) \rfloor$  when computing the vector shape description  $F_k$ .

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 and choose a normalized error  $\epsilon_N \simeq 0$  fixed for all 3D models. We consider that  $f_k(i) = f_k(i')$  if  $dist(f_k(i), f_k(i')) < \epsilon_N$ . In our implementation,  $dist(\cdot)$  is the usual Euclidean



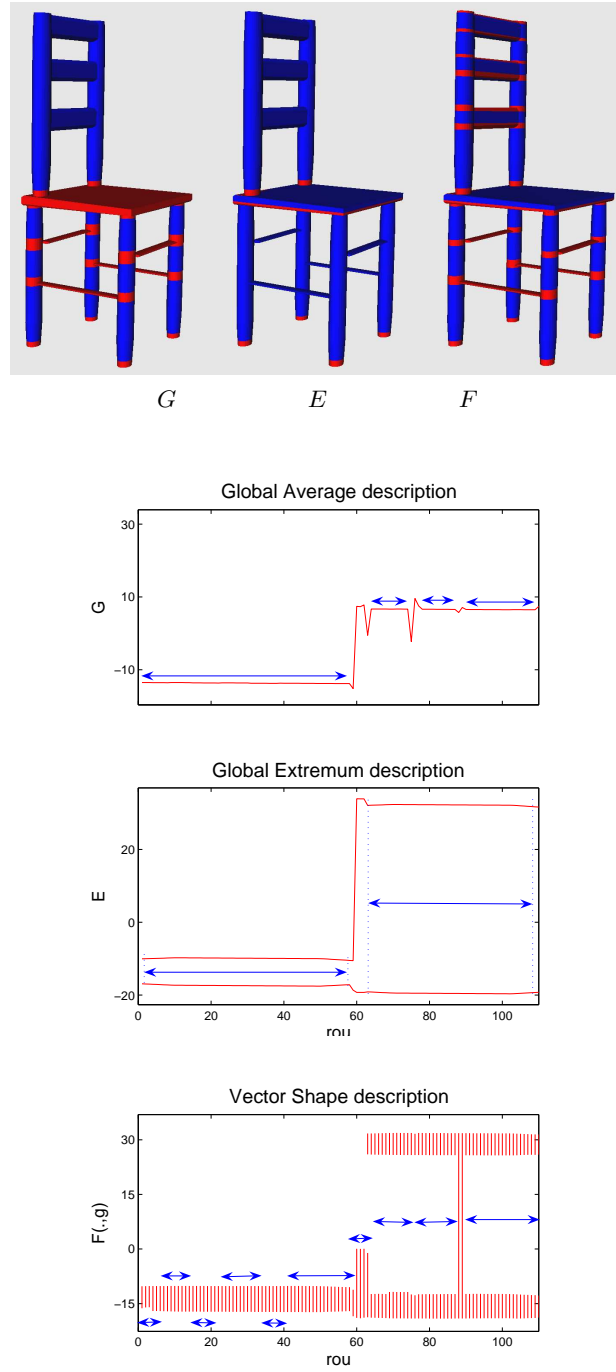


Figure 4: The three shape descriptors  $G$ ,  $E$  and  $F$  corresponding to the chair model are computed along the vertical direction. Surface positions associated to intervals of local translational invariance are colored in blue while the others are colored in red.

norm  $\|\cdot\|_1$  for the global average description and the global extremum description; it is defined as follows for the vector shape description:

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

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

The distance  $d_{F_k}(i, i')$  leads to a complexity  $\mathcal{O}(M_k M_k)$ . In order to reduce the computing time, we reduce the number of distance evaluations  $\|g_k^j(i) - g_k^{j'}(i')\|_1$ ,  $1 \leq j, j' \leq M_k$ . We make an a priori coherence assumption: we suppose that the index  $j_m = \text{argmin}_{1 \leq j' \leq M_k} \|g_k^j(i) - g_k^{j'}(i')\|_1$  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 not 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 [SMKF04] consisted of 907 polygonal models categorized into 92 distinct classes. We found that our approach produces coordinate frames that are robust and semantically correct for most of the models. Figure 5 shows a number of models from different classes aligned by our method. Moreover, our approach provides similar alignments for models belonging to the same class, see for example the alignments of the mailbox class in Figure 6.

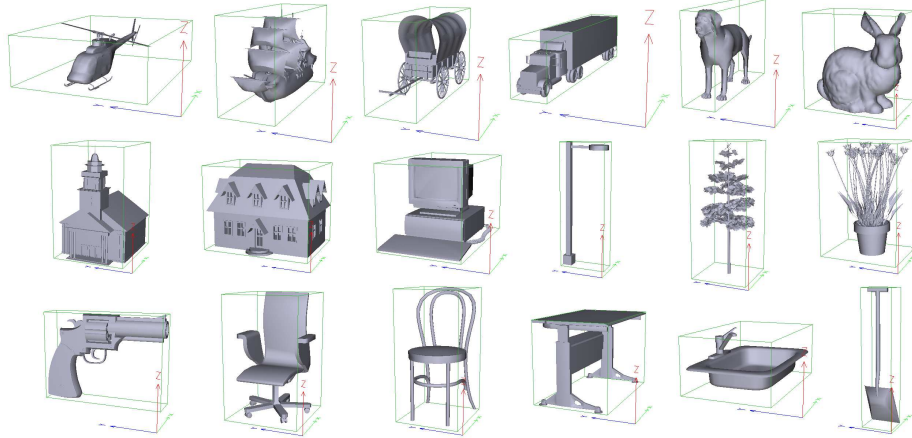


Figure 5: Alignments of models from different classes using our method.

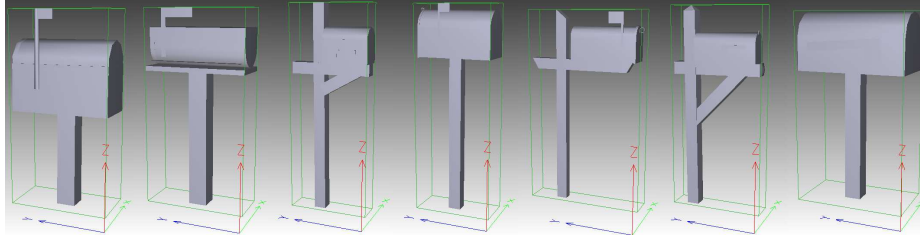


Figure 6: Alignments of the mailbox class using our method.

$N_{GA}$	0	1	2 <sup>+</sup>
Number of 3D models	181	467	259
Distribution( $N_{GA}$ )	20.0%	51.5%	28.5%

Table 1: Repartition of 3D models of the Test PSB database with respect of the number of retained CPCA-axes in our alignment method

We evaluated 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.

As Table 1 indicates, 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 the descriptors  $G$  and  $E$ .

To measure the efficacy of our alignment algorithm in shape retrieval task, we apply it as a normalization step in a general retrieval process. We use the shape descriptor DLA [CVB07] 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 compute Precision-Recall diagrams commonly used in information search (the query is not counted in the answer as in [Vra04]) and four quantitative measures for evaluating query results (see [SMKF04] for a description of this measures): **(1)** The Nearest Neighbor (NN), **(2)** The First Tier (FT), **(3)** the Second Tier (ST), **(4)** the Discounted Cumulative Gain (DCG).

Comparing the curves as well as the NN, FT, ST and DCG values in Figure 7, we conclude that our alignment method clearly outperforms the CPCA.

### 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 you can see in table 1, the CPCA provides, in our general algorithm, a quick alignment for 28.5% of the models that have at least two good alignment

Class	Nbr ( $\mathcal{M}$ )	CPCA	Our Method		
			$G$	$E$	$F$
Helicopter aircraft	18	77.7%	94.4%	100%	100%
Enterprise like 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 home building	14	35.7%	85.7%	92.9%	92.9%
Two story home 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 display	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 plant	26	53.8%	92.3%	88.5%	100%
Conical tree	10	70.0%	90.0%	80.0%	90.0%
Large sail boat sailboat	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  $F$ .

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 that don't have any good alignment axis within CPCA-axes.

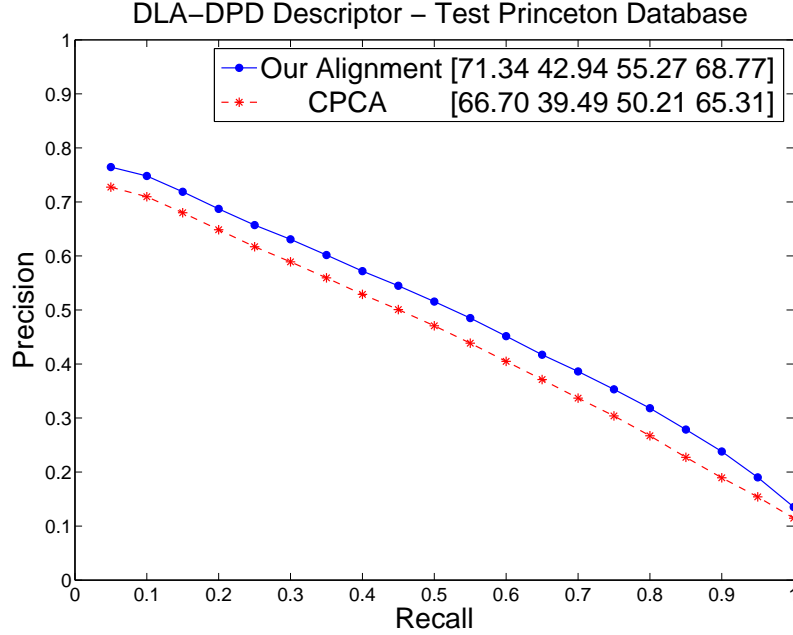


Figure 7: Average Precision-recall curves using the CPCA and our alignment followed by the depth line-based approach DLA (with dynamic programming distance DPD, 6 depth images of size 32x32). The mean NN, FT, ST and DCG values are given in the legends.

## 6 Conclusion

We have presented a new alignment method for 3D models. It retains the principal axes of the CPCA with respect to 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 about a given direction. This measure is used to compute the remaining alignment axes.

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.

## Appendix

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

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

*Proof.*

The vector  $\mathbf{n}$  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  $\mathbf{g}$  is the center of gravity of  $\mathfrak{S}$ .

We first construct the covariance matrix of  $\mathfrak{S}$ .

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

By previous Lemma  $\mathbf{g} \in \pi$  and by orthogonal projection for all  $\mathbf{v}$  onto  $\pi$  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  $\pi$  is the eigenvector of  $\mathfrak{S}$  and  $\frac{1}{S} \iint_{\mathbf{v}} d_v^2$  is the corresponding eigenvalue.  $\square$

## References

- [ASCE02] N. Aspert, D. Santa-Cruz, and T. Ebrahimi. MESH: Measuring errors between surfaces using the Hausdorff distance. In *IEEE International Conference on Multimedia and Expo (ICME 02)*, pages 705–708, Lausanne, Switzerland, August 2002.
- [BKSV04] B. Bustos, D. A. Keim, T. Schreck, and D. Vranic. An experimental comparison of feature-based 3D retrieval methods. In *3DPVT'04*, Thessaloniki, Greece, September 2004.
- [CO02] D. Chen and M. Ouhyoung. A 3D model alignment and retrieval system. In *International Computer Symposium, Workshop on Multimedia Technologies*, Hualien, Taiwan, 2002.
- [CRS98] P. Cignoni, C. Rocchini, and R. Scopigno. Metro: Measuring error on simplified surfaces. *Computer Graphics Forum*, 17(2):167–174, June 1998.
- [CVB07] M. Chaouch and A. Verroust-Blondet. 3D model retrieval based on depth line descriptor. In *IEEE International Conference on Multimedia & Expo (ICME'07)*, Beijing, China, July 2007.
- [DFN92] B. A. Dubrovin, A. T. Fomenko, and S. P. Novikov. *Modern geometry, methods and applications. Part I. , The Geometry of surfaces, transformation groups, and fields*. Springer-Verlag, 1992.
- [Fer00] 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.
- [Kaz07] M. Kazhdan. An approximate and efficient method for optimal rotation alignment of 3D models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(7):1221–1229, July 2007.
- [KFR04] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz. Symmetry descriptors and 3D shape matching. In *Symposium on Geometry Processing*, pages 117–126, July 2004.
- [MGP06] N. J. Mitra, L. Guibas, and M. Pauly. Partial and approximate symmetry detection for 3D geometry. *ACM Trans. on Graphics*, 25(3):560–568, 2006.
- [MIK93] P. Minovic, S. Ishikawa, and K. Kato. Symmetry identification of a 3-D object represented by octree. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(5):507–514, 1993.
- [MSHS06] A. Martinet, C. Soler, N. Holzschuch, and F. X. Sillion. Accurate detection of symmetries in 3D shapes. *ACM Trans. on Graphics*, 25(2):439–464, 2006.
- [PRM<sup>+</sup>00] E. Paquet, M. Rioux, A. Murching, T. Naveen, and A. Tabatabai. Description of shape information for 2-D and 3-D objects. *Signal Processing : Image Communication*, 16:103–122, 2000.

- 
- [PSG<sup>+</sup>06] J. Podolak, P. Shilane, A. Golovinskiy, S. Rusinkiewicz, and T. Funkhouser. A planar-reflective symmetry transform for 3D shapes. *ACM Trans. on Graphics*, 25(3):549–559, 2006.
- [Rus07] R.M. Rustamov. Augmented symmetry transforms. In *SMI'07*, pages 13–20, Lyon, France, 2007.
- [SMKF04] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The Princeton shape benchmark. In *SMI'04*, pages 167–178, Genova, Italy, June 2004.
- [SS97] C. Sun and J. Sherrah. 3D symmetry detection using the extended gaussian image. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(2):164–168, 1997.
- [Vra04] D.V. Vranic. *3D Model Retrieval*. PhD thesis, U. of Leipzig, 2004.
- [VSR01] D. Vranic, D. Saupe, and J. Richter. Tools for 3D-object retrieval: Karhunen-Loeve transform and spherical harmonics. In *2001 Workshop Multimedia Signal Processing*, Cannes, France, October 2001.
- [ZP04] T. Zaharia and F. Prêteux. 3D versus 2D/3D shape descriptors: A comparative study. In *SPIE Conf. on Image Processing: Algorithms and Systems*, CA, USA, January 2004.
- [ZPA95] H. Zabrodsky, S. Peleg, and D. Avnir. Symmetry as a continuous feature. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(12):1154–1166, 1995.



## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Related Work</b>	<b>4</b>
<b>3</b>	<b>Symmetry &amp; 3D Objects</b>	<b>5</b>
3.1	Plane Reflection Symmetry Analysis . . . . .	5
3.2	Principal Components & Plane Reflection Symmetry Analysis . .	6
<b>4</b>	<b>Alignment of 3D Objects</b>	<b>7</b>
4.1	Plane Reflection Symmetry . . . . .	8
4.1.1	Continuous Symmetry Distance . . . . .	8
4.1.2	Symmetry Descriptors . . . . .	10
4.2	Local Translational Invariance . . . . .	10
4.2.1	Shape Description over 1-D Space . . . . .	11
4.2.2	Local Translational Invariance Cost . . . . .	11
4.2.3	LTIC for Alignment . . . . .	11
4.2.4	Three shape descriptor models for $f_k$ . . . . .	12
<b>5</b>	<b>Experimental Results</b>	<b>15</b>
<b>6</b>	<b>Conclusion</b>	<b>18</b>



---

Centre de recherche INRIA Paris – Rocquencourt  
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Centre de recherche INRIA Futurs : Parc Orsay Université - ZAC des Vignes  
4, rue Jacques Monod - 91893 ORSAY Cedex

Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique  
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex

Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex

Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier

Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

---

Éditeur  
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)  
<http://www.inria.fr>  
ISSN 0249-6399