



## Continuous Generalized Procrustes analysis



Laura Igual<sup>a,b,\*</sup>, Xavier Perez-Sala<sup>b,c,d</sup>, Sergio Escalera<sup>a,b</sup>, Cecilio Angulo<sup>d</sup>,  
Fernando De la Torre<sup>e</sup>

<sup>a</sup> Universitat de Barcelona, Gran Via 585, 08007 Barcelona, Spain

<sup>b</sup> Computer Vision Center, Universitat Autònoma de Barcelona, Building O, Barcelona, Spain

<sup>c</sup> Fundació Privada Sant Antoni Abat, Rambla de l'Exposició, 59-69, Vilanova i la Geltrú, Spain

<sup>d</sup> Universitat Politècnica de Catalunya, Av. Víctor Balaguer 1, Vilanova i la Geltrú, Spain

<sup>e</sup> Carnegie Mellon University, Robotics Institute, 5000 Forbes Avenue, Pittsburgh, PA, USA

### ARTICLE INFO

#### Article history:

Received 23 July 2012

Received in revised form

23 July 2013

Accepted 3 August 2013

Available online 13 August 2013

#### Keywords:

Procrustes analysis

2D shape model

Continuous approach

### ABSTRACT

Two-dimensional shape models have been successfully applied to solve many problems in computer vision, such as object tracking, recognition, and segmentation. Typically, 2D shape models are learned from a discrete set of image landmarks (corresponding to projection of 3D points of an object), after applying Generalized Procrustes Analysis (GPA) to remove 2D rigid transformations. However, the standard GPA process suffers from three main limitations. Firstly, the 2D training samples do not necessarily cover a uniform sampling of all the 3D transformations of an object. This can bias the estimate of the shape model. Secondly, it can be computationally expensive to learn the shape model by sampling 3D transformations. Thirdly, standard GPA methods use only one reference shape, which can might be insufficient to capture large structural variability of some objects.

To address these drawbacks, this paper proposes continuous generalized Procrustes analysis (CGPA). CGPA uses a continuous formulation that avoids the need to generate 2D projections from all the rigid 3D transformations. It builds an efficient (in space and time) non-biased 2D shape model from a set of 3D model of objects. A major challenge in CGPA is the need to integrate over the space of 3D rotations, especially when the rotations are parameterized with Euler angles. To address this problem, we introduce the use of the Haar measure. Finally, we extended CGPA to incorporate several reference shapes. Experimental results on synthetic and real experiments show the benefits of CGPA over GPA.

© 2013 Elsevier Ltd. All rights reserved.

## 1. Introduction

Procrustes analysis (PA) [1–3] is a form of statistical shape analysis used to analyze the distribution of a set of shapes. Given two shapes PA “superimposes” both shapes by optimally translating, rotating and scaling one shape towards the other. If more than two shapes are registered, the problem is typically known as generalized Procrustes analysis (GPA). GPA has been typically used in computer vision as a first step to build 2D models of shape or appearance of objects. These 2D models have been applied to solve problems such as object recognition [4,5], facial feature detection and tracking [6,7] and image segmentation [8,9]. In particular, Point distribution models (PDMs) and active shape models (ASMs) [11] are among the most popular techniques to learn 2D objects

models. PDMs and ASMs build the shape models from a 2D training set of image landmarks. In PDMs and ASMs, first GPA is used to remove rigid transformations and, then principal component analysis (PCA) is applied to construct a subspace that models the variation of the normalized shapes [11].

Fig. 1 (left) illustrates the GPA process of building shape models for PDM or ASM: given a set of 2D views of one or several 3D rigid or non-rigid objects under several configurations, the shape of the object is represented by several landmarks that are consistently labeled across view-points. Observe that if the object is rigid and the projection is orthographic, all views can be represented using a three-dimensional subspace [10]. Given the set of shapes (2D projections across views, objects or non-rigid transformations of 3D objects), GPA aligns the shapes using a rigid transformation (e.g., Euclidean or affine) to a 2D reference shape such that it minimizes the least-squares error. Although GPA has been extensively used, it suffers from three main limitations when modeling non-rigid transformations of a 3D object or a class of 3D objects: (i) 2D training samples do not necessarily cover a uniform sampling of all 3D transformations of an object, thereby biasing the estimate of the 2D models towards some particular

\* Corresponding author at: Universitat de Barcelona, Gran Via 585, 08007 Barcelona, Spain. Tel.: +34 934 020 854; fax: +34 934 021 601.

E-mail addresses: [ligual@ub.edu](mailto:ligual@ub.edu), [lauraigual@gmail.com](mailto:lauraigual@gmail.com) (L. Igual), [xavier.perez-sala@upc.edu](mailto:xavier.perez-sala@upc.edu), [xavips@gmail.com](mailto:xavips@gmail.com) (X. Perez-Sala), [sergio@maia.ub.es](mailto:sergio@maia.ub.es) (S. Escalera), [cecilio.angulo@upc.edu](mailto:cecilio.angulo@upc.edu) (C. Angulo), [ftorre@cs.cmu.edu](mailto:ftorre@cs.cmu.edu) (F. De la Torre).

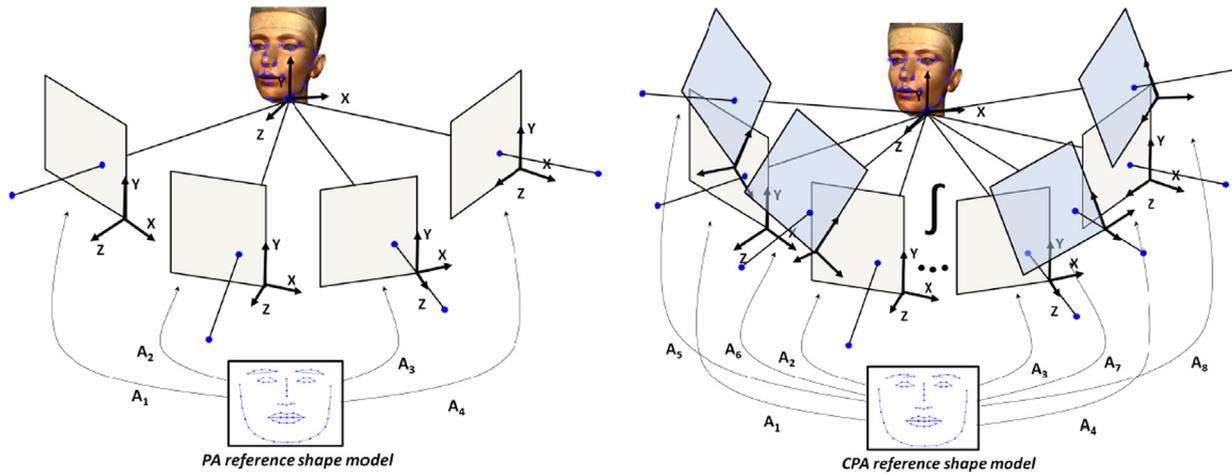


Fig. 1. Illustration of GPA (left) and CGPA (right) to construct 2D shape models from 3D objects.

configuration; (ii) it is computationally expensive to compute a rich set of 2D projections from all possible 3D transformations of a set of objects; and (iii) the large variability of the object class cannot necessarily be well registered with only one reference shape.

In order to deal with these limitations, we propose *continuous generalized Procrustes analysis* (CGPA). CGPA generalizes GPA using a *continuous* formulation that avoids the need to generate 2D projections from 3D configurations and uniformly covers the space of 3D transformations. Fig. 1 (right) illustrates the main idea behind CGPA, CGPA integrates over the space of 3D rotations avoiding the need to compute 2D projections. The continuous approach proposed in this paper is efficient in space and time, and is not biased to non-uniform sampling of the input space. A requirement of CGPA is to have access to a 3D mesh of several configurations of one or more 3D object, which is a realistic assumption in several computer vision problems. It is important to notice that building 2D models from 3D samples is a problem that has been relatively unexplored in computer vision [13,27].

A major challenge of the proposed work is to integrate 3D objects over the special orthogonal group in 3D:  $SO(3)$ . While there are many possible parameterizations of  $SO(3)$ , we have chosen Euler angles because it is easy to determine the relation between the rotation limits and the integration domain (unlike other parameterizations such as quaternions). However, Euler angles suffer from well-known problems such the lack of uniform integration over the space of rotations [12] or the gimbal lock effect. In this paper, to address these problems we use the Haar measure in the definition of the integral and uniformly integrate over the space of rotations. In addition, in some cases a simple mean in the case of GPA is not enough to model the variability of objects across view-points, and we propose a multi-reference CGPA by using several reference shapes. Experimental results in several synthetic and real experiments show the benefits of CGPA over GPA. A preliminary version of this work was presented in [13].

The rest of the document is organized as follows: Section 2 reviews previous work in GPA and functional data analysis (FDA), Section 3 gives the mathematical background necessary for CGPA formulation and Section 4 motivates and derives CGPA. Section 5 reports our experimental results and Section 6 presents the conclusions and outlines future lines of research. Finally, in Appendix A we review the GPA fitting algorithm.

## 2. Previous work

This section reviews previous work within the field of computer vision on Procrustes analysis and functional data analysis (FDA).

### 2.1. Generalized Procrustes analysis (GPA)

Let  $\mathbf{D} = [(\mathbf{D}_1^{(2)})^T, \dots, (\mathbf{D}_m^{(2)})^T]^T$  be a set of  $m$  shape samples that we wish to align. Note that the super-script  $(2)$  indicates that the shapes are 2D. Shape samples are represented as  $\ell$  2D landmarks embedded in an  $\mathbb{R}^{2 \times \ell}$  matrix  $\mathbf{D}_i^{(2)}$  (see footnote<sup>1</sup> for notation)

$$\mathbf{D}_i^{(2)} = \begin{pmatrix} x_{i1} & \dots & x_{i\ell} \\ y_{i1} & \dots & y_{i\ell} \end{pmatrix}.$$

GPA optimizes over the 2D geometric transformation  $\mathbf{T}_i$  (e.g., affine, Euclidean) that aligns each sample with respect to the reference shape, by minimizing the energy of the *reference-space model* (see Fig. 2 (right)) [14]

$$E_R(\mathbf{M}, \mathbf{A}) = \sum_{i=1}^m \|\mathbf{T}_i \mathbf{D}_i^{(2)} - \mathbf{M}\|_F^2, \quad (1)$$

where  $\mathbf{M} \in \mathbb{R}^{2 \times \ell}$  represents the reference shape, and  $\mathbf{T}_i \in \mathbb{R}^{2 \times 2}$  corresponds to the rigid transformation for the shape sample  $\mathbf{D}_i^{(2)}$ . GPA can also be optimized using the *data-space model* (see Fig. 2 (left)) in the following way [14]:

$$E_D(\mathbf{M}, \mathbf{A}) = \sum_{i=1}^m \|\mathbf{D}_i^{(2)} - \mathbf{A}_i \mathbf{M}\|_F^2, \quad (2)$$

where  $\mathbf{A}_i$  is the inverse transformation of  $\mathbf{T}_i$  and  $\mathbf{A} = [\mathbf{A}_1^T, \dots, \mathbf{A}_m^T]^T \in \mathbb{R}^{2m \times 2}$  corresponds to the rigid transformation for the reference shape  $\mathbf{M}$ .

Recall that the error function of the reference-space model minimizes the difference between the reference shape and the registered shape data; in the data-space model, the error function compares the observed shape points with the transformed reference shape, i.e., shape points predicted by the model and based on the notion of average shape [15]. This difference between the two models leads to different properties. Since the reference-space cost is a sum of squares and it is linear in the optimization parameters, it can be optimized via alternated least square methods. In contrast, the data-space cost is a bilinear problem and non-convex (in general). If there are no missing data, the data-space model can be solved using singular value decomposition (SVD). A major advantage of the data-space model is that it is

<sup>1</sup>  $\mathbb{N}$  and  $\mathbb{R}$  denote the set of natural and real numbers, respectively, and  $\mathbb{R}^d$  denotes the set of real vectors of dimension  $d$ . We assume that  $m, d, l, n, p, i \in \mathbb{N}$ . A bold capital letter denotes a matrix,  $\mathbf{D}$ ; a bold lower-case letter a column vector,  $\mathbf{d}$ ,  $\mathbf{D}_i$  represents the  $i$ th block matrix of the matrix  $\mathbf{D}$ . All non-bold letters denote scalar variables.  $\|\mathbf{D}\|_F^2 = \text{Tr}(\mathbf{D}^T \mathbf{D})$  designates the square of the Frobenius norm of a matrix. The set operation  $\mathbf{Q} \setminus \mathbf{F}$  stands for the set difference of  $\mathbf{Q}$  and  $\mathbf{F}$ .  $\nabla_u f$  is the gradient operator with respect to  $u$  of the function  $f$ .

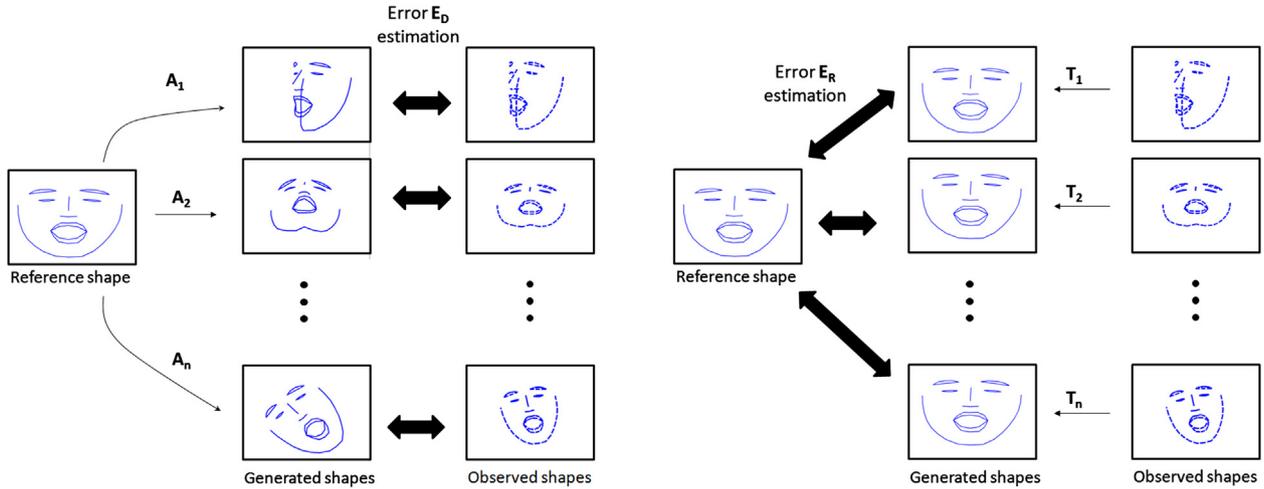


Fig. 2. Left: data-space model. Right: reference-space model. Note that  $A_i = T_i^{-1}$ .

*gauge invariant* [14] (i.e., the cost does not depend on the coordinate frame in which the reference shape and the transformations are expressed).

In the computer vision literature, the problem of PA has been extensively explored. Procrustes analysis has been applied to aligning shapes (e.g., [16]) and appearance (e.g. [6,17–19]).

In the last 10 years, several algorithms that align data with respect to geometric transformations using appearance features have become popular. Frey and Jojic [20] proposed a method for learning a factor analysis model that is invariant to geometric transformations. The computational cost of this method grows polynomially with the number of possible spatial transformations and it can be computationally intensive when working with high-dimensional motion models. To improve upon that, De la Torre and Black [17] proposed parameterized component analysis: a gradient-based method that learns a PCA model invariant to affine transformations. Baker et al. [19] showed how to learn active appearance models (AAMs) in a way that are invariant to rigid and non-rigid motion. De la Torre and Nguyen [6] extended parameterized component analysis to deal with non-linear appearance representations (using kernels) and non-rigid transformations. Miller et al. proposed the congealing method [18], which uses an entropy measure to align images with respect to the distribution of the data. Cox et al. [19] extended [18] through a least-squares optimization. Kookinos and Yuille [21] proposed a probabilistic framework and extended previous approaches [17–19] to deal with articulated objects using a Markov random field (MRF) on top of AAMs.

Previous work on PA uses 2D shapes or images, and hence suffer from non-uniform sampling and high computational complexity. If one has access to the 3D model of the object, CGPA can provide a better 2D model of the object.

Pizarro et al. [16] have recently proposed a convex approach for GPA based on the reference-space model. In their case, the cost function is expressed with a quaternion parametrization which allows conversion to a sum of squares program (SOSP). Finally, the equivalent semi-definite program of a SOSP relaxation is solved using a convex optimization tool and providing the global minimum.

### 2.2. Functional data analysis (FDA)

Our work is related to previous work on FDA [22]. FDA [22] is a branch of statistics that analyzes data providing formation about

functions. FDA methods are adaptations of classical multivariate methods such as PCA [22], Linear Discriminant Analysis (LDA) or analysis of variance (ANOVA) [23].

There have been several works in computer vision that make use of FDA. Ormeneit et al. [24] proposed a robust automatic method for modeling cyclic 3D human motion, such as person walking sequence, using motion-capture data. The pose of the body is represented as a time series of joint angles which are automatically segmented into a sequence of motion cycles. The mean and the functional principal components of these cycles are computed using a new algorithm that enforces smooth transitions between the cycles by operating in the Fourier domain. An advantage of this method is that it automatically deals with noise and missing data. The model is later used for Bayesian tracking of 3D human motion. Levin and Shashua [25] applied a continuous formulation in the case of PCA to model faces under different illuminations. Their method integrates over the convex hull of the sample data, and achieves unbiased estimates of the principal components of the images.

### 3. Mathematical background

This section describes the mathematical background to our work. We review basic statements from the calculus of variations and integral calculus, as well as details regarding  $SO(3)$ , and measures defined on it.

#### 3.1. Calculus

Let  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  be a smooth scalar function. If  $\mathbf{x}^* \in \mathbb{R}^n$  is a solution of the problem

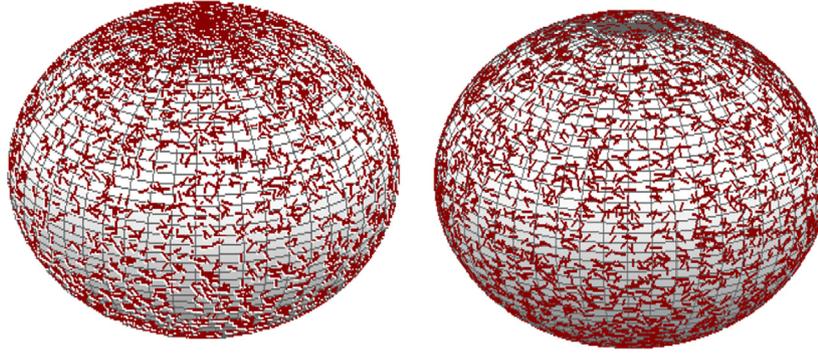
$$f(\mathbf{x}^*) = \min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}), \tag{3}$$

then the following equation is satisfied:

$$\nabla_{\mathbf{x}} f(\mathbf{x}^*) = 0, \tag{4}$$

where  $\nabla_{\mathbf{x}}$  is the gradient operator of the function  $f(\mathbf{x})$  with respect to  $\mathbf{x}$ .

Now let  $\Omega \subset \mathbb{R}^n$  be an open and a bounded subset, let  $F : \mathbb{R}^d \rightarrow \mathbb{R}$  be a mapping, and we want to find a solution,  $\mathbf{v}^* : \Omega \rightarrow \mathbb{R}^d$ , to the



**Fig. 3.** Non-uniform (left) and uniform (right) distributions of rotations in  $SO(3)$ . Rotation samples are represented by means of 10,000 rotated arrows onto the unit sphere surface.

following functional problem:

$$\int_{\Omega} F(\mathbf{v}^*(\mathbf{x})) \, d\mathbf{x} = \min_{\mathbf{v}} \left\{ \int_{\Omega} F(\mathbf{v}(\mathbf{x})) \, d\mathbf{x} \right\}, \quad (5)$$

where the minimum is taken among all the functions  $\mathbf{v} : \Omega \rightarrow \mathbb{R}^d$  belonging to  $L^p(\Omega; \mathbb{R}^d) := \{\mathbf{v} : \Omega \rightarrow \mathbb{R}^d : \mathbf{v} \text{ is measurable and } \int_{\Omega} |\mathbf{v}(\mathbf{x})|^p \, d\mathbf{x} < \infty\}$ .

Then, it can be shown that the function  $\mathbf{v}^* \in L^p(\Omega; \mathbb{R}^d)$  satisfies  $\nabla_{\mathbf{v}} F(\mathbf{v}^*(\mathbf{x})) = 0, \quad \forall \mathbf{x} \in \Omega \setminus \Gamma,$  (6)

with  $\int_{\Gamma} 1 \, d\mathbf{x} = 0$ , i.e.,  $\Gamma$  is a null set.

The interested reader is referred to [28,26] for a more detailed review of the calculus of variations.

In order to manipulate multiple integrals, Fubini's Theorem determines conditions under which it is possible to compute a multiple integral using iterated integrals [29].

Let  $\Omega_p \subset \mathbb{R}^p$  and  $\Omega_q \subset \mathbb{R}^q$  be complete measure spaces. Let  $f(x, y) \in L(\Omega_p \times \Omega_q; \mathbb{R}^d)$ , i.e.

$$\int_{\Omega_p \times \Omega_q} |f(x, y)| \, d(x, y) < \infty, \quad (7)$$

with  $f(x, y)$  measurable, then

$$\int_{\Omega_p \times \Omega_q} f(x, y) \, d(x, y) = \int_{\Omega_p} \left( \int_{\Omega_q} f(x, y) \, dy \right) \, dx = \int_{\Omega_q} \left( \int_{\Omega_p} f(x, y) \, dx \right) \, dy.$$

The integral with respect to a product of two measures can be calculated as iterated integrals with respect to those two measures.

### 3.2. Integration over the $SO(3)$ group

$SO(3)$  forms a group whose action is the composition of all rotations. Each rotation is a linear transformation that preserves the length and spatial orientation of vectors.

Next, we will show why Euler angle parametrization of 3D rotations can be adopted in the CGPA formulation when defining a proper Haar measure, while it is unclear how to do it for quaternions.

#### 3.2.1. $SO(3)$ parameterizations

There are several parameterizations for 3D rotations around the origin, but the most common ones are Euler angles and quaternions. Euler angles encode orientations in the 3D Euclidean space  $\mathbb{R}^3$  through the composition of three rotations  $(\alpha, \beta, \gamma)$ , each one around a single axis of a basis. The final rotation is obtained by multiplying three rotation matrices,  $\mathbf{R} = \mathbf{R}_z(\gamma)\mathbf{R}_y(\beta)\mathbf{R}_x(\alpha)$  where

$$\mathbf{R}_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix},$$

$$\mathbf{R}_y(\beta) = \begin{pmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{pmatrix},$$

$$\mathbf{R}_z(\gamma) = \begin{pmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Euler angles are frequently used to define three dimensional rotations, nevertheless, they lead to a non-uniform distribution of rotations (see Fig. 3 (left), image better seen in color) when the three angles are uniformly sampled among their typical domains [12]. Fig. 3 (left) represents 10,000 samples of 3D angles that have been sampled uniformly in the Euler space. That is, for each of the roll, yaw and pitch angles, we have randomly selected 10,000 values between  $\alpha, \gamma = \mathcal{U}(-\pi, \pi]$  and  $\beta = [-\pi/2, \pi/2]$ , respectively.  $\mathcal{U}$  means that angles can take values uniformly distributed on the given interval. Each of these three angles are represented by a point in a sphere (two angles) and there is a line in the tangent space that conveys information of the other angle. We can observe in this figure how the higher density of points is located in the poles, and hence we did not achieve a uniform distribution over the surface.

Beyond the non-uniform sampling the Euler angles also suffer from the Gimbal Lock problem [31]. Gimbal Lock occurs on rotations in a 3D space, when two of the three axes are parallel. One degree of freedom is lost and, therefore, only rotations in 2D space can be performed. A simple example to help to understand this issue arises when using the convention Z–Y–Z, i.e., first, a rotation on the Z-axis by the angle  $\alpha$ , followed by a turn on the rotated Y-axis by the angle  $\beta$  and, finally, a rotation by the angle  $\gamma$  on the new Z-axis. If  $\beta = 0$ , it produces a rotation by the angle  $\delta_1 = \alpha + \gamma$ , only on the Z-axis. In this case, the system loses a degree of freedom and it is “locked” rotating in a degenerate 2D space.

Quaternions are generally used as a standard solution to these issues. Quaternions were conceived by Hamilton [33] by extending complex numbers  $\mathbf{q} = [a, bi, cj, dk]$ . Each unit quaternion ( $\|\mathbf{q}\| = 1$ ) can be interpreted as a 4D point in the unit 3-sphere  $S^3$ , which also represents a rotation in the 3D space. For any unit quaternion,  $\mathbf{q} = [\cos(\theta/2), \sin(\theta/2)\hat{\mathbf{u}}]$ , and for any vector  $\mathbf{v} \in \mathbb{R}^3$ , the action of the triple product

$$\mathbf{v}_R = \mathbf{q}\mathbf{v}\mathbf{q}^* = \mathbf{R}_q\mathbf{v} \text{ where}$$

$$\mathbf{R}_q = \begin{pmatrix} 1-2(c^2+d^2) & 2(bc-ad) & 2(bd+ac) \\ 2(bc+ad) & 1-2(b^2+d^2) & 2(cd-ab) \\ 2(bd-ac) & 2(cd+ab) & 1-2(b^2+c^2) \end{pmatrix},$$

may be geometrically interpreted as a rotation of the vector  $\mathbf{v}$  through an angle  $\theta$ , with  $\hat{\mathbf{u}}$  being the axis of rotation.

Fig. 3 (right) illustrates the uniform distribution achieved with a quaternion representation using the method proposed by [32], where the four quaternion parameters  $q = [\cos(\theta_2)r_2, \sin(\theta_1)r_1, \cos(\theta_1)r_1, \sin(\theta_2)r_2]$  are calculated through the use of three random variables  $X_i \in \mathcal{U}(0, 1), i = 1, \dots, 3$ , where  $\theta_1 = 2\pi X_2, \theta_2 = 2\pi X_3, r_1 = \sqrt{1-X_1}$  and  $r_2 = \sqrt{X_1}$ . Observe that in this case the distribution of angles is more homogeneous over the surface. The points are no longer concentrated on the poles, neither other regions of the sphere. Although the method presented in [32] achieves uniformly distributed random rotations, the relation between its three parameters and the rotation angles is not trivial. This fact makes the parametrization of CGPA using quaternions unfeasible, since a CGPA parametrization depending on the rotation angles is required, in order to bound the integration domains. Therefore, we have chosen Euler angles with Haar measure [34], which is detailed in the next section.

### 3.2.2. Haar measure

Integration of functions on a particular space involves the definition of a specific measure on that space [29].

The Haar measure is defined such that it assigns an “invariant volume” to subsets of locally compact topological groups and subsequently defines an integral for functions on those groups [34]. We may associate to any Haar measure  $\mu$  on a group a bounded linear functionals  $F \in L(\mathbb{R}^p; \mathbb{R})$

$$F(f) = \int_G f(\omega) d\mu(\omega).$$

As an example, the Haar measure on the group of rotations  $SO(3)$  [34, Section 7 of Chapter 1] leads to

$$\int_{SO(3)} f(\omega) d\mu(\omega) = \int_0^\pi d\gamma \int_0^{2\pi} d\beta \int_0^{2\pi} d\alpha \frac{1}{8\pi^2} \sin(\beta) f(\omega(\alpha, \beta, \gamma)). \tag{8}$$

Using the Haar measure, we obtain an invariant integral for functions on the rotation group. Thus, the problem of discrete non-uniform distribution using Euler angles discussed above is avoided in the definition of the integral.

## 4. Continuous generalized Procrustes analysis

In this section, we formulate the proposed continuous generalized Procrustes analysis (CGPA). CGPA extends GPA by adopting a continuous formulation that incorporates the information of all rigid 3D transformations.

### 4.1. Energy functional for continuous generalized Procrustes analysis

We formulate the problem of CGPA as an energy functional minimization, involving 3D landmarks of objects and continuous 3D rotations. The formulation uses the reference shape as the shape model in the data-space cost function (Fig. 2 (left)). Data-space is chosen because it is gauge invariant and its derivation is simpler than using reference-space. Our main assumption is that the best reference shape is the one that can approximate all possible 3D shape configurations of a given set of shapes. We interpret this in the following way: we consider a set of 3D shapes, we perform a predefined set of rotations, and we project them onto the 2D space. Then, we estimate the reference shape by aligning it with each shape configuration using an estimated affine transformation. Given the previous issues, we derive the energy functional to be minimized.

Let  $\mathbf{D}_i^{(3)} \in \mathbb{R}^{3 \times \ell}$  be a 3D shape described by  $\ell$  landmark points

$$\mathbf{D}_i^{(3)} = \begin{pmatrix} x_{i1} & \dots & x_{i\ell} \\ y_{i1} & \dots & y_{i\ell} \\ z_{i1} & \dots & z_{i\ell} \end{pmatrix}.$$

$\mathbf{D} = [\mathbf{D}_1^{(3)T}, \dots, \mathbf{D}_n^{(3)T}]^T$  is the set of samples, where  $n$  is the number of training examples, and  $\Omega = \{\omega = (\alpha, \beta, \gamma) \in \mathbb{R}^3\}$  is the set of 3D rotation domains, where  $\omega$  are the Euler angles. We assume the 3D data has been centered. CGPA minimizes the following energy functional:

$$E_{CGPA}(\mathbf{M}, \mathbf{A}_1(\omega), \dots, \mathbf{A}_n(\omega)) = \sum_{i=1}^n \int_{\Omega} F(\mathbf{M}, \mathbf{A}_i(\omega)) d\omega = \sum_{i=1}^n \int_{\Omega} \|\mathbf{PR}(\omega)\mathbf{D}_i^{(3)} - \mathbf{A}_i(\omega)\mathbf{M}\|_F^2 d\omega, \tag{9}$$

where  $\mathbf{M} \in \mathbb{R}^{2 \times \ell}$  is the mean reference shape and each matrix  $\mathbf{A}_i(\omega)$  in  $\mathbf{A}(\omega) = [\mathbf{A}_1(\omega)^T, \dots, \mathbf{A}_n(\omega)^T]^T \in \mathbb{R}^{2n \times 2}$ , is a linear transformation of the landmark coordinates. The matrix  $\mathbf{R}(\omega) \in \mathbb{R}^{3 \times 3}$  corresponds to the 3D rotation matrix that depends on the Euler angle  $\omega$  (as defined in Section 3.2.1), and  $\mathbf{P} \in \mathbb{R}^{2 \times 3}$  is the matrix describing the orthographic projection onto the plane  $Z=0$ , defined as

$$\mathbf{P} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}. \tag{10}$$

Note that for Euler angles  $\omega = (\alpha, \beta, \gamma)$ , the Haar measure can be computed for every domain  $\Omega$ . For instance, for a complete sphere this measure corresponds to  $d\omega = (1/8\pi^2) \sin(\beta) d\alpha d\beta d\gamma$ .

The main differences between the functional in Eq. (9) and the energy function in Eq. (2) are: (i) in the former, the affine transformations  $\mathbf{A}_i(\omega)$  are functions that depend on the Euler angles  $\omega$ , whereas in Eq. (2),  $\mathbf{A}_i$  are variables; (ii) the 2D shape projection depends directly on the 3D structure of the object  $\mathbf{D}_i^{(3)}$  and the 3D transformation parameters; and (iii) it is a continuous formulation, and discrete sums are extended by integrals. Note that  $n \leq m$ , since  $n$ , given in Eq. (9), denotes the number of 3D objects and  $m$ , given in Eq. (2), denotes the number of 2D projections of those objects after different rigid 3D transformations. That is,  $n$  times the number of rotations.

### 4.2. Optimization for CGPA

In order to minimize the CGPA functional (Eq. (9)):

$$\min_{\mathbf{M}, \mathbf{A}_1, \dots, \mathbf{A}_n} E_{CGPA}(\mathbf{M}, \mathbf{A}_1, \dots, \mathbf{A}_n), \tag{11}$$

we propose an algorithm based on the closed-form solution of two optimization subproblems.

Note that, for simplicity, we omit the variable  $\omega$  when it is understood, we use  $\mathbf{A}$  to denote  $[\mathbf{A}_1^T, \dots, \mathbf{A}_m^T]^T$ , and we omit the superscript 3 from  $\mathbf{D}_i^{(3)}$ .

Unlike standard GPA, in the present formulation,  $\mathbf{A}_i : \Omega \rightarrow \mathbb{R}^{3 \times 3}$  are functions and not parameters. Moreover, it is worth noticing that the dependence of  $E_{CGPA}$  on the functions  $\mathbf{A}_i$  is non-linear. This makes the minimization of  $E_{CGPA}$ , Eq. (11), a non-linear variational problem. Although the existence of a solution  $(\mathbf{M}^*, \mathbf{A}_1^*, \dots, \mathbf{A}_n^*)$  to the problem in Eq. (11) is guaranteed from a theoretical point of view, it is not easy to find its explicit expression (see Section 3.1). For this reason, we propose the following minimization algorithm to find a stationary point. First, we set an initial value  $\mathbf{M} = \mathbf{M}^0$  and we optimize over the functions  $\mathbf{A}_1, \dots, \mathbf{A}_n$ , obtaining a close solution for  $[\mathbf{A}_1^*, \dots, \mathbf{A}_n^*]$ . In the next step, we minimize over  $\mathbf{M}$  the functional  $\mathbf{M} \rightarrow E_{CGPA}(\mathbf{M}, \mathbf{A}_1^*, \dots, \mathbf{A}_n^*)$ . This two step algorithm is detailed below.

Step 1: Optimizing  $E_{CGPA}$  over the functions  $\mathbf{A}_i$ , i.e.

$$\min_{\mathbf{A}_i} E_{CGPA}(\mathbf{M}, \mathbf{A}),$$

can be solved using the following equation:  $\nabla_{\mathbf{A}_i} F(\mathbf{M}, \mathbf{A}_i) = \mathbf{0}$ , where  $\nabla_{\mathbf{A}_i}$  is the gradient operator with respect to the unknown parameters of the matrix  $\mathbf{A}_i$ .

First, let us rewrite  $F(\mathbf{M}, \mathbf{A}_i)$  with the following equivalent expression:

$$F(\mathbf{M}, \mathbf{A}_i) = \text{Tr}((\mathbf{PRD}_i)^T (\mathbf{PRD}_i)) + \text{Tr}((\mathbf{A}_i \mathbf{M})^T (\mathbf{A}_i \mathbf{M})) - 2 \text{Tr}((\mathbf{PRD}_i)^T \mathbf{A}_i \mathbf{M})$$

Then

$$\begin{aligned} \nabla_{\mathbf{A}_i} F(\mathbf{M}, \mathbf{A}_i) &= \mathbf{A} \mathbf{M} \mathbf{M}^T + \mathbf{A} \mathbf{M} \mathbf{M}^T - 2(\mathbf{PRD}_i) \mathbf{M}^T \\ &= 2\mathbf{A} \mathbf{M} \mathbf{M}^T - 2(\mathbf{PRD}_i) \mathbf{M}^T = \mathbf{0} \end{aligned}$$

Finally, the solution of these equations is

$$\mathbf{A}_i^*(\omega) = \mathbf{PR}(\omega) \mathbf{D}_i \mathbf{M}^T (\mathbf{M} \mathbf{M}^T)^{-1}.$$

Step 2: To optimize  $E_{CGPA}$  over  $\mathbf{M}$ , i.e.,  $\min_{\mathbf{M}} E_{CGPA}(\mathbf{M}, \mathbf{A})$ , the necessary conditions are:  $\nabla_{\mathbf{M}} E_{CGPA}(\mathbf{M}, \mathbf{A}) = \mathbf{0}$ .

Given that  $\mathbf{M}$  and  $\mathbf{D}_i$  do not depend on the rotation, the functional can be rewritten involving three definite integrals,  $\mathbf{I}_i \in \mathbb{R}^{3 \times 3}$ ,  $\mathbf{J}_i \in \mathbb{R}^{2 \times 2}$  and  $\mathbf{K}_i \in \mathbb{R}^{3 \times 2}$  as follows:

$$E_{CGPA}(\mathbf{M}, \mathbf{A}) = \sum_{i=1}^n \text{Tr} \left[ \mathbf{D}_i^T \left( \underbrace{\int_{\Omega} (\mathbf{P})^T (\mathbf{PR}) d\omega}_{\mathbf{I}_i} \right) \mathbf{D}_i + \mathbf{M}^T \left( \underbrace{\int_{\Omega} \mathbf{A}_i^T \mathbf{A}_i d\omega}_{\mathbf{J}_i} \right) \mathbf{M} - 2 \mathbf{D}_i^T \left( \underbrace{\int_{\Omega} (\mathbf{PR})^T \mathbf{A}_i d\omega}_{\mathbf{K}_i} \right) \mathbf{M} \right].$$

And, we write

$$\nabla_{\mathbf{M}} E_{CGPA}(\mathbf{M}, \mathbf{A}) = \frac{\partial}{\partial \mathbf{M}} \left( \sum_{i=1}^n \text{Tr}[\mathbf{D}_i^T \mathbf{I}_i \mathbf{D}_i + \mathbf{M}^T \mathbf{J}_i \mathbf{M} - 2 \mathbf{D}_i^T \mathbf{K}_i \mathbf{M}] \right).$$

The first term of the functional does not depend on  $\omega$ , therefore

$$\begin{aligned} \nabla_{\mathbf{M}} E_{CGPA}(\mathbf{M}, \mathbf{A}) &= \frac{\partial}{\partial \mathbf{M}} \left( \sum_{i=1}^n \text{Tr}[\mathbf{M}^T \mathbf{J}_i \mathbf{M}] \right) - 2 \frac{\partial}{\partial \mathbf{M}} \left( \sum_{i=1}^n \text{Tr}[\mathbf{D}_i^T \mathbf{K}_i \mathbf{M}] \right) \\ &= \frac{\partial}{\partial \mathbf{M}} \left( \text{Tr} \left[ \mathbf{M}^T \left( \sum_{i=1}^n \mathbf{J}_i \right) \mathbf{M} \right] \right) - 2 \frac{\partial}{\partial \mathbf{M}} \left( \text{Tr} \left[ \mathbf{D}_i^T \left( \sum_{i=1}^n \mathbf{K}_i \right) \mathbf{M} \right] \right) \\ &= \left( \sum_{i=1}^n \mathbf{J}_i \right) \mathbf{M} + \left( \sum_{i=1}^n \mathbf{J}_i \right)^T \mathbf{M} - 2 \left( \sum_{i=1}^n \mathbf{K}_i \right)^T \mathbf{D}_i \\ &= 2 \left( \sum_{i=1}^n \mathbf{J}_i \right) \mathbf{M} - 2 \left( \sum_{i=1}^n \mathbf{K}_i \right)^T \mathbf{D}_i = \mathbf{0}. \end{aligned}$$

Finally, the solution for  $\mathbf{M}$  can be expressed as follows:

$$\mathbf{M} = \left( \sum_{i=1}^n \mathbf{J}_i \right)^{-1} \left( \sum_{i=1}^n (\mathbf{K}_i)^T \mathbf{D}_i \right).$$

It depends on the integrals  $\mathbf{J}_i$  and  $\mathbf{K}_i$ . We can rewrite the integral form using the solution of the previous step as follows:

$$\begin{aligned} \mathbf{J}_i &= \int_{\Omega} \mathbf{A}_i^T \mathbf{A}_i d\omega = \int_{\Omega} (\mathbf{PRD}_i \mathbf{M}^T \mathbf{L})^T (\mathbf{PRD}_i \mathbf{M}^T \mathbf{L}) d\omega \\ &= \mathbf{L}^T \mathbf{M} \mathbf{D}_i^T \left( \int_{\Omega} (\mathbf{PR})^T \mathbf{PR} d\omega \right) \mathbf{D}_i \mathbf{M}^T \mathbf{L} \end{aligned}$$

$$= \mathbf{L}^T \mathbf{M} \mathbf{D}_i^T \left( \int_{\Omega} \mathbf{X} d\omega \right) \mathbf{D}_i \mathbf{M}^T \mathbf{L},$$

where  $\mathbf{L} = (\mathbf{M} \mathbf{M}^T)^{-1}$  and  $\mathbf{X} = (\mathbf{PR})^T \mathbf{PR}$ . Equivalently

$$\mathbf{K}_i = \int_{\Omega} (\mathbf{PR})^T \mathbf{A}_i d\omega = \left( \int_{\Omega} \mathbf{X} d\omega \right) \mathbf{D}_i \mathbf{M}^T \mathbf{L},$$

In order to compute the value of these integrals we only need to solve the definite integral for  $\mathbf{X}$ . For instance, considering,  $\Omega = \{(\alpha, \beta, \gamma) \in \mathbb{R}^3; |\alpha| \leq \pi/2, |\beta| \leq \pi/2, |\gamma| \leq \pi/2\}$ , we obtain

$$\int_{\Omega} \mathbf{X} d\omega = \begin{pmatrix} \frac{\pi^2}{8} + \frac{\pi^3}{16} & 0 & 0 \\ 0 & \frac{\pi}{8} + \frac{3\pi^3}{32} & 0 \\ 0 & 0 & \frac{-\pi}{8} + \frac{\pi^2}{8} + \frac{3\pi^2}{32} \end{pmatrix}.$$

The special orthogonal group of rotations in 3D space  $SO(3)$  is smooth except for a polar coordinate singularity along an angle of zero [35]. In order to avoid Euler singularities, we use Fubini's theorem. We divide the domains containing zero into disconnected intervals, and we compute the joint integral as iterated integrals. For instance, in order to compute the integral in the domain defined above, i.e.

$$\int_{\Omega} \mathbf{X} d\omega = \int_{-\pi/2}^{\pi/2} \int_{-\pi/2}^{\pi/2} \int_{-\pi/2}^{\pi/2} \mathbf{X} d\omega,$$

we compute the integrals

$$\mathbf{Z}^1 = \int_{-\pi/2}^0 \mathbf{X} d\phi + \int_0^{\pi/2} \mathbf{X} d\phi$$

$$\mathbf{Z}^2 = \int_{-\pi/2}^0 h \sin(\beta) \mathbf{Z}^1 d\beta + \int_0^{\pi/2} h \sin(\beta) \mathbf{Z}^1 d\beta,$$

where  $h$  is the Haar measure for the Euler angle interval. And finally

$$\int_{\Omega} \mathbf{X} d\omega = \int_{-\pi/2}^0 \mathbf{Z}^2 d\psi + \int_0^{\pi/2} \mathbf{Z}^2 d\psi$$

Step 3: In the last step, convergence is measured by the condition  $\|\mathbf{M}^j - \mathbf{M}^{j-1}\|_F < \epsilon_{CGPA}$ , where  $j$  indicates the  $j$ th iteration, and  $\epsilon_{CGPA}$  is a threshold. If convergence is not achieved, a maximum number of iterations is considered.

## 5. Experimentation

This section describes the experimental validation that compares the performance of CGPA to standard discrete PA methods.

### 5.1. Data

We used the following databases in our experiments.

- *Stanford 3D scanning repository*: the data consist of dense 3D meshes of real objects captured with 3D scanners. We considered the ‘‘Stanford Bunny’’ model from the Stanford 3D Scanning Repository.<sup>2</sup> Fig. 4 (left) shows the original 3D shape model. The object contains 453 landmark points; however, for visualization purposes only frontal landmark points are displayed in the graphical results with a triangulated mesh.
- *Athena and Venus models*: we considered two 3D models: Athena and Venus.<sup>3</sup> The shape of these models is more

<sup>2</sup> <http://graphics.stanford.edu/data/3Dscanrep/>

<sup>3</sup> <http://graphics.im.ntu.edu.tw/robin/courses/gm05/model/>

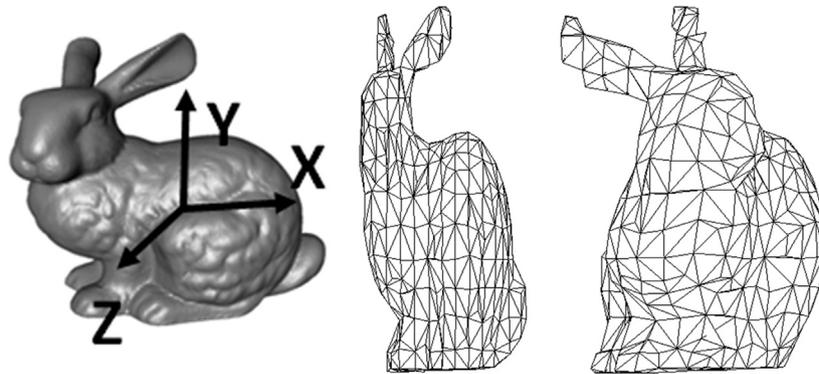


Fig. 4. The bunny 3D shape (left), and reference shapes computed in the qualitative experiment using GPA (middle) and CGPA (right).



Fig. 5. Face samples of the training set: 3D faces (first row) and their projections in 2D space (second row).

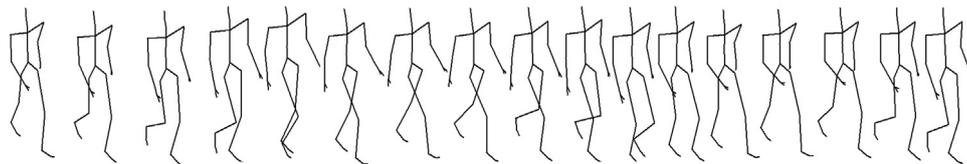


Fig. 6. Samples of the person walking sequence in 3D.

complex than those in the previous databases. Fig. 7 shows the 3D models. The number of landmark points for Athena and Venus is 560 and 819, respectively. The landmarks are connected with lines for better visualization.

- *Multi-PIE database*: we used the 3D face models built from images from the Multi-PIE database [36] using structure from motion [37]. Three-dimensional faces of 30 subjects each adopting three different expressions were considered: neutral, smile and scream. Fig. 5 represents five 3D faces (top row) belonging to the data set and their 2D projections (bottom row). The number of landmark points is 66. The landmarks are connected with lines for better visualization.
- *MoCap database*: the CMU Graphics Lab Motion Capture database [38] contains sequences of several subjects performing different activities captured with a Vicon Motion Capture system. We considered one of the motion sequences: a person walking sequence (#0201). This sequence was acquired at a rate of 30 frames per second. Fig. 6 displays samples of the 3D person walking sequence. The number of landmark points is 343. The landmarks are connected with lines for better visualization.

## 5.2. Methods and parameters

We compared CGPA with the discrete approach, GPA. To learn the 2D reference shape model, CGPA uses 3D data for training, and

GPA uses 2D projections of the 3D training set. Hence, the training set for CGPA has size  $n \times 3 \times l$ , where  $n$  is the number of 3D samples (see Eq. (9)) and  $l$  the number of landmark points; whereas GPA uses a training set of size  $m \times 2 \times l$ , where  $m$  is the number of the 2D samples (projections from a 3D model) considered, see Eq. (2). Both continuous and discrete models are used to approximate new 2D projections of 3D shapes under different configurations (e.g., view points, non-rigid configurations). The performance of the two methods is evaluated qualitatively and quantitatively by measuring the reconstruction error of unseen 2D test samples. Specifically, we computed the mean squared error between the unseen shape and the reconstructed shape, after model fitting.

For GPA we used the standard algorithm [11], and we have described it in Appendix A for completeness. To fit the CGPA, we used the optimization algorithm presented in Section 4.2. The tolerance parameter,  $\epsilon_{GPA}$ , for computing the reference shape using the GPA iterative algorithm was set to the machine precision. The convergence threshold,  $\epsilon_{CGPA}$ , for computing the model in the optimization algorithm, was also set to the machine precision.

We performed three sets of experiments to validate CGPA:

- *Qualitative experiment with Stanford Bunny Model* (Section 5.3): the first experiment dealt with learning the continuous 2D reference shape of a 3D object under 3D rotations. The objective of this experiment was to illustrate the ability of CGPA to obtain an unbiased mean model when the 3D rotation

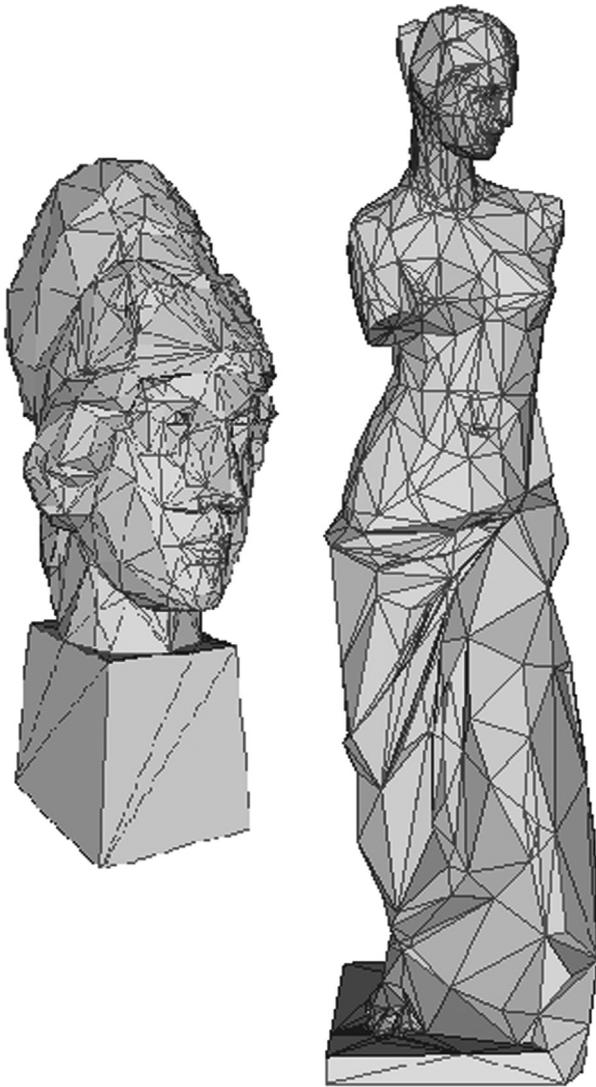


Fig. 7. Samples of the Athena and Venus 3D models.

domain is known. Recall that in the continuous formulation the integration domain simulates all the possible vantage points from which a 3D object can be projected (Eq. (9)).

- *Reference shape model experiment on 3D faces* (Section 5.4): in this experiment, we compared the performance of CGPA and GPA 2D reference shape estimation to approximate a set of 3D faces from the Multi-PIE database [37]. The training set for GPA was obtained by projecting the 3D faces under different viewpoints by sampling  $\Omega$  (angle domain). In the case of CGPA, we integrated continuously over  $\Omega$ . The test set of 90 images was generated by un-sampled angles during the training in the  $\Omega$  domain.
- *Subspace model experiment* (Sections 5.5– 5.7): in this experiment, we used several CGPA reference shapes to approximate the rigid 3D deformations of a set of objects. We compared the reconstruction capability of the multi-reference CGPA with GPA followed by PCA on the aligned shapes [11] (GPA+PCA). The multi-reference CGPA maintains the advantages of the continuous approach.

### 5.3. Qualitative experiment with Stanford Bunny model

The first experiment was qualitative and just for illustrating how CGPA can estimate the reference shape model in a particular

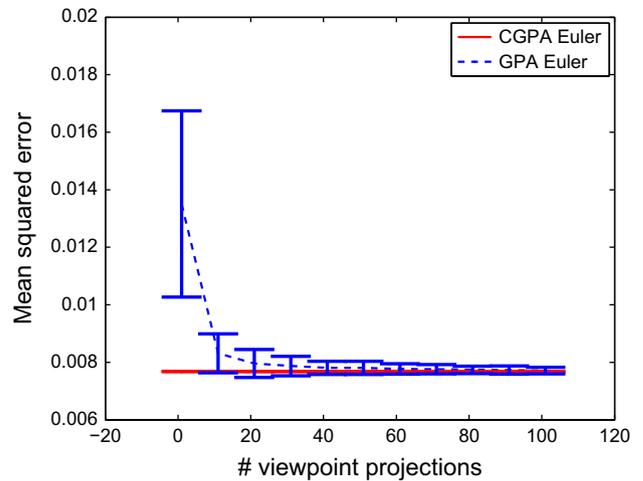


Fig. 8. Reference shape model results for faces. Shown are the reconstruction error for GPA (dashed line) as a function of the number of viewpoint projections in the training set; and reconstruction error for the CGPA reference (solid line) using 3D shapes as training set.

domain. We used the Stanford Bunny from the Stanford 3D Scanning Repository. The continuous method was trained with the single 3D object in the domain:  $\Omega = \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / |\alpha| = 0, 0 \leq \beta \leq \pi/2, |\gamma| = 0\}$ . To make more evident the effect achieved by the CGPA approach, we consider the most degenerated case for the sampling in the discrete training for GPA and we limited the discrete training to one single sample of the rotation angles. In particular, the discrete training set consisted of the sample build by the rotation  $\Omega = \{(\alpha, \beta, \gamma) = (0, 0, 0)\}$ , i.e. the projection of the 3D Bunny on the Z-plane without performing any 3D rotation.

Fig. 4 shows the reference shape models computed using GPA and CGPA. Observe that the model obtained by CGPA is not fixed in the direction corresponding to the frontal viewpoint. We can visually compare the reference model with the frontal projection obtained using the GPA reference model. CGPA can be used to compute a reference shape oriented as desired.

### 5.4. Reference shape model experiment on 3D faces

In this experiment, we used the Multi-PIE database reference. We considered 30 3D faces randomly chosen from the subjects available adopting three different expressions: neutral, smile and scream.

The 2D reference shape for this set of 3D faces was learned under the 3D rotation domain  $\Omega$ . Specifically, we set the rotation domain to:  $\Omega = \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / |\alpha| \leq \pi/4, |\beta| \leq \pi/4, |\gamma| = 0\}$ . See Fig. 9 (left) for an illustration of this domain. We avoided Z-axis rotations to simplify the visualization of the samples. To build the discrete reference shape model, we applied GPA under several viewpoint projections of the 3D shapes, sampling the whole rotation domain  $\Omega$ . For the training set of the CGPA approach, we used the set of 3D faces. We compared the reconstruction capability of the CGPA and GPA reference shapes.

We built the test set using an unseen set of faces rotated through 90 angles randomly selected in the domain  $\Omega$  and projected onto the 2D plane. All the faces were rotated through the same set of angles in  $\Omega$ . We computed the reconstruction error as the mean squared error between the unseen shape and the reconstructed shape (after fitting). In this case, to provide a relative error measure, the error measures are given with respect to the mean eye distance of the test set.

Fig. 8 shows the performance of GPA versus CGPA. The dashed line represents the mean reconstruction error for GPA as a function of the number of training viewpoint projections. The solid line

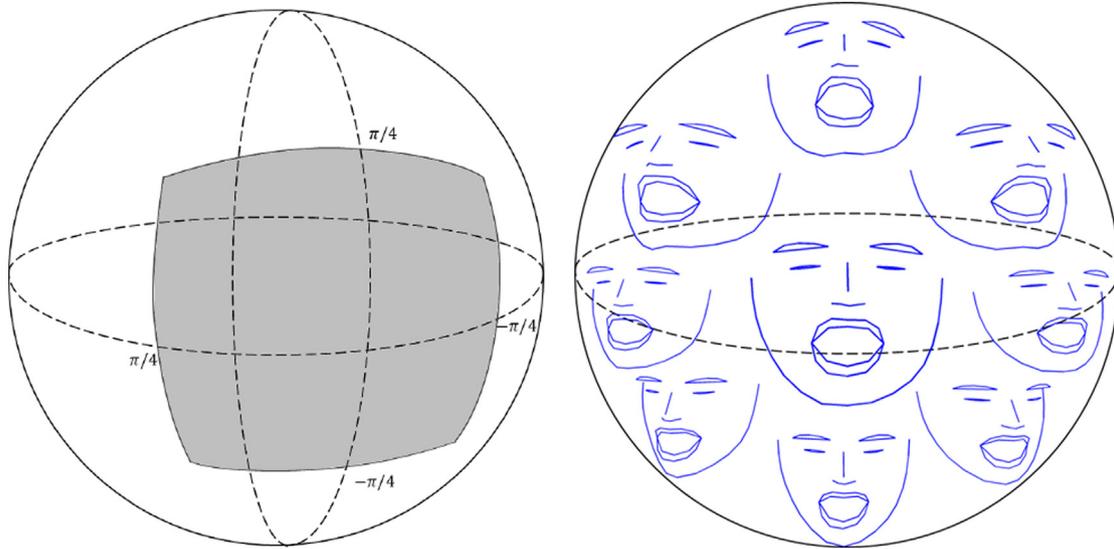


Fig. 9. Left: illustration of the rotation domain  $\Omega$ . Right: the nine reference shapes computed by CGPA in the subspace model experiment with faces.

corresponds to the reconstruction error for CGPA. The training was repeated 30 times, and the standard deviation for each experiment is marked by the horizontal lines. Note that the mean reconstruction error for GPA decreases when the size of the training set increases. Given a large enough number of projections, GPA converges to CGPA, which provides a low error and negligible variance throughout the whole experiment. In this experiment, with 30 faces, GPA required around 80 rotations for each sample face to achieve similar results to CGPA. Thus, GPA needs 80 times more storage space than CGPA and the computational cost of GPA increases by  $80 \cdot 30$  operations.

### 5.5. Subspace model experiment on 3D faces

In this experiment, we have a similar set up to the previous experiments. We also considered 30 3D faces chosen randomly from the available subjects under three different expressions: neutral, smile and scream.

We set the rotation domain to:  $\Omega = \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / |\alpha| \leq \pi/4, |\beta| \leq \pi/4, |\gamma| = 0\}$ . We also avoided Z-axis rotations to simplify the visualization. To learn the discrete subspace model, we first randomly sampled the whole rotation domain,  $\Omega$ , to build the training set of several viewpoint projections of the 3D shapes. We applied GPA to extract the discrete reference shape based on this training set. Then, we removed the mean of the training set using the reference shape. Finally, we performed PCA and we chose the first eight eigenvectors as the basis elements of the subspace. For the continuous model, we estimated nine reference shapes using CGPA. The nine different models were trained with the same 3D shapes, only changing the domain subintervals

$$\begin{aligned} \Omega_1 &= \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / -\pi/4 \leq \alpha \leq 0, \beta = 0, \gamma = 0\}, \\ \Omega_2 &= \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / 0 \leq \alpha \leq \pi/4, \beta = 0, \gamma = 0\}, \\ \Omega_3 &= \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / \alpha = 0, -\pi/4 \leq \beta \leq 0, \gamma = 0\}, \\ \Omega_4 &= \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / \alpha = 0, 0 \leq \beta \leq \pi/4, \gamma = 0\}, \\ \Omega_5 &= \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / -\pi/4 \leq \alpha \leq 0, -\pi/4 \leq \beta \leq 0, \gamma = 0\}, \\ \Omega_6 &= \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / -\pi/4 \leq \alpha \leq 0, 0 \leq \beta \leq \pi/4, \gamma = 0\}, \\ \Omega_7 &= \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / 0 \leq \alpha \leq \pi/4, -\pi/4 \leq \beta \leq 0, \gamma = 0\}, \\ \Omega_8 &= \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / 0 \leq \alpha \leq \pi/4, 0 \leq \beta \leq \pi/4, \gamma = 0\}, \\ \Omega_9 &= \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / |\alpha| \leq \pi/4, |\beta| \leq \pi/4, \gamma = 0\}. \end{aligned}$$

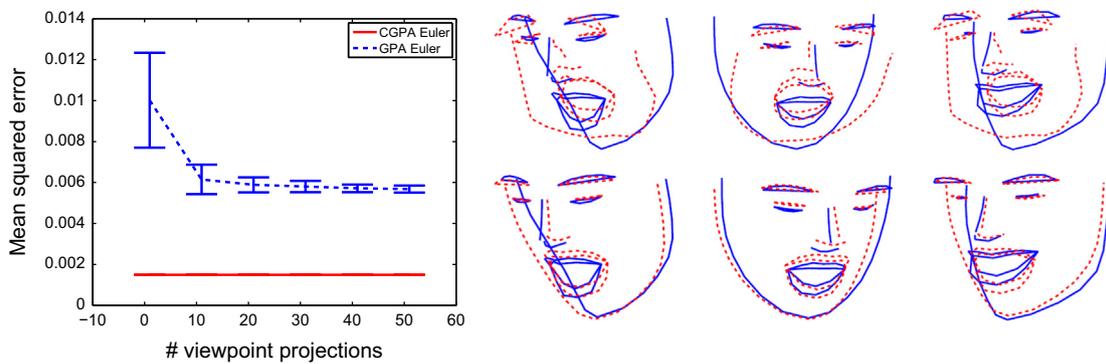
These subintervals were chosen to cover the transformations corresponding to different viewpoint projections of the object. In Fig. 9 (right), we show the nine CGPA reference shapes computed for the nine subintervals.

We built the test set with unseen shapes randomly rotated through 90 angles in  $\Omega$ . All the faces were rotated through the same set of angles in  $\Omega$ . We computed the reconstruction error as the mean squared error between the test shape and the reconstructed shape (after fitting). The error measures are given with respect to the mean eye distance of the test set. Fig. 10 (left) shows the performance of the reference shapes obtained with CGPA versus GPA+PCA. The dashed line represents the mean reconstruction error for GPA+PCA as a function of the number of viewpoint projections in the discrete training set. The solid line shows the reconstruction error for CGPA. The training was repeated 30 times and the standard deviation for each experiment is marked by the horizontal lines. Note that the mean reconstruction error for GPA decreases when the size of the training set increases. In this experiment, CGPA produces competitive results compared to the method GPA+PCA. Therefore, if discrete sampling is to be avoided, CGPA represents an alternative to the standard GPA+PCA method. Qualitative results in Fig. 10 (right) show how CGPA and GPA+PCA perform with the Multi-PIE database.

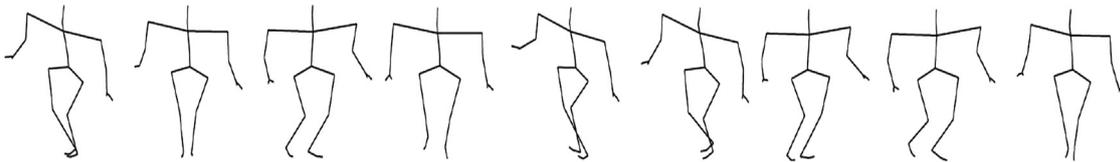
We could have used GPA to build nine discrete reference shapes and compare this discrete subspace model with the continuous subspace model; however, this option was ruled out, since the previous experiment was focused on this comparison.

### 5.6. Subspace model experiment with MoCap

In this experiment, we used the MoCap database. We considered 3D skeletons from 30 frames of the person walking sequence. We rotated them in the following rotation domain:  $\Omega = \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / |\alpha| \leq \pi/4, |\beta| \leq \pi/4, |\gamma| = 0\}$ . As before, we avoided Z-axis rotations to simplify the visualization. To build the discrete subspace model, we first randomly sampled the whole rotation domain  $\Omega$  to build the training set of several viewpoint projections of the 3D shapes. We applied GPA to extract the discrete reference shape from this training set. After aligning to the reference shape, we performed PCA and we chose the first eight eigenvectors as the basis elements of the subspace. For the continuous model, we estimated nine reference shapes using CGPA for the same nine



**Fig. 10.** Subspace model results for faces. Left: reconstruction error for GPA+PCA (dashed line) as a function of the number of viewpoint projections in the training set; and reconstruction error for CGPA reference shapes (solid line) using 3D shapes as the training set. Right: qualitative results of the subspace model experiment with faces. Reconstruction performance using GPA+PCA (top row) and CGPA reference shapes (bottom row).



**Fig. 11.** The nine reference shapes computed by CGPA in the subspace model experiment with motion capture.

subintervals as in the previous experiment. Fig. 11 shows the nine CGPA reference shapes obtained.

We built the test set with unseen shapes randomly rotated through 90 angles in  $\Omega$ . All the skeletons considered were rotated through the same set of selected angles in  $\Omega$ . We computed the reconstruction error as the mean squared error between the test shape and the reconstructed shape (after fitting).

Fig. 12 (left) shows the performance of GPA+PCA versus the reference shapes obtained with CGPA. The dashed line represents the mean reconstruction error for GPA+PCA as a function of the number of training viewpoints. All the 3D skeletons considered were rotated to the same number of viewpoints in the  $\Omega$  domain. The solid line shows the reconstruction error for CGPA. Training was repeated 30 times and the standard deviation for each experiment is indicated by the horizontal lines. As in the previous experiment, the mean reconstruction error for GPA decreases when the size of the 2D training set increases. In this experiment too CGPA produces competitive results compared to the standard GPA+PCA method, and discrete sampling is avoided. The qualitative results in Fig. 12 (right) show the performance of CGPA and GPA+PCA in the person walking sequence from the MoCap database.

### 5.7. Subspace model experiment with Athena and Venus

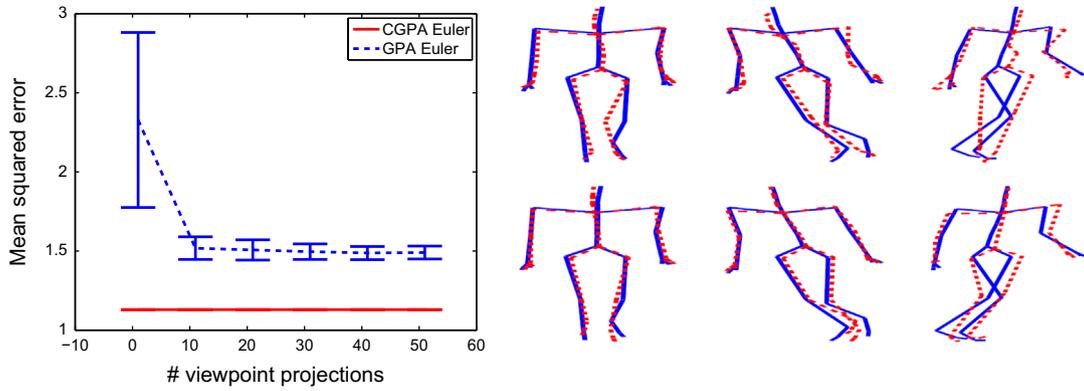
In this experiment, we used the Athena and Venus models. We rotated them in the following rotation domain:  $\Omega = \{(\alpha, \beta, \gamma) \in \mathbb{R}^3 / |\alpha| \leq \pi/4, |\beta| \leq \pi/4, |\gamma| = 0\}$ . As before, we avoided Z-axis rotations to simplify visualization. To build the discrete subspace model, we first randomly sampled the whole rotation domain  $\Omega$  to build the training set of several viewpoint projections of the 3D shapes. We applied GPA to extract the discrete reference shape from this training set. Then, we performed PCA and we chose the first eight eigenvectors as the basis elements of the subspace. For the continuous model, we estimated nine reference shapes using CGPA for the same nine subintervals as in the previous experiment. In Figs. 13 and 14, we show the nine CGPA reference shapes for the Athena and Venus model, respectively.

We built the test set with unseen shapes randomly rotated through 90 angles in  $\Omega$ . We computed the reconstruction error as the mean squared error between the test shape and the reconstructed shape (after fitting).

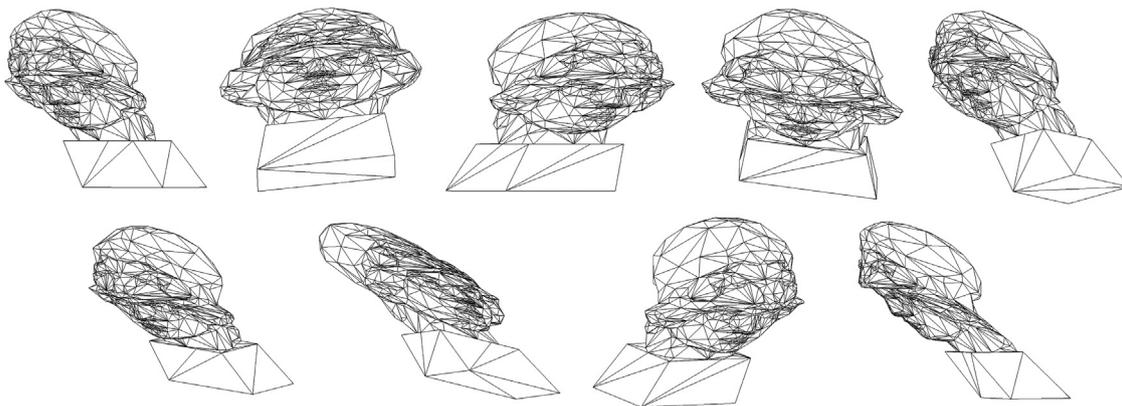
Figs. 15 and 16 show the performance of GPA+PCA versus the reference shapes obtained with CGPA for the Athena and Venus models, respectively. The dashed line represents the mean reconstruction error for GPA+PCA as a function of the number of training viewpoints and the solid line indicates the reconstruction error for CGPA. Training was repeated 30 times and the standard deviation for each experiment is indicated by the horizontal lines. The mean reconstruction error is normalized by an estimation of the mean value of the inpoint size of these shape models. As in the previous experiments, the mean reconstruction error for GPA decreases when the size of the 2D training set increases. In this experiment too CGPA produces competitive results compared to the standard GPA+PCA method, and discrete sampling is avoided. Qualitative results in the right-hand images of Figs. 15 and 16 show the performance of CGPA and GPA+PCA via three test Athena and Venus shapes. Surface colors represent the mean squared error; brighter colors represent greater errors.

## 6. Conclusions

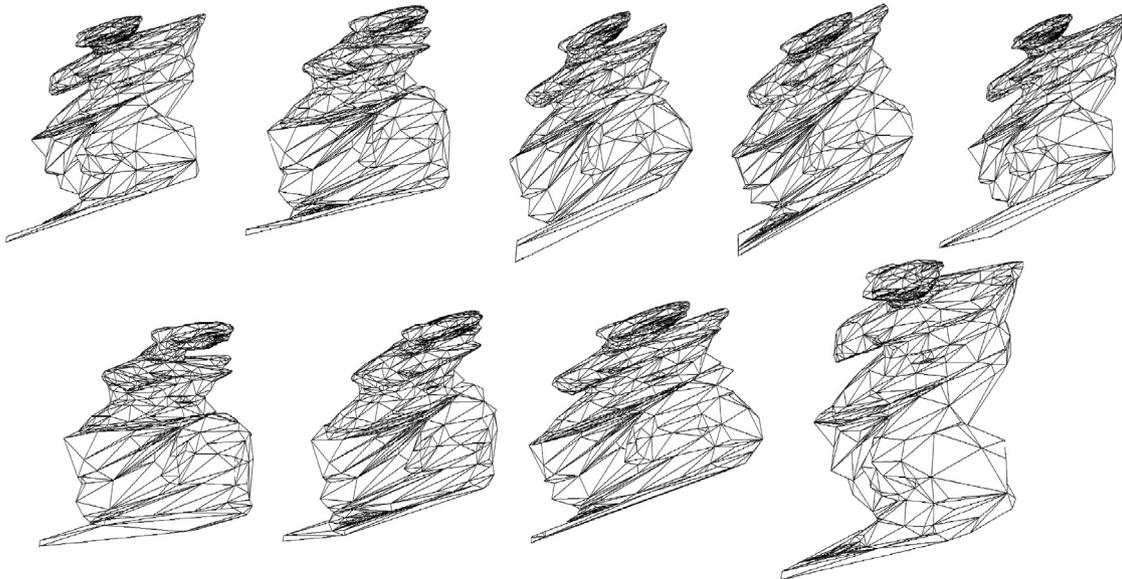
In this work, we have proposed continuous generalized Procrustes analysis, a continuous alternative to GPA, as a method for learning 2D shape models from 3D objects. CGPA has three main advantages over GPA: (i) it does not need to generate 2D samples, (ii) unbiased 2D models are constructed, and (iii) the memory requirements for CGPA are only those of the storage of the 3D objects and the reference shape(s). Moreover, we have reviewed the problems of the uniform sampling of 3D transformations of an object by different parameterizations: Euler angles and quaternions. Finally, we have tested the construction of the CGPA reference shape model experimentally and compared it to GPA for 2D training sets of different sizes. We have also observed that the GPA reference model converges to the CGPA



**Fig. 12.** Subspace model results for MoCap. Left: reconstruction error for GPA+PCA (dashed line) as a function of the number of viewpoint projections in the training set; and reconstruction error for CGPA reference shapes (solid line) using 3D shapes as the training set. Right: qualitative results of the subspace model experiment with MoCap skeletons. Reconstruction performance using GPA+PCA reference shapes (top row) and CGPA (bottom row). Test shape (solid line) and reconstruction shape (dashed line).



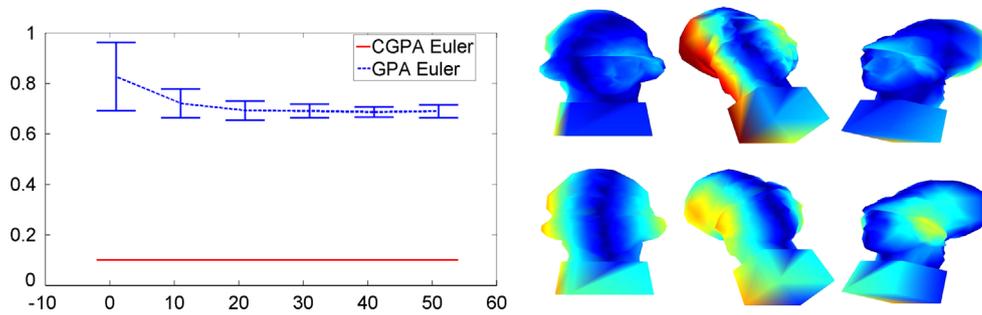
**Fig. 13.** The nine reference shapes computed by CGPA in the subspace model experiment with Athena.



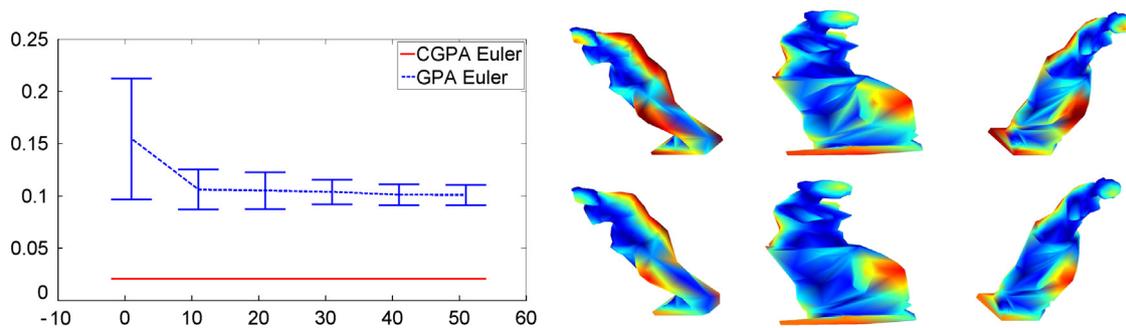
**Fig. 14.** The nine reference shapes computed by CGPA in the subspace model experiment with Venus.

reference model when using a sufficient number of training samples. Moreover, the experiments have shown that the continuous model using several CGPA reference shapes can be an alternative to the classical discrete subspace approach to avoid discrete sampling.

As a future line of research, we plan to incorporate a subspace into the energy functional and to study different optimization methods using optimal discretization of the functional. Moreover, in the case of AAM where only 2D samples are available, the CGPA formulation may incorporate a structure



**Fig. 15.** Subspace model results with the Athena model. Left: reconstruction error for GPA+PCA (dashed line) as a function of the number of viewpoint projections in the training set; and reconstruction error for CGPA reference shapes (solid line) using 3D shapes as the training set. Right: qualitative results of the subspace model experiment with Athena. Reconstruction performance using GPA+PCA reference shapes (top row) and CGPA (bottom row). Mean squared error over the test shape surface. Brighter colors represent greater errors. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)



**Fig. 16.** Subspace model results with Venus model. Left: reconstruction error for GPA+PCA (dashed line) as a function of the number of viewpoint projections in the training set; and reconstruction error for CGPA reference shapes (solid line) using 3D shapes as the training set. Right: qualitative results of the subspace model experiment with Venus. Reconstruction performance using GPA+PCA reference shapes (top row) and CGPA (bottom row). Mean squared error over the test shape surface. Brighter colors represent greater errors. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this article.)

from the motion step to infer and impose consistency on the 3D structure of shapes.

### Conflict of interest

None declared.

### Acknowledgments

This research was supported by the projects TIN2009-14404-C02, CONSOLIDER-INGENIO 354CSD2007-00018, TIN2012-38416-C03-01, FEDER funds, and the Comissionat per a Universitats i Recerca del Departament d'Innovació, Universitats i Empresa de la Generalitat de Catalunya.

### Appendix A. GPA fitting

Given a target shape  $\mathbf{D}_i^{(2)}$  we vectorize it as:  $\mathbf{d}_i = (x_{i1}, y_{i1}, \dots, x_{i\ell}, y_{i\ell})$ . Then we use the subsequent iterative approach to solve the following reference-space error:  $E_{GPA}(T, \mathbf{c}) = \sum_{i=1}^n \|\mathbf{d}_i - T(\mathbf{m} + \mathbf{B}\mathbf{c})\|_2^2$ , where  $\mathbf{m}$  is the vectorized version of the reference shape  $\mathbf{M}$ ,  $\mathbf{B}$  is the basis built using PCA, and  $T$  is a similarity transformation.

1. Initialize coefficients  $\mathbf{c}$  (shape parameters) to zero.
2. Generate the model instance:  $\mathbf{x} = \mathbf{m} + \mathbf{B}\mathbf{c}$ .
3. Find the pose parameters  $T$  which best map  $\mathbf{x}$  to  $\mathbf{d}_i$  using PA.
4. Invert the transformation  $T$  and use it to project  $\mathbf{d}_i$  into the model co-ordinate frame:  $\mathbf{d}'_i = T^{-1}\mathbf{d}_i$
5. Normalize the shape.
6. Update the model parameters to match  $\mathbf{d}'_i$ :  $\mathbf{c} = \mathbf{B}^T(\mathbf{d}'_i - \mathbf{m})$ .

7. Apply constraints on  $\mathbf{c}$ .

8. If the error does not converge (i.e., use the reconstructed shape  $\mathbf{x}' = \mathbf{m} + \mathbf{B}\mathbf{c}$ ,  $E = \|\mathbf{x} - \mathbf{x}'\|_2^2 < \epsilon_{GPA}$ ), return to step 2.

### References

- [1] I.L. Dryden, K.V. Mardia, *Statistical Shape Analysis*, Wiley, Chichester, 1998.
- [2] J.C. Gower, G.B. Dijksterhuis, *Procrustes Problems*, Oxford University Press, 2004.
- [3] C. Goodall, Procrustes methods in the statistical analysis of shape, *Journal of the Royal Statistical Society. Series B* 53 (2) (1991) 285–339.
- [4] S. Ullman, R. Basri, Recognition by linear combinations of models, *Transactions on Pattern Analysis and Machine Intelligence* 13 (10) (1991) 992–1006.
- [5] M.J. Jones, T. Poggio, Multidimensional morphable models, in: *ICCV*, 1998.
- [6] F. De la Torre, M. Nguyen, Parameterized kernel principal component analysis: theory and applications to supervised and unsupervised image alignment, in: *CVPR*, 2008.
- [7] T. Cootes, G. Edwards, C. Taylor, Active appearance models, *Transactions on Pattern Analysis and Machine Intelligence* 23 (6) (2001) 681–685.
- [8] S. Osher, N. Paragios, *Geometric Level Set Methods in Imaging, Vision, and Graphics*, Springer-Verlag, 2003.
- [9] D. Mumford, J. Shah, Optimal approximations by piecewise smooth functions and associated variational problems, *Communications on Pure and Applied Mathematics* 42 (5) (1989) 577–685.
- [10] C. Tomasi, T. Kanade, Shape and motion from image streams under orthography: a factorization method, *International Journal of Computer Vision* 9 (2) (1992) 137–154.
- [11] T.F. Cootes, C.J. Taylor, *Statistical Models of Appearance for Computer Vision*, Technical Report, University of Manchester, M13 9PT, 2001.
- [12] J. Kuffner, Effective sampling and distance metrics for 3d rigid body path planning, in: *ICRA*, 2004.
- [13] L. Igual, F. De la Torre, Continuous Procrustes analysis to learn 2d shape models from 3d objects, in: *NORDIA Workshops, in Conjunction with CVPR*, 2010.
- [14] A. Bartoli, D. Pizarro, M. Loog, Stratified generalized procrustes analysis, in: *BMVC*, 2010.

- [15] A.J. Yezzi, S. Soatto, Deformation: deforming motion shape average and the joint registration and approximation of structures in images, *International Journal of Computer Vision* 53 (2) (2003) 153–167.
- [16] D. Pizarro, A. Bartoli, Global optimization for optimal generalized procrustes analysis, in: CVPR, 2011.
- [17] F. De la Torre, M.J. Black, Robust parameterized component analysis: theory and applications to 2d facial appearance models, *Computer Vision and Image Understanding* 91 (1) (2003) 53–71.
- [18] E.G. Learned-Miller, Data driven image models through continuous joint alignment, *Transactions on Pattern Analysis and Machine Intelligence* 28 (2) (2006) 236–250.
- [19] S. Baker, I. Matthews, J. Schneider, Automatic construction of active appearance models as an image coding problem, *Transactions on Pattern Analysis and Machine Intelligence* 26 (10) (2004) 1380–1384.
- [20] B.J. Frey, N. Jojic, Transformation-invariant clustering using the EM algorithm, *Transactions on Pattern Analysis and Machine Intelligence* 25 (1) (2003) 1–17.
- [21] I. Kookinos, A. Yuille, Unsupervised learning of object deformation models, in: ICCV, 2007.
- [22] J.O. Ramsay, B.W. Silverman, *Functional Data Analysis*, Springer-Verlag, 1997.
- [23] D. Freedman, *Statistical Models: Theory and Practice*, Cambridge University Press, 2005.
- [24] D. Ormoneit, M.J. Black, T. Hastie, H. Kjellström, Representing cyclic human motion using functional analysis, *Image Vision Computing* 23 (14) (2005) 1264–1276.
- [25] A. Levin, A. Shashua, Principal component analysis over continuous subspaces and intersection of half-spaces, in: ECCV, 2002.
- [26] B. Dacorogna, *Direct Methods in the Calculus of Variations*, Springer-Verlag, 1989.
- [27] A. Franco, D. Maio, D. Maltoni, 2D face recognition based on supervised subspace learning from 3D models, *Pattern Recognition* 41 (12) (2008) 3822–3833.
- [28] I. Fonseca, G. Leoni, *Modern Methods in the Calculus of Variations:  $L^p$  Spaces*, Springer-Verlag, 2007.
- [29] M. Spivak, *Calculus*, Cambridge University Press, 2006.
- [31] E.Y. Chao, Justification of triaxial goniometer for the measurement of joint rotation, *Journal of Biomechanics* 13 (12) (1980) 989–1006.
- [32] K. Shoemake, Uniform random rotations, in: D. Kirk (Ed.), *Graphics Gems III*, Academic Press Professional, 1992, pp. 124–132.
- [33] S. Hamilton, *Lectures on Quaternions*, Hodges and Smith, 1853.
- [34] M.A. Naimark, *Linear Representation of the Lorentz Group*, Macmillan, 1964.
- [35] K. Shoemake, Animating rotation with quaternion curves, in: SIGGRAPH, 1985.
- [36] R. Gross, I. Matthews, J.F. Cohn, T. Kanade, S. Baker, Multi-PIE, *Image and Vision Computing* 28 (5) (2010) 807–813.
- [37] J. Gonzalez-Mora, F. De la Torre, N. Guil, E.L. Zapata, Learning a generic 3d face model from 2d image databases using incremental structure from motion, *Image and Vision Computing* 28 (7) (2010) 1117–1129.
- [38] Carnegie Mellon Motion Capture Database (<http://mocap.cs.cmu.edu>).

**Laura Igual** received the degree in Mathematics from the University of Valencia in 2000. She obtained her Ph.D. Thesis in 2006 from the University Pompeu Fabra and since then she is a research member at the Computer Vision Center of Barcelona. Since 2009, she is a lecturer at University de Barcelona.

**Xavier Perez-Sala** received the B.Sc. degree in Industrial Electronics (2008) and the M.Sc. degree in Artificial Intelligence (2010) from the Technical University of Catalonia. He is currently pursuing a Ph.D. degree in Artificial Intelligence at the same university and the Fundació Privada Sant Antoni Abat. Since 2012 he is member of the Computer Vision Center of Barcelona.

**Sergio Escalera** received the B.S. and M.S. degrees from the Universitat Autònoma de Barcelona in 2003 and 2005, respectively. He obtained the Ph.D. degree on multi-class visual categorization systems at Computer Vision Center, UAB. Currently, Lecturer of Universitat de Barcelona. Partial time professor at the Universitat Oberta de Catalunya.

**Cecilio Angulo** received the M.Sc. degree in Mathematics from the University of Barcelona (1993) and a Ph.D. in Sciences from the Technical University of Catalonia (2001). He is a associated professor at the same university from 2007. He is coordinator of the master's degree in Automatic Control and Robotics.

**Fernando De la Torre** received his B.Sc. in Telecommunications (1994), M.Sc. (1996), and Ph.D. (2002) in Electronic Engineering from La Salle of Engineering in Ramon Llull University. He was an assistant and associate professor in La Salle (1997, 2000). Since 2005, he is a research assistant professor in the Robotics Institute at Carnegie Mellon University.