Comput. Geom. 90 (2020) 101657

Contents lists available at ScienceDirect

Computational Geometry: Theory and Applications

www.elsevier.com/locate/comgeo

Dihedral deformation and rigidity

Nina Amenta [a,*], Carlos Rojas [b]

[a] University of California, Davis, United States of America
[b] San José State University, United States of America

ARTICLE INFO

Article history:
Available online 12 May 2020

Keywords:
Shape space
Dihedral angles
Deformation
Rigidity

ABSTRACT

We consider defining the embedding of a triangle mesh into $\mathbb{R}^3$, up to translation, rotation, and scale, by its vector of dihedral angles. On the theoretical side, we show that locally the map from realizable vectors of dihedrals to mesh embeddings is one-to-one almost everywhere. On the implementation side, we are interested in using the dihedral parameterization in shape analysis. This demands a way to visualize statistical results, for instance an average shape. To this end, we give a heuristic method for mapping interpolations in dihedral space to interpolations between input mesh embeddings, and we visualize statistical analyses of several families of organic shapes.

1. Introduction

Usually a polygon mesh is represented by its mesh combinatorics and a vector of 3D vertex coordinates, specifying the immersion of the mesh into $\mathbb{R}^3$. A polygon mesh is *rigid* when the only motions of the vertex coordinates for which the faces are not deformed in any way are the rigid motions (rotation and translation). Non-rigid polyhedra do exist [1–3], although they are rare; a non-rigid polyhedron has some *flexing* motion in which the faces move but do not deform. That is, the dihedral angles between faces change continuously while the faces themselves remain rigid. Herman Gluck [4] proved

Theorem 1 *(Gluck). A generic immersion of any mesh topology homeomorphic to the sphere is rigid.*

By *generic* we mean all vectors of vertex coordinates, except some "degenerate" subset of measure zero. So, for example, if you construct the edge-skeleton of a triangulated surface such as the Stanford bunny, with rigid stick edges held together at flexible joints, it will almost certainly be rigid.

Mesh deformation is an important topic in computer graphics, computer vision and scientific shape analysis. Typically, both edge lengths and dihedrals change during deformation. The change in edge lengths defines a path in the space of discrete metrics. The subset of discrete conformal transformations in the space of discrete metrics has been especially well-studied, e.g. [5–8]. Lam and Pinkall [9] recently proved that infinitesimally rigid polyhedra always allow infinitesimal conformal deformations, and that the space of these deformations can be parameterized by a discrete curvature function defined on the vertices. But discrete metrics do not correspond to 3D shape in any precise sense: a vector of edge lengths typically has multiple discrete realizations as a rigid mesh, and it might even correspond to a flexible polyhedron.

We are interested in the complementary approach: characterizing deformation by the change in a mesh's vector of dihedral angles. The first mathematical question one might ask is whether there are motions in which all the dihedral

* Corresponding author.
E-mail addresses: amenta@cs.ucdavis.edu (N. Amenta), carlos.rojas@sjsu.edu (C. Rojas).

https://doi.org/10.1016/j.comgeo.2020.101657
0925-7721/

angles stay the same, but the edge lengths change. This is trivially possible; consider a cube deforming into an arbitrary box. Notice that during such a deformation the inner face angles (plane angles) remain unchanged. So next we ask if there are deformations in which the dihedral angles remain fixed, but the inner angles change; we can call this a *dihedral flex*. We say that a polyhedron which does *not* allow a dihedral flex is *dihedral-rigid*. It is not known if dihedral-flexible polyhedra exist. In the first part of this paper, we prove the following analog of Gluck's theorem:

Theorem 2. *A generic immersion of any triangle mesh homeomorphic to the sphere is dihedral-rigid.*

While interesting as a result in rigidity theory, of course this is only a small step towards a theoretical justification of the idea of representing mesh embeddings by their dihedral vectors. In the second part of the paper, we give some experimental evidence that the dihedral representation is a useful and natural shape space. Let's define the combinatorial structure of the mesh with the graph (V, E), so that the dihedral vector is a point in $\mathbb{R}^{|E|}$. Given an input data set of corresponding meshes, we analyzed them using the normal statistical techniques and software as points in $\mathbb{R}^{|E|}$, using the usual Euclidean metric. Working in a Euclidean space has obvious advantages, such as simplicity and flexibility.

Before we can do this analysis, however, we need to have a set of shapes represented as different embeddings of a common combinatorial mesh. This can be produced from, and is roughly equivalent to, a dense surface correspondence between all of the input specimens. While correspondence problems are famously difficult, there has been terrific recent progress on the problem of computing a common correspondence for a family of similar input shapes, particular through the use of functional maps [10], and recently by using deep learning as well, e.g. [11].

Probably the most common approach to shape analysis uses the usual representation of an embedding by its aligned vertex positions. An input set of vertex correspondences is aligned as well as possible, and then treated as points in $\mathbb{R}^{3|V|}$. This gives a Euclidean shape space, with the root-mean-square distance (also known as Procrustes space). We will compare the two Euclidean shape spaces, dihedral and Procrustes, in Section 6.

To visualize results of a statistical analysis in shape space, such as an average or a PCA vector, we need to be able to construct a mesh embedding for an arbitrary new vector of dihedrals. This problem appears to be very difficult. Instead, we visualize the results of operations such as averaging and PCA using a heuristic method to interpolate embeddings. We use two algorithms, first a heuristic to get an approximate embedding and then an optimization algorithm to reduce the total error of the dihedrals.

Using this approach, we find that lines or planes in dihedral space lead to smooth spaces of natural-looking morphs connecting sometimes very different shapes. We take this as evidence that the dihedral metric is a good measure of what we intuitively mean by shape difference. Specifically, the morphs are smooth and the shapes of parts that differ by a rigid motion interpolate via a rigid motion. In earlier work [12], we found that attempting to optimize the embedding towards interpolated edge lengths, rather than interpolated dihedrals, produced morphs with discontinuities and glitches. We believe that this is related to the fact that, outside of very simple examples, a specific set of edge lengths does not have a single unique embedding. We conjecture that there is at most one embedding per vector of dihedrals.

2. Related work

In 1968, Stoker [13] conjectured that a convex polyhedron is uniquely defined by its combinatorics and dihedral angles, that is, that it is dihedral-rigid. This would be the dihedral version of Cauchy's theorem on the rigidity of convex polyhedra [14]. A fairly simple proof of Stoker's conjecture for triangulated convex polyhedra was given by Pogorelov [15]; we draw on his work as well as that of Gluck. Only recently was a complete proof of Stoker's conjecture provided by Mazzeo and Montcouquiol [16], using much more sophisticated techniques and applying to the interesting case of ideal hyperbolic polyhedra as well. The Maxwell-Cremona correspondence, a classic construction from mechanical statics, relates infinitesimal edge-length flexes to dihedrals in a "lifted" structure; an interesting modern presentation, related to a projective Grassmannian framework, can be found for example in [17]. We are indebted to the excellent online textbook by Pak [18] for much of the background in this area.

In computer graphics, there is an ongoing interest in constructing shape spaces in which geodesic paths correspond to physically natural-looking morphs, which can be used for applications such as artistic modeling, shape exploration, and deformation transfer. These spaces tend to be curved and difficult to deal with, e.g. [19]. A recent series of papers [20–22] explores the curved shape space implied by the elastic energy of a deformation, which is the sum of a membrane term depending on edge lengths and a bending term depending on dihedrals. They prove [20] that it forms a Riemannian manifold, that is, that the bending term and the membrane term cannot both be zero under a non-rigid deformation, which means that there are no elastic-energy flexible polyhedra. Using this energy as a metric, they produce shape averages, principal components and splines in "shell space". Each of these operations in the curved shape space proves to be challenging, both mathematically and computationally.

There is a practically successful line of work [7,23,24] on interpolating mesh embeddings by interpolating both their dihedral angles and their edge lengths, and then doing a least-squares reconstruction to produce an interpolating mesh. These methods cannot realize both the dihedrals and the edge lengths exactly - there are roughly $6|V|$ parameters and $3|V|$ degrees of freedom in the embedding - but they are fairly simple to describe and implement and they produce very nice-looking results. Our first, rough embedding algorithms belongs in this family of techniques.

The space of dihedral angles was proposed as a representation for deformation by Paillé et al. [25], albeit for a tetrahedralized volume. Again, the number of dihedrals in a tetrahedralization is much larger than the dimension of the space of realizable meshes, so in general only an approximation of a given set of dihedrals is possible.

Finally, an interesting experiment in [26] showed that ignoring both edge lengths and dihedrals and just using connectivity to reconstruct shapes can be surprisingly successful.

3. Dihedral rigidity

One's first instinct when considering the possibility of a dihedral flex is to consider the vertex positions p_i as functions $p_i(t)$ of some parameter t. Then we can consider a triangle (p_j, p_i, p_{j+1}) and the derivatives, with respect to t, of its inner angles, e.g. $\beta_{j,i,j+1}$, and edge dihedrals, e.g. $\alpha_{i,j}$. At any time along a traditional edge-length flex, the length derivatives $l'_{i,j} = 0$ at every edge, while at least some of the $\alpha'_{i,j}$ are non-zero. Similarly along any dihedral flex (if such a thing exists!) we expect to find an infinitesimal motion such that all $\alpha'_{i,j} = 0$, while there are inner angles for which the derivatives $\beta'_{j,i,j+1}$ are non-zero. We call a polyhedron which admits such a motion *dihedral infinitesimally non-rigid*.

A polyhedron which is dihedral non-rigid must be dihedral infinitesimally non-rigid. It may well be possible, however, for a polyhedron to be dihedral infinitesimally non-rigid while being rigid; we know that there are many polyhedra which are (length) infinitesimally non-rigid, but actually rigid. Following Gluck, we prove here that a generic immersion of a mesh forms a polyhedron which is dihedral infinitesimally rigid, and hence dihedral rigid.

We need to formulate equations that express the fact that a given mesh configuration allows an infinitesimal dihedral flex. Fortunately, working with derivatives is a good way to linearize a problem!

3.1. Constraint on rotations around a vertex

There is a very nice relationship between the derivatives of the dihedral angles α' and the derivatives of the triangle inner angles β'; Gluck used it in his theorem on length rigidity, and we shall too. We have, going around the one-ring of any vertex p_i,

$$\sum_j \alpha'_{ij}\vec{e}_{ij} + \sum_j \beta'_{j,i,j+1}\vec{n}_{j,i,j+1} = \mathbf{0} \tag{1}$$

where $\vec{n}_{j,i,j+1}$ is the normal to triangle $t_{j,i,j+1}$, and $\vec{e}_{ij} = (p_i - p_j)/||p_i - p_j||$ is the unit vector in the direction of edge e_{ij}. This equation expresses the fact that the instantaneous angular velocities in the one-ring have to change in a coordinated fashion for the one-ring to continue to "hold together".

Since the edge and normal vectors have three coordinates each, this gives us three equations at each vertex, for a total of $3|V|$. Let's call these the vertex equations.

Gluck considered the case in which the change in edge lengths, and hence the inner angle derivatives β', are all zero, so that the edge length infinitesimally non-rigid configurations were those with

$$\sum_j \alpha'_{ij}\vec{e}_{ij} = \mathbf{0}.$$

This system has $3|V|$ equations in $|E| = 3|V| - 6$ variables, for a mesh homeomorphic to a sphere.

We make the opposite assumption, that the dihedral angles α remain unchanged, so we are interested in non-zero solutions to

$$\sum_j \beta'_{j,i,j+1}\vec{n}_{j,i,j+1} = \mathbf{0}.$$

In our case we have $3|V|$ equations in the $6|V| - 12$ variables β'.

3.2. Angular momentum around a vertex

While Equation (1) is well-know in, for instance, physics and robotics, we provide a derivation here for completeness; readers familiar with, or uninterested in, this material can feel free to skip this section.

Let the matrix R_a perform the rotation through the origin with axis $a = (a_x, a_y, a_z)$ by angle α. Let's think of this as a rotation with speed α. At time t, the position of a point p affected by the rotation is $p(t) = R_a(t)p$, where $R_a(t)$ is the rotation through a by angle $t\alpha$. The derivative

$$\frac{dp(t)}{dt} = \frac{dR_a(t)}{dt}p = \omega_a \times R_a(t)p = S_a R_a(t)p$$

where ω_a is the angular velocity vector $(\alpha a_x, \alpha a_y, \alpha a_z)$, that is, the axis a scaled by the speed α of the rotation, and the anti-symmetric matrix

$$S_a = \begin{bmatrix} 0 & -\alpha r_z & \alpha r_y \\ \alpha r_z & 0 & -\alpha r_x \\ -\alpha r_y & -\alpha r_x & 0 \end{bmatrix}$$

is the matrix that performs the cross-product. So the derivative of $R_a(t)$ is $S_a R_a(t)$.

Now let's consider the derivative of a series of rotations. Say

$$R_d = R_a R_b R_c.$$

We get

$$\frac{dR_d}{dt} = \frac{dR_a}{dt} R_b R_c + R_a \frac{dR_b}{dt} R_c + R_a R_b \frac{dR_c}{dt}$$
$$S_d R_d = S_a R_a R_b R_c + R_a S_b R_b R_c + R_a R_b S_c R_c.$$

Applying $S_d R_d$ to point p, and expressing it in terms of the angular momentum vectors, we see that $\omega_d \times R_d p_0$ is

$$\omega_a \times R_a R_b R_c p + (R_a \omega_b) \times R_a R_b R_c p + (R_a R_b \omega_c) \times R_a R_b R_c p.$$

Here we are using the fact that $R(q \times p) = Rq \times Rp$, where R is a rotation matrix. Just looking at the angular momentum vectors, then, we have

$$\omega_d = \omega_a + (R_a \omega_b) + (R_a R_b \omega_c).$$

Recall each vector ω was defined in its own local coordinate system. The rotations in this formula transform them all into a single global coordinate system. Equation (1) is a specific example of this formula, in which the sequence of rotations are those performed by a particle traveling in an infinitesimal circle around vertex p_i of the mesh; the terms have been rearranged to group together the dihedral angles and the inner angles.

3.3. Constraints on inner angles

There are additional constraints on the β' variables which determine the validity of the mesh. One is that the sum of the inner angles of any triangle add up to π. Taking the derivative of this condition is gives us

$$\beta'_i + \beta'_j + \beta'_{j+1} = 0.$$

We call these the face equations.

Finally, the Law of Sines implies the following differential cotangent formula for the triangles around a given one-ring

$$\sum_j (\cot \beta_{i,j,j+1}) \beta'_{i,j,j+1} - (\cot \beta_{i,j+1,j}) \beta'_{i,j+1,j} = 0.$$

We call these the cotangent equations. To see this, let p_i be a vertex, and consider the vertices of its one-ring, p_j, p_{j+1}, etc. Using the Law of Sines, we have

$$\frac{\sin \beta_{i,j,j+1}}{\sin \beta_{i,j+1,j}} = \frac{l_{i,j+1}}{l_{i,j}}.$$

Going around the one-ring,

$$\prod_j \frac{l_{i,j+1}}{l_{i,j}} = 1$$

and so

$$\prod_j \frac{\sin \beta_{i,j,j+1}}{\sin \beta_{i,j+1,j}} = 1.$$

Taking the natural logarithm, we have

$$\sum_j \ln \sin \beta_{i,j,j+1} - \ln \sin \beta_{i,j+1,j} = 0.$$

Next we take the derivative. We have $(\ln x)' = 1/x$ and $(\sin x)' = \cos x$, so we get

$$\sum_j \frac{\cos \beta_{i,j,j+1}}{\sin \beta_{i,j,j+1}} \beta'_{i,j,j+1} - \frac{\cos \beta_{i,j+1,j}}{\sin \beta_{i,j+1,j}} \beta'_{i,j+1,j} = 0$$

which give the cotangent equations.

3.4. Condition for a solution

Together, the vertex equations, face equations and cotangent equations form a system M with $3V + 2V - 4 + V = 6V - 4$ equations in $6V - 12$ variables.

$$M\beta' = \mathbf{0}$$

A mesh is dihedral infinitesimally non-rigid if this system M has some non-zero solution for the βs, that is, if there is an infinitesimal motion of the mesh that leaves the dihedrals fixed but allows the inner angles to flex somehow, while maintaining a valid mesh.

Following Gluck, we observe that there is a non-zero solution for β if and only if the coefficient matrix M has rank less than $6V - 12$. And for this to be true, it must be the case that every $6V - 12 \times 6V - 12$ sub-matrix of M has zero determinant. We can write this condition on the coefficient matrix itself as a system of $\binom{6V-4}{8}$ polynomials in the matrix elements; call this system F. In Gluck's proof, he dealt with a matrix whose coefficients were themselves polynomials in the vertex coordinates of the mesh, and this allowed him to argue that the resulting variety formed a set of measure zero.

In our case, the coefficients are the face normals, the cotangents of the inner angles, and the scalar value one. The normals and cotangents are not polynomials in the vertex coordinates. To get around this, we treat the normals and cotangents as variables themselves; for notational clarity, let's write $c_{j,i,j+1} = \cot \beta_{j,i,j+1}$. The c and n variables are not independent of each other. The normals must all have length one; for $n_{j,i,j+1} = (n_x, n_y, n_z)$, we have

$$n_x^2 + n_y^2 + n_z^2 = 1. \tag{2}$$

In addition, the normal and cotangent variables are conveniently related to each other, and to the vertex coefficients, by the following formula:

$$\begin{aligned} &\big[(p_i - p_j) \times (p_i - p_{j+1})\big] c_{j,i,j+1} = \\ &\big[(p_i - p_j) \cdot (p_i - p_{j+1})\big] n_{j,i,j+1}. \end{aligned} \tag{3}$$

This formula relates the cotangent to the scaling of the cross-product to form the triangle normal. To see this, observe that

$$(p_i - p_j) \times (p_i - p_{j+1}) = ||p_i - p_j||||p_i - p_{j+1}|| \sin \beta_{j,i,j+1} n_{j,i,j+1}.$$

We also know that

$$(p_i - p_j) \cdot (p_i - p_{j+1}) = ||p_i - p_j||||p_i - p_{j+1}|| \cos \beta_{j,i,j+1}.$$

So we can write

$$\begin{aligned} &[(p_i - p_j) \times (p_i - p_{j+1})] \cos \beta_{j,i,j+1} = \\ &[(p_i - p_j) \cdot (p_i - p_{j+1})] \sin \beta_{j,i,j+1} n_{j,i,j+1} \end{aligned}$$

and hence

$$\begin{aligned} &[(p_i - p_j) \times (p_i - p_{j+1})] \cot \beta_{j,i,j+1} = \\ &[(p_i - p_j) \cdot (p_i - p_{j+1})] n_{j,i,j+1}. \end{aligned}$$

Since the cross-product and dot-product are both polynomial functions, this formula is a polynomial as well.

In order for a mesh configuration to be dihedral infinitesimally non-rigid, we need Equations (2) and (3) to be true for every angle, as well as for all of the sub-determinants of M to be zero. These conditions are all polynomial, and they define a variety (the intersection of their zero-sets) in the space of the p, n, c variables.

An arbitrary assignment of values to p, n, c does not correspond to an immersion of the mesh; the p-variables are all free, but the n and c will not obey Equations (2) and (3). Given a choice of p variables, the n and c variables of that embedding uniquely satisfy (2) and (3) (that is, the normal is indeed the cross-product, scaled as required). So there is a unique lifting of the Euclidean space defined by the vertex coordinate space p into (p, n, c)-space. Let $\tilde{P}$ be this lifting of the vertex coordinate space, which is Euclidean. The space $\tilde{P}$ is similarly simply connected and $3n$-dimensional.

If we have a connected component of an algebraic variety and we add an additional polynomial constraint to the system, either the whole component satisfies the new equation, or the dimension of the new variety is reduced by the intersection with the new equation. Thus, if there is any point of $\tilde{P}$ which does *not* also satisfy the system F verifying that a configuration is rigid, the set of common zeros (the space of dihedral infinitesimally non-rigid polyhedra) has smaller dimension than $\tilde{P}$, and forms a subset of measure zero in the space of possible configurations.

So all we need to do to show that the dihedral infinitesimally non-rigid polyhedra form a set of measure zero is to display some point in $\tilde{P}$ which is *not* in F; that is, a dihedral infinitesimally rigid polytope. As it happens, we can do this for every mesh topology; the results of Pogorelov [15] and Mazzeo and Montcouquiol [16] show that every convex polyhedron is dihedral infinitesimally rigid, and we know that every mesh topology can be realized as a convex polyhedron (this is Steinitz' theorem). This completes the proof of Theorem 2.

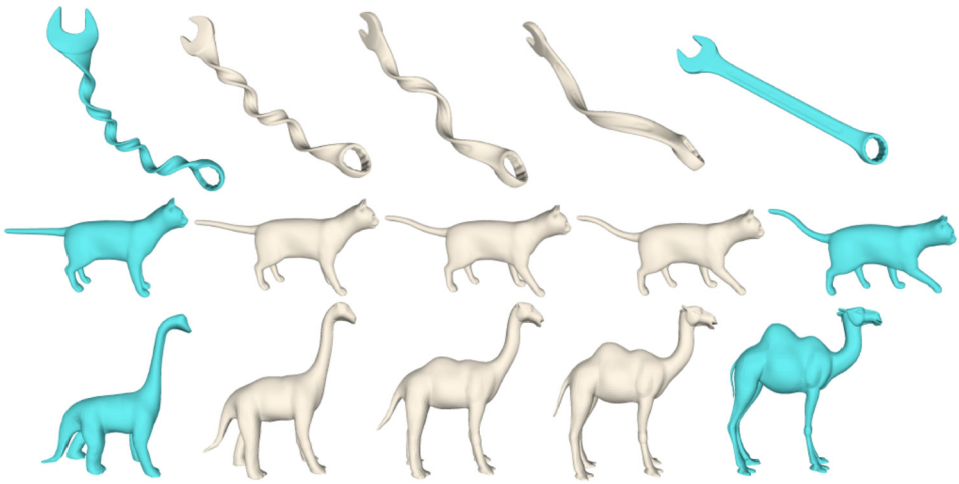

Fig. 1. Three examples of morphs between different shapes. The twisting of the wrench is handled nicely by considering dihedrals. The poses of the cat are interpolated naturally, without distortion in the intermediate shapes. The dinosaur and the camel have the same mesh topology, but are very different embeddings.

4. Visualizing dihedral interpolations

In the following, we will experiment with treating the dihedral vectors for a given mesh topology as a Euclidean shape space, doing arithmetic in that space and visualizing the results. For instance, Fig. 1 shows smooth interpolations between different configurations of some models from computer graphics, computed by linearly interpolating the dihedral vectors from one shape to that of the other, and producing a mesh approximating the interpolated dihedrals at each point. Videos of these smooth morphs can be seen at https://vimeo.com/270302684.

The space of possible shapes formed by embedding a triangulated surface had dimension $3|V| - 7$, that is, the number of mesh vertices modulo the seven-dimensional transformation space of rotation, translation and scale. A triangle mesh homeomorphic to the sphere has $3|V| - 6$ dihedral angles, one extra, so we do not necessarily expect the intermediate linearly-interpolated dihedral vectors to exactly correspond to embeddings; and, sadly, the dihedrals do not happen to lie in a linear subspace. One easy way to see this is to consider the seven ways a tetrahedron can be flattened into the plane; each coordinate in one of these seven dihedral vectors is either 0 or π, and they form a matrix of rank six, that is, $(3v - 6)$ not $(3v - 7)$.

Our plan is to nonetheless treat the dihedral space as Euclidean, perform statistical operations, and then visualize the results by producing embeddings that approximate the interpolated dihedral vector. This can be thought of as a projection to the subspace of realizable dihedrals, although we do not guarantee that we find the closest such point.

Our main tool to compute an embedding is a least-squares algorithm, Algorithm 1, that constructs an embedding, given two inputs: a vector of dihedrals and a vector of edge lengths. In the following section, we also give a refinement algorithm, Algorithm 2, that begins with this initial embedding and reduces the error of the dihedral vector at the expense of the edge-length error. While this second algorithm does reduce dihedral error, it does not give noticeably better visualizations and so most of our experiments are visualized using Algorithm 1 alone.

4.1. Algorithm 1: embedding

Constructing an embedding from a vector of dihedrals and a vector of edge lengths is a well-explored problem, and our approach in Algorithm 1 borrows ideas from previous algorithms in this space, particularly [7,23,24], and, computationally, [27]. The basic idea of finding a rotation and translation at each vertex of the mesh will be familiar from the ubiquitous as-rigid-as-possible (ARAP) interpolation method [6], although the computational approach is different. The input is the mesh topology, a vector of edge lengths, and a vector of dihedrals. There is no requirement that either the edge lengths or the dihedrals can be realized nor that both vectors can be realized by the same embedding. The output is an embedding that attempts to fit both edge and dihedral vectors.

First, at each vertex p_i, we construct a least-squares approximation to its star (the set of triangles containing p_i), achieving the desired dihedrals but introducing error in the edge lengths opposite p_i. We also define an arbitrary orthonormal coordinate system F_i at each vertex p_i. For every two stars at p_i and p_j connected by an edge e_{ij}, we find the three dimensional rotation R_{ij} that takes F_i to F_j when the two stars are merged along edge e_{ij}. This gives us a rotation associated with each edge. We use these rotations to solve for a coordinate system G_i at each vertex, that is, an orthonormal set of vectors in the global coordinate system, representing some rotation of F_i, by solving the system of equations:

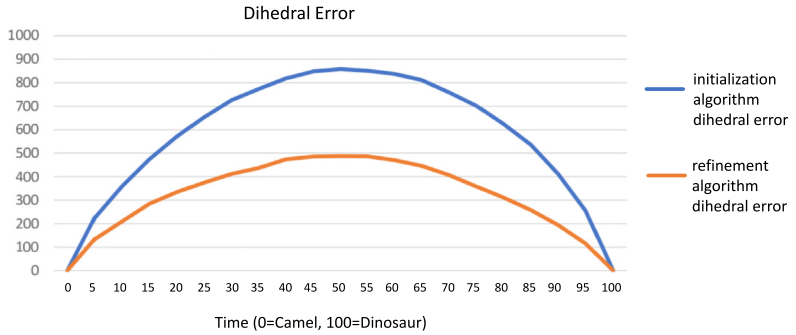

Fig. 2. Dihedral error reduction due to the refinement step, for the camel-to-dinosaur morph. The vertical axis is total dihedral error in radians. The horizontal axis is the time over which the morph occurs.

$$\min \sum_{e_{ij}} ||G_i R_{ij} - G_j||_F^2$$

where $||\cdot||_F$ indicates the Frobenius norm. This is a least-squares solve for the G_i, from the fixed R_{ij}. An arbitrary G_i is set to the identity. If the R_{ij} are consistent with some mesh, the resulting G_i would be a consistent set of orthonormal coordinate systems (or equivalently, a set of rotation matrices). But the R_{ij} may not be consistent with any set of global rotations (because all the stars adjacent to each vertex do not match up perfectly), so we typically end up with G_i that are not actually orthonormal. Following [28], we correct these using the singular value decomposition. We then use the G_i matrices to reconstruct vertex positions, again by least squares:

$$\min \sum_{\vec{e}_{ij}} ||(p_j - p_i) - G_i p_{ij}||^2.$$

Here p_{ij} represents the position of the copy of vertex p_j in the original local coordinate frame F_i at p_i; as transformed by the rotation G_i, it should be equal to $p_j - p_i$. The resulting coordinates p_i give the mesh embedding.

4.2. Algorithm 2: dihedral angle refinement

Algorithm 1, in the previous section, fits a mesh to both the interpolated dihedrals and interpolated edge lengths. This is not the mesh that best fits the interpolated dihedrals alone. In Algorithm 2, we refine the output of Algorithm 1 towards the interpolated dihedrals, allowing the edge lengths to change in the process. We use an iterative algorithm, which is computationally similar to the popular ARAP method [6].

We define an energy function E on the mesh, which considers both the normal vectors $\vec{n}_k$ and the vertices p_i:

$$E = \alpha \sum_{\substack{\text{adjacent} \\ \text{triangles } k,l}} ||M_{kl}\vec{n}_k - \vec{n}_l||_F^2 + \beta \sum_{\substack{\text{edge } i,j \\ \text{triangle } k}} ||L_{ijk}(p_i - p_j) - \vec{n}_k||^2. \tag{4}$$

Here $\vec{n}_k$ is the normal of triangle k and $\vec{n}_l$ is the normal of triangle l, adjacent across edge i, j. The matrix M_{kl} is a rotation by exactly the desired dihedral angle δ_{ij} between $\vec{n}_k$ and $\vec{n}_l$, with the axis of rotation $\vec{e}_{ij} = (p_i - p_j)/||p_i - p_j||$. Thus the first energy term measures how well the normals achieve the dihedral angles at every edge. The second term measures how well the normals and vertices agree with each other. The matrix $L_{i,j,k}$ takes edge i, j into the normal of one of its adjacent triangles k. It is the product of a rotation by $\pi/2$, along with a scaling to normalize the length. We use the weights $\alpha = 0.6$ and $\beta = 0.4$, which we selected experimentally.

At each step, we recompute M and L from the current mesh, solve for new $\vec{n}_k$ while keeping the p_i fixed, and finally solve for new values of p_i, repeating until convergence (see Fig. 3).

We see that this algorithm succeeds in reducing the dihedral error of the interpolations by about half. We define the dihedral error simply as the Euclidean difference between the desired interpolated dihedral vector and the actual dihedrals achieved by our embedding; an example appears in Fig. 2. When using one of the input shapes as the initialization mesh, we found that the refinement energy function took more iterations to converge and it did not achieve the same minimum error, as shown in Fig. 3.

5. Shape analysis: human body shapes

This section shows some experiments with shape analysis in the dihedral space, on some well-known collections of three-dimensional triangulated meshes of human body shapes. The meshes in each set are combinatorially identical.

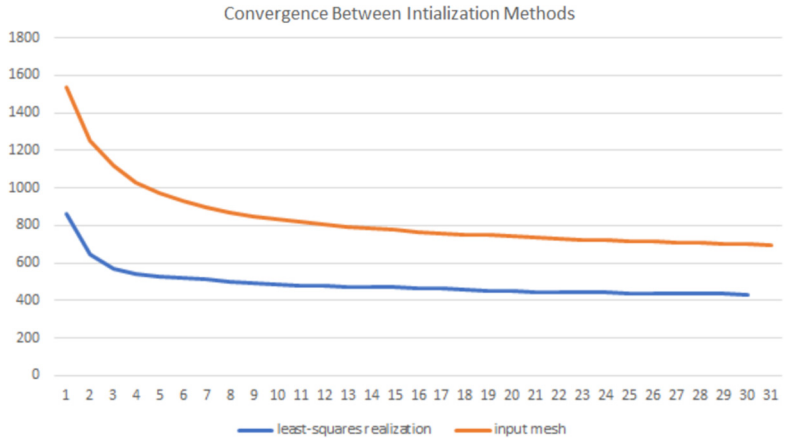

Fig. 3. Convergence when using one of the input meshes for initialization versus Algorithm 1.

We use the following approach for all of the output visualizations. First, we compute an average set of edge lengths E for the entire input collection. Then, for every dihedral vector V_k we want to visualize, we use Algorithm 1 to create an embedding approximately realizing V_k and E. This ensures that differences that we see in the visualizations depend only on V_k, and not on E.

We begin with the ground-truth subset of the MPI FAUST dataset. The entire FAUST dataset include 300 human 3D laser scans in a wide range of poses [29], and it is intended as a benchmark for registration methods. Its ground truth subset is given as a set of embeddings of a single topological mesh, representing 10 subjects each in 10 different poses, labeled by subject and pose. Each mesh has approximately 7,000 vertices.

In the dihedral space, we used PCA to reduce the dimensions of the ground-truth dataset; a scatterplot on these first two principal coordinates is shown at the top in Fig. 4. There are two obvious clusters, corresponding to gender. That is, in dihedral space the most salient features are those that reflect body shape rather than pose or size. This is not at all true in Euclidean Procrustes space formed by the $3|V|$ vertex coordinates. Quite large changes in shape can be visualized by varying only the dihedrals, not the edge lengths.

Variation of the body shape along the first principal component in dihedral space is shown at the bottom of Fig. 4. Fig. 5 shows a similar visualization but focuses on the males in the dataset; without the significant male-female body shape variation, pose becomes the most significant component. All of the shapes representing different values of the principal component are natural-looking poses, without distortion. Again, it is well-known that this would not be the case in the Procrustes space.

Within the Euclidean dihedral space, we can apply simple vector space operations. In Fig. 6, we capture pose variation via dihedral space vector operations on the human subjects from the FAUST dataset, by removing variation based on the body-shape of the individual subjects, leaving only variation based on pose. First, we produce a per-subject average body shape $A_i, i \in [0, ..., 9]$ for each of the ten subjects. We subtract the overall average A of the entire dataset from each A_i to produce a per-subject body-shape offset. Next, for a given scan $S_{i,j}$ of subject i in pose j, we subtract the per-subject offset, such that the residual vector of dihedrals $P_{i,j}$ should represent the average person demonstrating subject i's realization of pose j:

$$P_{i,j} = S_{i,j} - (A_i - A). \tag{5}$$

Fig. 6 plots these pose examples $P_{i,j}$ using the first two principal components (just for poses $j \in [5, ..., 9]$; poses 0-4 were omitted for clarity, since it is hard to see all ten clusters in a 2D projection). We can see that this residual pose data indeed reveals five distinct clusters. On the right, we visualize three poses from the same cluster, all of which have the average (hermaphrodite) body shape.

Finally, we examine The Human Body Models collection, which is derived from the CEASAR dataset. It contains 1,500 registered male and female meshes with a vertex-to-vertex correspondence [30]. This dataset differs from the FAUST dataset in that the subjects are all in (roughly) the same pose (they were collected this way so that the Procrustes space could be used for shape analysis). Each mesh has approximately 12,000 vertices.

Again, the first principal component in dihedral space shows the variation between the male and female subjects, see Fig. 7, and the second principal component captures one of the slight variations in pose, which is mostly a difference in stance width.

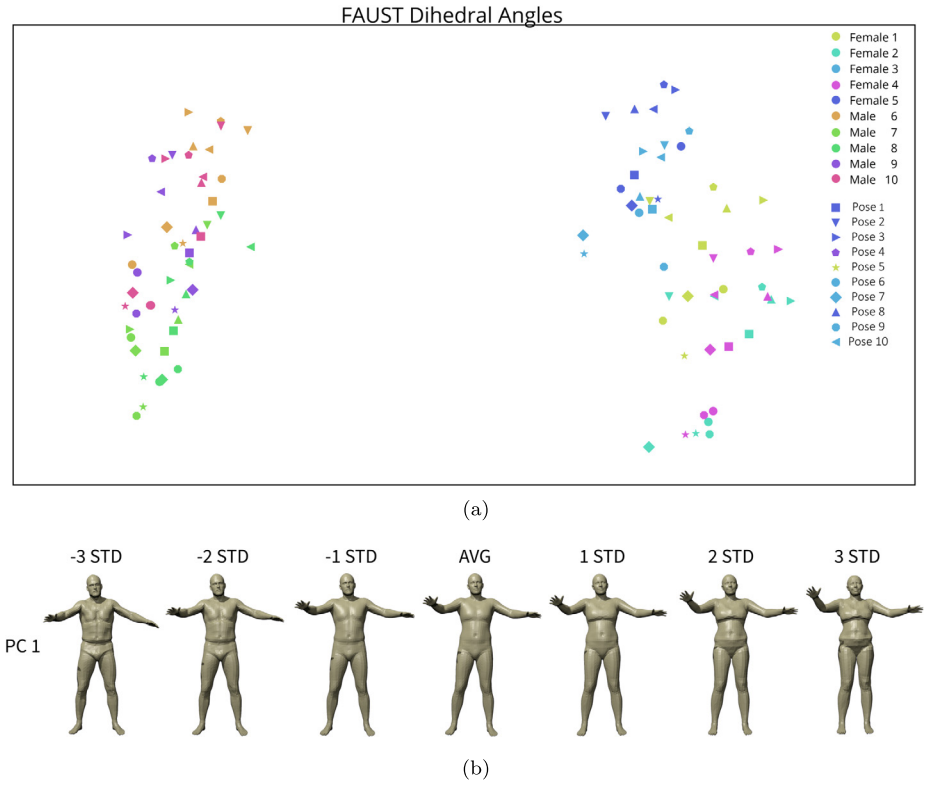

Fig. 4. The top principal component in dihedral shape space for the FAUST human body shape data describes the fundamental shape difference between male and female bodies. When we plot the first two principal components (a) we clearly see the separation between the group of male and female subjects. In Figure (b) we warp the average shape in the direction of the first principal component, by adding multiples of it to the average shape. All of the figures are reconstructed using the same edge lengths. Each edge length is averaged over the whole input set.

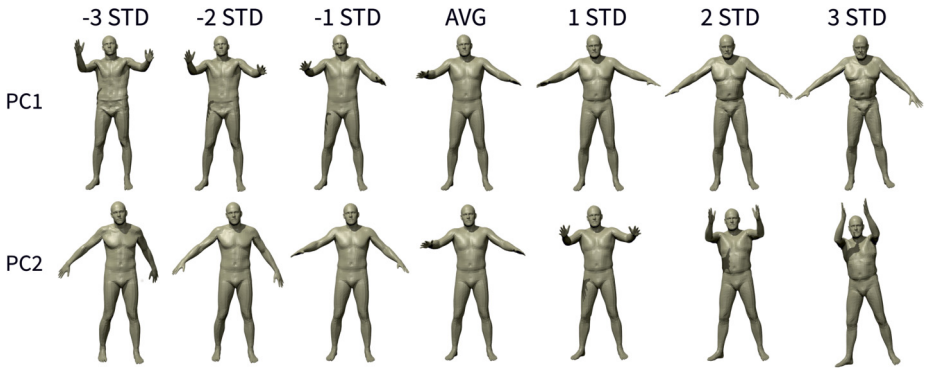

Fig. 5. The top two principal components of just the males from the FAUST dataset. This visualization shows the variation in pose.

6. Shape analysis: comparison of Euclidean shape spaces

To do shape analysis on a set of mesh embeddings using Procrustes space (aka root-mean-squared distance), we begin by aligning the meshes as well as possible to reduce the effect of rigid motions and scale. The generation of a mutual alignment for a shape collection is known as generalized Procrustes analysis (GPA). As illustrated in Fig. 10, Procrustes space is widely known to be inappropriate for the analysis of shape collections in which individual parts differ by large rigid motions, even after the models are rigidly aligned. The twisted wrench is an extreme example, but the rigid motions of parts exhibited by articulated figures such as humans and animals occurs all the time. Procrustes space is, however, widely used for the comparison of shapes where pose is less of an issue, especially in medical imaging and anthropology, and in facial animation ("blend shapes").

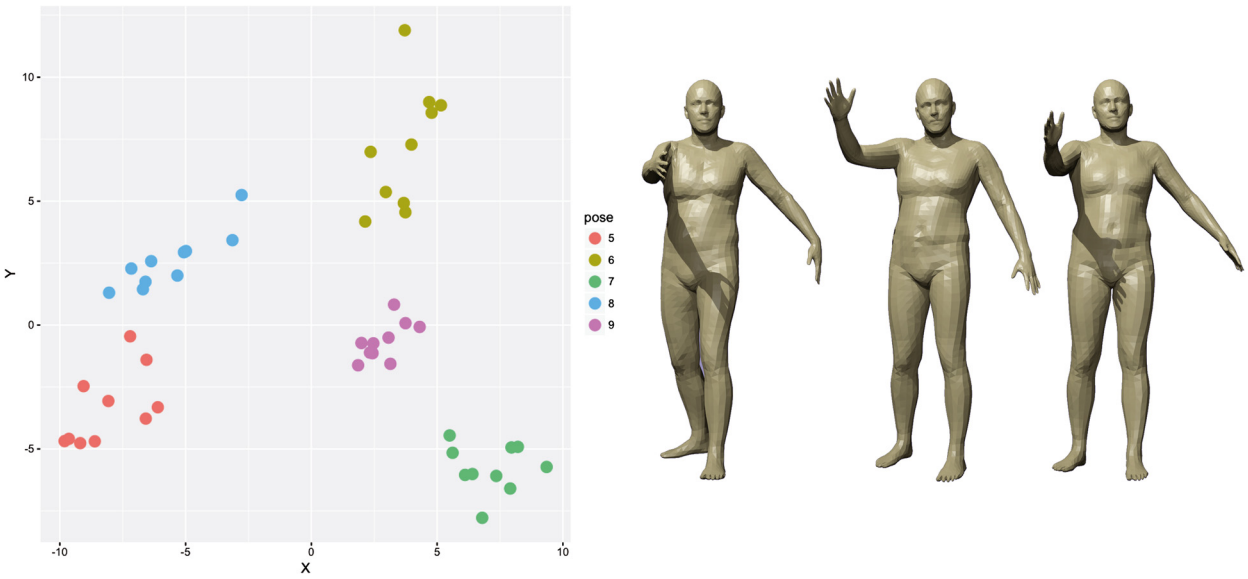

Fig. 6. On the left we removed the individual specific features from each mesh in the FAUST data using simple linear algebra in dihedral space. Plotting the first two principal component values of the residual shows good clustering by pose. On the right, three of the residual models in pose 9, corresponding to the three of the pink points. Like most of the models in this section, these are all reconstructed using the average edge lengths over the whole dataset. We see that the residual body shapes are slightly different, but close to the average. The similar body shapes emphasize the difference in how the three people struck the pose, as opposed to the appearance of the three subjects. (For interpretation of the colors in the figure(s), the reader is referred to the Web version of this article.)

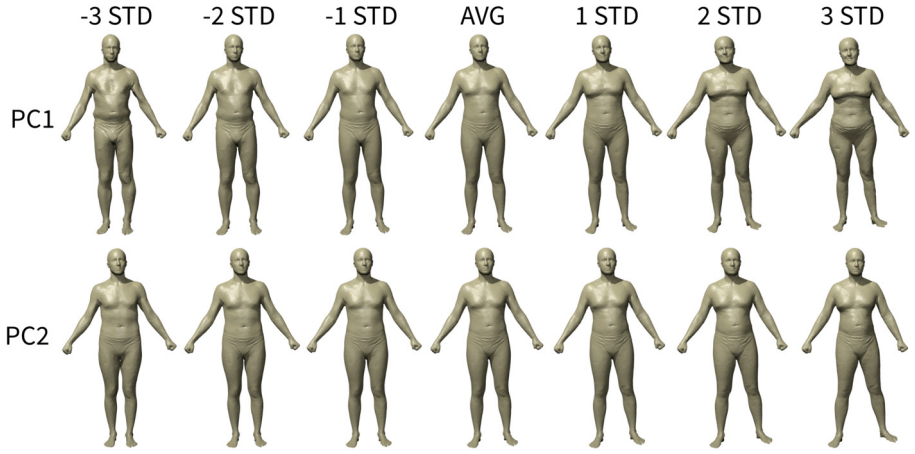

Fig. 7. The first two principal components in dihedral space of the Human Body Models dataset. As with the much smaller FAUST dataset, the first component corresponds to gender. The second corresponds almost entirely to the small differences in the pose adopted by the subjects.

A related theoretical problem with Procrustes space is that the best mutual alignment of the entire set of models only approximates the shape distances given by the best-possible alignments of each pair of meshes. The "correct" shape space realizing the pair-wise distances is a curved subspace of $\mathbb{R}^{3|V|}$.

To compare dihedral space with Procrustes space, we looked at principal components on two examples in which a good mutual alignment is possible (so that Procrustes space can be used). The first is a set of human face scans gathered by Notre Dame for their 2002 facial recognition contest, distributed by NIST. Each facial scan is labeled with a facial expression.

We took a sub-collection of paired scans, each showing the same person with both a happy and a sad expression, and computed the top three principal components in each space. Interpolations along these principal components are shown in Fig. 11. Here Procrustes space does a better job of separating expression into the first principal component and face shape into the others, although dihedral does this as well to some extent.

A second example, in which dihedral space provides more useful information, is a collection of Old World monkey crania, provided by the New York Consortium on Evolutionary Primatology. The collection is made up of four species of baboon, six species of macaque, and a mandrill. Baboons and macaques are closely related species, but baboons have long noses, and macaques have short noses. Mandrills are long-nosed monkeys that are more distantly related to the other two species.

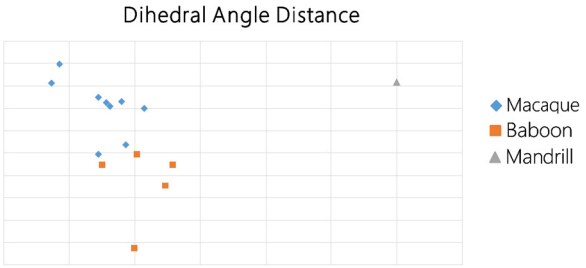

Fig. 8. Three species of Old World monkeys. Macaques have a long snout while Baboons have a short snout. Mandrill has a long snout and it is distantly related.

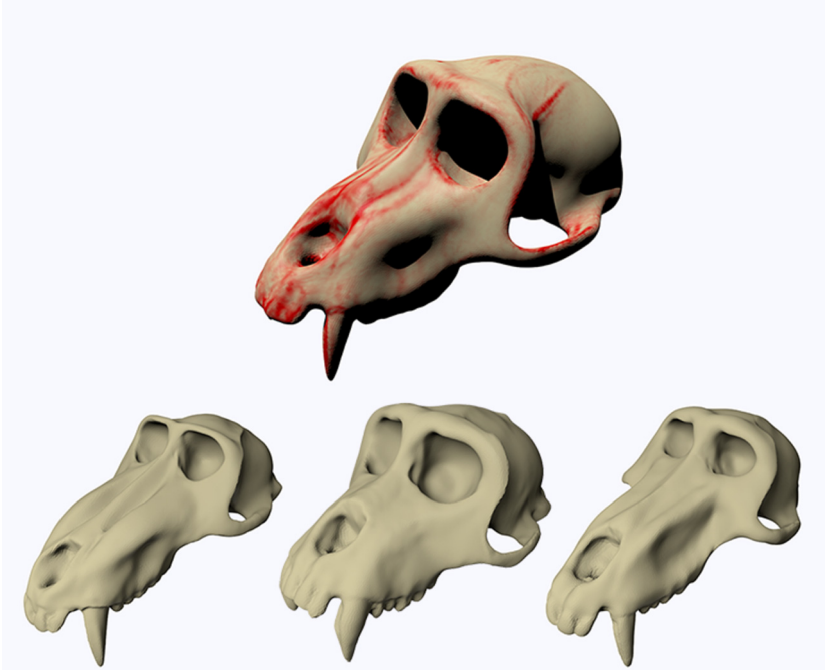

Fig. 9. In dihedral space, the first principal component of the monkey cranium data clearly differentiates the mandrill from the macaques and baboon. We see a mandrill (bottom left) and a baboon (bottom right) cranium, along with the average cranial shape (including the short-nosed macaques) in dihedral space (bottom middle). The top cranium is the average of the mandrill and baboon where red indicates regions that vary significantly along the first principal component.

Fig. 10. Linear interpolation between a wrench and a twisted wrench in Procrustes space; compare with linear interpolation in dihedral space in Fig. 1. Interpolated shapes lose parts of the model.

In Fig. 8, we see in the projection to the first two principal components that mandrill crania differ in shape from macaque and baboon crania. Because they are both long-nosed, the Procrustes distance from baboon to mandrill is less than the Procrustes distance between baboon and macaque; that is, the dihedral distance does a better job of capturing the subtle evolutionary shape similarity between baboons and macaques which is obscured in Procrustes space by the larger-scale difference of long- vs. short-nosed crania. Highlighting the areas in which the dihedral distance is greatest in Fig. 9, we see the shape of the eye sockets, ridges behind the eye sockets, and the shape of the snout. Looking at the first three principal components in Fig. 12, we see these shape differences highlighted in the first few components of the dihedral distribution, while the Procrustes space mostly highlights the long- vs. short-nosed difference. We see in this case that dihedral space gives qualitatively different information, which is relevant to the evolutionary relationship of the species.

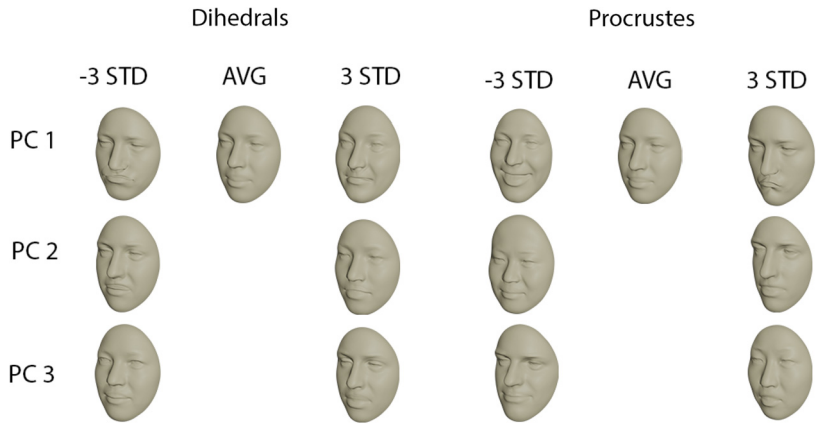

Fig. 11. Visualization of the top three principal components of the face data in dihedral space and Procrustes space. While both dihedral space and Procrustes space capture the variation between happy and sad faces as the first principal component, the dihedral space (or at least our visualization of it) is less emphatic. In the second component, Procrustes space focuses on wide vs. narrow faces, while dihedral space only captures differences in features and expression.

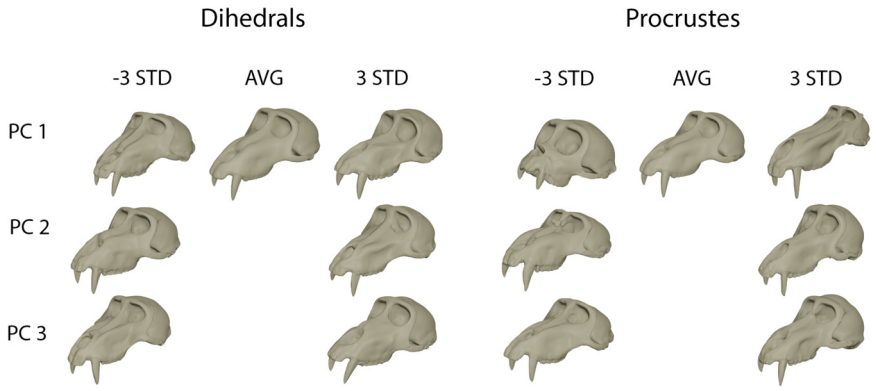

Fig. 12. The top three principal components of the Old World monkey data in dihedral space and Procrustes space. The first principal component in dihedral space captures the significant shape difference between the closely-related baboons and macaques on the one hand, and the mandrill on the other. In Procrustes space, the first principal component reflects the larger-scale distinction between long-nosed and short-nosed monkeys. This large-scale difference does not allow the macaques and baboons to ever be aligned well, and their similarity, highlighted in dihedral space, is not detected. The distinctions between the shapes of the eye sockets, the ridges on the nose, and so on, which distinguish the mandrill from the baboon, do not show up, even in the second and third components. For complicated shapes like this, dihedral space provides essential information that Procrustes space does not.

7. Discussion

There is a large gap between the very basic level of our mathematical understanding of the dihedral vectors of mesh embeddings and the potential reflected in our experimental work. Perhaps the most significant open conjecture is:

Conjecture 1. *There is at most one set of inner face angles consistent with an embedding of a mesh realizing a given vector of dihedral angles.*

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We thank the referees for many constructive suggestions, and we gratefully acknowledge the support of NSF grant IIS 1117663 for the early stages of this project, and of the Tim Bucher chair at UC Davis.

References

[1] R. Bricard, Mémoire sur la théorie de l'octaèdre articulé, J. Math. Pures Appl. 3 (1897) 113–148.

[2] R. Connelly, A counterexample to the rigidity conjecture for polyhedra, Publ. Math. Inst. Hautes Études Sci. 47 (1977) 333–338.
[3] R. Connelly, The rigidity of polyhedral surfaces, Math. Mag. 52 (1979) 275–283.
[4] H. Gluck, Almost all simply connected closed surfaces are rigid, in: Geometric Topology, Springer, 1975, pp. 225–239.
[5] A.I. Bobenko, U. Pinkall, B.A. Springborn, Discrete conformal maps and ideal hyperbolic polyhedra, Geom. Topol. 19 (2015) 2155–2215.
[6] O. Sorkine, M. Alexa, As-rigid-as-possible surface modeling, in: Symposium on Geometry Processing, vol. 4, 2007, p. 30.
[7] S.-Y. Baek, J. Lim, K. Lee, Isometric shape interpolation, Comput. Graph. 46 (2015) 257–263.
[8] Z. Zhang, G. Li, H. Lu, Y. Ouyang, M. Yin, C. Xian, Fast as-isometric-as-possible shape interpolation, Comput. Graph. 46 (2015) 244–256.
[9] W.Y. Lam, U. Pinkall, Infintessimal conformal deformations of triangulated surfaces in space, Discrete Comput. Geom. 60 (2018) 831–858.
[10] M. Ovsjanikov, M. Ben-Chen, J. Solomon, A. Butscher, L. Guibas, Functional maps: a flexible representation of maps between shapes, ACM Trans. Graph. 31 (2012) 30.
[11] D. Boscaini, J. Masci, E. Rodolà, M. Bronstein, Learning shape correspondence with anisotropic convolutional neural networks, in: Advances in Neural Information Processing Systems, 2016, pp. 3189–3197.
[12] C. Rojas, A. Tsui, S. He, L. Simons, S. Li, N. Amenta, Edge length interpolation, in: ACM Symposium on Solid and Physical Modeling, 2014. Poster paper.
[13] J.J. Stoker, Geometrical problems concerning polyhedra in the large, Commun. Pure Appl. Math. 21 (1968) 119–168.
[14] A.L. Cauchy, Sur les polygones et polyedres, J. Éc. Polytech. 16 (1813) 87–99.
[15] A. Pogorelov, On a problem of stoker, Dokl. Akad. Nauk 385 (2002) 25–27.
[16] R. Mazzeo, G. Montcouquiol, et al., Infinitesimal rigidity of cone-manifolds and the stoker problem for hyperbolic and Euclidean polyhedra, J. Differ. Geom. 87 (2011) 525–576.
[17] H. Crapo, W. Whiteley, Statics of frameworks and motions of panel structures: a projective geometric introduction, Topol. Struct. 6 (1982) 42–82.
[18] I. Pak, Lectures on Discrete and Polyhedral Geometry, 2010, www.math.ucla.edu/~pak/book.htm, online book.
[19] M. Kilian, N.J. Mitra, H. Pottmann, Geometric modeling in shape space, ACM Trans. Graph. 26 (2007) 64.
[20] B. Heeren, M. Rumpf, P. Schröder, M. Wardetzky, B. Wirth, Exploring the geometry of the space of shells, Comput. Graph. Forum 33 (2014) 247–256.
[21] B. Heeren, M. Rumpf, P. Schröder, M. Wardetzky, B. Wirth, Splines in the space of shells, Comput. Graph. Forum 35 (2016) 111–120.
[22] C. Zhang, B. Heeren, M. Rumpf, W.A. Smith, P.C.A. Shell, Statistical shape modelling in shell space, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1671–1679.
[23] S. Kircher, M. Garland, Free-form motion processing, ACM Trans. Graph. 27 (2008) 12.
[24] T. Winkler, J. Drieseberg, M. Alexa, K. Hormann, Multi-scale geometry interpolation, Comput. Graph. Forum 29 (2010) 309–318.
[25] G.-P. Paillé, N. Ray, P. Poulin, A. Sheffer, B. Lévy, Dihedral angle-based maps of tetrahedral meshes, ACM Trans. Graph. 34 (2015) 54.
[26] M. Isenburg, S. Gumhold, C. Gotsman, Connectivity shapes, in: Proceedings of the Conference on Visualization'01, IEEE Computer Society, 2001, pp. 135–142.
[27] Y. Lipman, O. Sorkine, D. Levin, D. Cohen-Or, Linear rotation-invariant coordinates for meshes, ACM Trans. Graph. 24 (2005) 479–487.
[28] J. Mao, Optimal orthonormalization of the strapdown matrix by using singular value decomposition, Comput. Math. Appl. 12 (1986) 353–362.
[29] F. Bogo, J. Romero, M. Loper, M.J. Black, Faust: dataset and evaluation for 3d mesh registration, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 3794–3801.
[30] Y. Yang, Y. Yu, Y. Zhou, S. Du, J. Davis, R. Yang, Semantic Parametric Reshaping of Human Body Models, 2nd International Conference on 3D Vision (3DV), 2014, vol. 2, IEEE, 2014, pp. 41–48.