Using isometries for computational design and fabrication

CAIGUI JIANG, KAUST, Saudi Arabia
HUI WANG, KAUST, Saudi Arabia
VICTOR CEBALLOS INZA, KAUST, Saudi Arabia
FELIX DELLINGER, TU Wien and TU Graz, Austria
FLORIAN RIST, KAUST and TU Wien, Austria
JOHANNES WALLNER, TU Graz, Austria
HELMUT POTTMANN, KAUST, Saudia Arabia

We solve the task of representing free forms by an arrangement of panels that are manufacturable by precise isometric bending of surfaces made from a small number of molds. In fact we manage to solve the paneling task with surfaces of constant Gaussian curvature alone. This includes the case of developable surfaces which exhibit zero curvature. Our computations are based on an existing discrete model of isometric mappings between surfaces which for this occasion has been refined to obtain higher numerical accuracy. Further topics are interesting connections of the paneling problem with the geometry of Killing vector fields, designing and actuating isometries, curved folding in the double-curved case, and quad meshes with rigid faces that are nevertheless flexible.

CCS Concepts: • **Computing methodologies** → **Shape modeling**; *Optimization algorithms.*

Additional Key Words and Phrases: discrete differential geometry, computer-aided design, computational fabrication, architectural geometry, discrete isometry, isometric registration, Killing vector field

1 INTRODUCTION

In this work we apply recent progress in isometric mappings to the paneling of freeform architectural skins. The paneling problem, i.e., the decomposition of curved shapes into smaller elements that are more easily manufacturable, is crucial to the actual realization of freeform designs on the architectural scale. The choice of panel geometry is central to visual appearance and thus is an integral part of the artistic aspect of freeform architecture. At the same time this choice has far-reaching consequences on manufacturing and poses big computational challenges. There are at least two reasons

Authors' addresses: Caigui Jiang, KAUST, Saudi Arabia; Hui Wang, KAUST, Saudi Arabia; Victor Ceballos Inza, KAUST, Saudi Arabia; Felix Dellinger, TU Wien and TU Graz, Graz, Austria; Florian Rist, KAUST and TU Wien, Wien, Austria; Johannes Wallner, TU Graz, Graz, Austria; Helmut Pottmann, KAUST, Saudia Arabia.
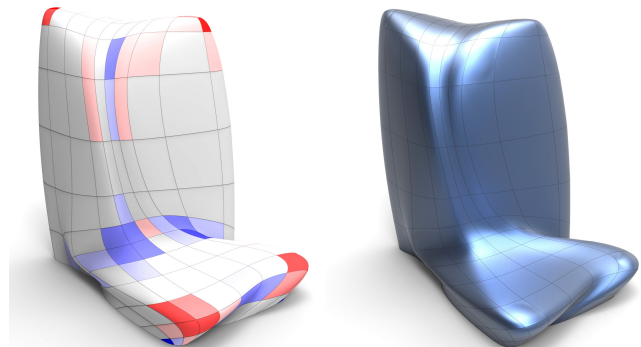
Fig. 1. Smooth freeform skins from flexible panels. It turns out that constant Gaussian curvature surfaces, i.e., spheres and pseudospheres, provide almost universal molds for the manufacturing of panels bending isometrically. Here a design by Zaha Hadid Architects is covered by panels struck from a small number of spherical molds (red) and pseudospherical molds (blue), as well as single-curved panels that are intrinsically flat (white).

for that: one is that a geometric fit of all panels may amount to a global and nonlinear system of constraints that is hopeless to solve by black box methods. Another is the sometimes prohibitive cost of manufacturing panels, in particular double-curved ones. For an overview of this broad topic we refer to [Pottmann et al. 2015].

In the case of an overall smooth skin, the most successful paradigm for reducing costs has been to break up the fabrication process into two stages, namely manufacturing *molds* first, and afterwards making the panels from them. Since typically molds are more expensive than striking an individual panel from the mold, the idea is to reuse molds. This concept has been used in practice several times already, e.g. for the Arena Corinthians stadium in São Paolo which was completed in 2014.

Beyond its literal meaning, "mold" may refer to any machine or technique used to produce a certain family of panels. One can think of all flat glass panels to be associated with the same mold (which is cheap, of course). Likewise all glass panels that lie on the surface of any right circular cylinder correspond to a single "mold" which in reality is the machine capable of producing cylindrical shapes, cf. [Baldassini et al. 2013].

Our work considers molds in the literal sense. In contrast to previous work on paneling in architecture, panels are put into place not by a rigid transformation, but are bent isometrically. This method

is applicable only to materials that allow isometric deformations like sheet metal, certain plastics and textiles, and it drastically reduces the number of molds that are needed. We are able to accurately model this process thanks to recent progress in the computational treatment of isometric mappings: we employ and improve the 'checkerboard pattern' method proposed by Jiang et al. [2020].

A perhaps surprising discovery is that we found it sufficient to use a rather small set of molds of high intrinsic symmetry, namely surfaces of constant Gaussian curvature. This means that in most instances we were able to cover free forms by panels that are either developable, or are obtained from a small set of spherical and pseudo-spherical molds. On an abstract mathematical level this procedure corresponds to the approximation of the Gaussian curvature function by a piecewise-constant function. It was unexpected how well this idea works in practice.

It is important to stress the difference between our work and previous progress on non-rigid isometric registration in Computer Graphics and Computer Vision [Bronstein et al. 2008], the greater part of which is concerned with as-isometric-as-possible matching of different shapes, not with exact isometric registration.

We would like to add two historical remarks. Firstly, the well known sails of the Sydney opera house enjoy spherical geometry of a single radius [Lewis 1973] and thus constitute a panelization via a single spherical mold.

Secondly we point to the 19th century invention of *French curves* useful for drawing a great number of curves for which ruler and compass are not sufficient but which occur in technical drawings. The act of drawing can be thought of as panelization by means of the French curve set as molds. One could say that our experimental results confirm that as far as isometrically bending panels are concerned, we can realize a great number of freeform architectural skins by flat/spherical/pseudospherical molds, and we do not have to resort to more complicated "French curve" molds.

1.1 Contributions

The new contributions of this paper are the following.

- We improve the precision of Jiang et al.'s [2020] checkerboard method for isometric mappings by means of adapted regularizers (§2.2). We also provide new insights on approximation order (§2.1).

- §3.1 presents a structure-preserving discretization of the concept of infinitesimal isometry.

- We compute Killing vector fields, i.e., intrinsic infinitesimal isometries, in a simple way (§3.2). They are employed to detect surfaces that possess advantages in fabrication because they are isometric to surfaces exhibiting symmetries like rotational surfaces.

- In §4 we discuss precise isometric registration, which in §5.1 is used for the paneling problem. We present two computational pipelines for paneling freeform architectural skins with bendable materials — one with constant Gaussian curvature surfaces as molds, and another one with molds of rotational symmetry.

- Further applications (§5.2) include the design of quad meshes which are flexible if the faces are moving rigidly and are connected by the edges as hinges.

1.2 Previous Work

Previous Work in Freeform Architecture. On the architectural scale, smooth shapes can be obtained by different means, e.g. as equilibrium of a membrane. More often they are an assembly of individual panels that seamlessly join to form the surface. The high cost of manufacturing double-curved panels in the past has led architects to designs with repetitive geometry. This can e.g. take the form of a hidden rotational symmetry which is not visually obvious because panel boundaries attract attention away from it – the 2003 lens-shaped entrance to the St. Lazare metro station in Paris is an instance of this. For such simple surfaces, panels in different places of the surface can be struck from the same mold, and the cost of molds plus panels is significantly reduced. An extension of this idea to true free forms has been performed by Eigensatz et al. [2010] and has been used in practice [Schiftner et al. 2013]. The present paper can be seen as a continuation of this work, but with non-rigid panels that bend isometrically.

As to curved panels originating in a flat rest state, prior contributions are mainly concerned with elastic materials, in which case paneling is modelled by mappings that are not isometric. Examples include flat pre-stretched tilings assuming a desired shape upon release [Guseinov et al. 2017], auxetic micro-mechanisms [Konaković-Luković et al. 2018], micro-mechanisms based on spiral springs [Malomo et al. 2018] and deployable structures of a gridshell nature [du Peloux et al. 2013; Panetta et al. 2019]. Most recently, and relevant to our work, Gavriil et al. [2020] used learning methods to panel architectural freeform skins by cold-bent glass, taking the restrictions of the material into account.

Previous Work on Isometric Mappings Between Surfaces. Finding correspondences between geometric shapes is one of the most significant and diverse problems in geometry processing. We here focus on *isometric* mappings, i.e., mappings preserving the intrinsic distances measured inside the surface. Since in general two surfaces cannot be connected by an isometry, mappings that are near-isometric have been studied [Claici et al. 2017; Liu et al. 2009; Pietroni et al. 2010; Sorkine and Alexa 2007]. Relevant work on isometric mappings has been done by Chern et al. [2018]. Based on a discrete theory of spin structures, they treat isometric mappings, particularly immersions. Recently, Sassen et al. [2020] investigated

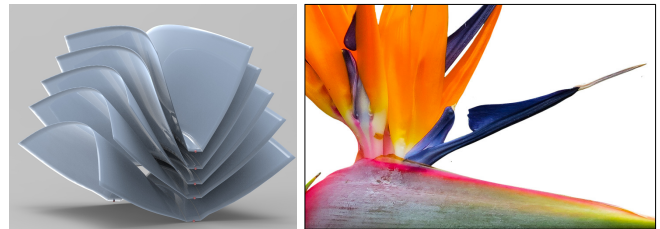Fig. 2. *Isometric deformation of a mesh.* The deformation at left is analogous to the "flectofin" shading elements [Lienhard et al. 2011] which are bionic tech inspired by the naturally occurring hinge in the flower of *Strelitzia reginae* (blue, in photo at right). The intrinsic geometry of the surface implies that a light bending of the bottom corresponds to a much larger movement of the flaps to either side.
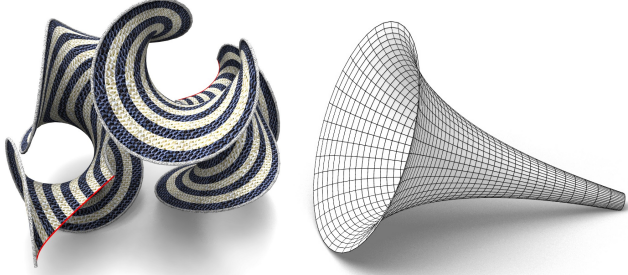
Fig. 3. Surfaces of constant negative Gaussian curvature K. They are (locally) isometric if and only if their respective Gaussian curvatures agree. The unit pseudosphere shown at right enjoys $K = -1$ and is generated by rotating the curve $y = \log(\frac{1}{x}(1 + \sqrt{1 - x^2})) - \sqrt{1 - x^2}$ about the y axis.

the flexions of triangle meshes with rigid faces via the manifold of realizations of fixed edge lengths but variable dihedral angles. They formulate the corresponding integrability conditions and solve both exact and approximate reconstruction of meshes, which includes computing isometric mappings. Our paper is based on a different approach, namely checkerboard patterns as proposed by Peng et al. [2019]. They work with patterns exhibiting rectangles. Extending this idea to the general case of parallelograms, Jiang et al. [2020] developed an isometric deformation model by requiring that the parallelograms in a checkerboard pattern move rigidly. This is an instance of the more general principle to model isometry via keeping the shape of small elements, see e.g. [Bouaziz et al. 2012].

Isometric Bending. The differential geometry of time-continuous isometric bending of surfaces is extensively surveyed by [Sabitov 1992]. Relevant to our work are statements on existence of bending determined by initial values.

As to applications in engineering and nature, it is well known that the bending of paper and similar thin-sheeted materials is modelled by isometric mappings. Most of the extensive literature on this subject is concerned with developable surfaces created by bending flat sheets. We do not review it here but instead refer to [Jiang et al. 2020; Liu et al. 2006]. Bending of non-flat surfaces occurs in manifold ways in herbal tissue, and special cases of it have inspired technology: The "Flectofin" shading mechanism is modelled after the natural hinge present in the bending behaviour of the flower of *Strelitzia reginae*. See [Lienhard et al. 2011; Masselter et al. 2012] and Figure 2.

Infinitesimal Isometries. We also make use of the concept of infinitesimal isometry, which is the time-derivative vector field of a deformation that is isometric up to the first order of differentiation. This is an old topic and connections to meshes are summarized by Sauer [1970]. Tangential infinitesimal isometries (Killing vector field) have a flow that is isometric. For the geometry of Killing vector fields we refer to the survey [Myers 1936]. Computation and applications have been demonstrated by [Ben-Chen et al. 2010; Solomon et al. 2011a]. In particular they have been employed to detect parts of surfaces enjoying intrinsic symmetries [Solomon et al. 2011b].

Constant Gaussian Curvature Surfaces. In this paper we employ surfaces of constant Gaussian curvature, i.e., surfaces isometric to

parts of the plane, or to appropriately scaled copies of the unit sphere S^2 resp. the unit hyperbolic plane H^2 [Alekseevskij et al. 1993]. An often-used surface exhibiting $K = -1$ is the *unit pseudosphere*, see Fig. 3. Efforts to build surfaces isometric to a part of H^2 such as the one shown by Fig. 3 started as soon as H^2 was discovered. They include kinematic metal models [Wunderlich 1951] and even crochet [Henderson and Taimina 2001]. Both the textile and the metal versions are relevant to paneling on the architectural scale.

Previous Work on Isometric Registration. Non-rigid registration is a much studied topic in Vision and Geometry Processing. The special case of isometric registration has been discussed by various authors, see e.g. [Huang et al. 2008; Wand et al. 2007]. Both the initial alignment (*global registration*) and the later fine tuning through numerical optimization (*local registration*) have been treated. These methods have been successfully employed for tasks in engineering [Klein et al. 2014; Sacharow et al. 2011]. Most of the work however is not addressing high accuracy applications like the ones we have in mind and thus cannot be used directly. Huang et al. [2008] use features of the target surfaces for registration. This is not suitable for our setting, architectural freeform skins typically lacking prominent features.

2 DISCRETE ISOMETRIES

For the computations with surfaces and their isometric mappings, we represent a surface by a quad mesh which is further subdivided into a checkerboard pattern as proposed by Jiang et al. [2020]. Each original face is associated with a parallelogram formed by the edge midpoints, while each original vertex is associated with a quad, formed by the midpoints of edges emanating from that vertex. In the inset figure, inscribed parallelograms are shown in yellow. Figure 4 illustrates the fact that they are parallel to the diagonals of faces of the original mesh. Following Jiang et al. [2020], we say the surface deforms isometrically, if each inscribed parallelogram moves as a rigid body.

2.1 Approximation Order of Discrete Isometries

We need to convince ourselves that discrete-isometric mappings of checkerboard patterns do indeed approximate continuous isometries, and that the discrete-isometric conditions take away just the right number of degrees of freedom.

For that, consider a quad mesh which is the image, under a smooth mapping ϕ, of the standard lattice $h\mathbb{Z}^2$ with stepsize h. Focus on the

$$\frac{1}{2}(v_2 - v_0)$$
$$= m_{12} - m_{01}$$
$$= m_{23} - m_{30}$$

$$\frac{1}{2}(v_3 - v_1)$$
$$= m_{30} - m_{01}$$
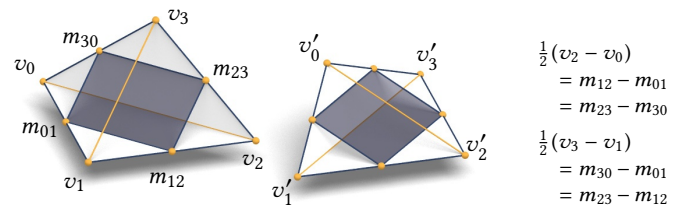$$= m_{23} - m_{12}$$

Fig. 4. A parallelogram $m_{01}m_{12}m_{23}m_{30}$ inscribed in a quad $v_0v_1v_2v_3$. The edges of the former are half the diagonals of the latter. A mapping of vertices $v_i \mapsto v_i'$ is called isometric, if inscribed parallelograms transform rigidly. This image is taken from [Jiang et al. 2020].
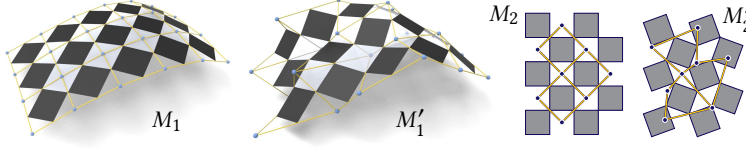
Fig. 5. *Nonsmooth discrete isometries.* Discrete isometries $M_i \to M_i'$ approximate smooth ones only if meshes are fair and edge vectors can be interpreted as derivatives. Deviation from fairness may happen orthogonally to smooth reference surfaces or even inside them, as demonstrated by flat meshes M_2, M_2'.

face $v_0 v_1 v_2 v_3$ whose vertices are images of $(0,0)$, $(h,0)$, (h,h) and $(0,h)$ respectively.

It is easy to approximate derivatives of ϕ by first order differences of vertices, e.g. $\partial_x \phi = \frac{1}{h}(v_1 - v_0) + O(h)$ and $\partial_y \phi = \frac{1}{h}(v_3 - v_0) + O(h)$. Actually, these differences properly express the derivatives of ϕ in edge midpoints, since $\partial_x \phi(m_e) = \frac{1}{h}(v_1 - v_0) + O(h^2)$, for $m_e = (h/2, 0)$. An analogous statement holds for ∂_y and the edge midpoint $(0, h/2)$. In order to achieve the higher approximation order $O(h^2)$ for both first derivatives in the *same* point, we resort to a trick and apply a change of coordinates. With $u = x + y$ and $v = x - y$, derivatives in the face midpoint $m_f = (h/2, h/2)$ are expressed as

$$\partial_u \phi(m_f) = \frac{v_2 - v_0}{\sqrt{2}h} + O(h^2), \ \partial_v \phi(m_f) = \frac{v_3 - v_1}{\sqrt{2}h} + O(h^2),$$

as can be seen from a Taylor expansion centered in m_f.

As to mappings from one mesh to another, consider meshes $M = \phi(h\mathbb{Z}^2)$ and $M' = \phi'(h\mathbb{Z}^2)$, so that the mapping $\psi : M \to M'$ obeys $\psi \circ \phi = \phi'$. With differentials $d\phi = (\partial_u \phi, \partial_v \phi)$ and $d\phi' = (\partial_u \phi', \partial_v \phi')$ we have $d\psi \circ d\phi = d\phi'$. This directly leads to the second order approximation

$$d\psi \cdot (v_2 - v_0, \ v_3 - v_1) = (v_2' - v_0', \ v_3' - v_1') + O(h^2),$$

which is valid in the face "midpoint" $\phi(m_f)$. In terms of edges of the inscribed parallelograms according to Figure 4, this relation between differentials is expressed as

$$d\psi \cdot (m_{12} - m_{01}, m_{30} - m_{01}) = (m_{12}' - m_{01}', m_{30}' - m_{01}') + O(h^2).$$

One interpretation of this equation is the following: ψ is isometric if and only if, up to an error of magnitude h^2, the inscribed parallelograms are isometric to each other.

This isometry of parallelograms is exactly our definition of a discrete-isometric mapping. It amounts to 3 scalar constraints per face. The higher approximation order is achieved because the set of constraints respects the mesh's combinatorial symmetries.

In analogy, ψ is a continuous isometry, if and only if everywhere on the surface the linear mapping $d\psi$ is isometric. Thus we expect that discrete-isometric mappings as we defined them faithfully approximate continuous isometries. This expectation has also been confirmed by experiments.

If we had insisted that the faces of the original mesh move rigidly, we would have imposed not 3, but 4 scalar constraints per face. Actually most quad meshes are rigid if faces are kept rigid, so requiring this would not have been a faithful discretization of isometric mapping. We are coming back to this topic in §5.2.1.

2.2 Computing Discrete Isometries with High Accuracy

Jiang et al. [2020] use optimization to compute isometries between meshes M, M'. We recap their method to the extent necessary to explain the improvements we make. The variables in the optimization

are the vertices of M', and the target functional to be minimized has the form $\lambda_{iso} E_{iso} + \lambda_{fair} E_{fair} + \dots$, where $\lambda_{iso}, \dots$ are weights and $E_{iso}, \dots$ are functionals penalizing deviation from the desired property.

2.2.1 Isometry Constraints. By definition, meshes M, M' are isometric if the parallelograms inscribed in corresponding faces $f = v_0 v_1 v_2 v_3$ and $f' = v_0' v_1' v_2' v_3'$ of M resp. M' are isometric, i.e., if and only if corresponding diagonals have the same lengths and enclose the same angle:

$$c_{iso,0}(f) = (v_0 - v_2)^2 - (v_0' - v_2')^2 = 0, \tag{1}$$
$$c_{iso,1}(f) = (v_1 - v_3)^2 - (v_1' - v_3')^2 = 0,$$
$$c_{iso,2}(f) = (v_0 - v_2) \cdot (v_1 - v_3) - (v_0' - v_2') \cdot (v_1' - v_3') = 0.$$

We thus define

$$E_{iso} = \sum_{f \in F} \sum_{j=0}^{2} c_{iso,j}(f)^2. \tag{2}$$

$E_{iso} = 0$ expresses the discrete-isometric property.

2.2.2 Fairness. Our discretization is based on the interpretation of edges and diagonals as first derivatives. This interpretation requires *fairness* of mesh polylines. Otherwise meshes might deform in a nonsmooth manner but still $E_{iso} = 0$, see Fig. 5. Jiang et al. [2020] use a fairness energy of the form

$$E_{fair, M'} = \sum_{\text{triples } i,j,k} \|\Delta_{ijk}'\|^2, \text{ with } \Delta_{ijk}' = v_i' - 2v_j' + v_k'. \tag{3}$$

Summation is over all triples v_i', v_j', v_k' of consecutive vertices of a mesh polyline in M'. If the mesh M is not fixed during optimization, an analogous energy $E_{fair, M}$ is used. This fairness energy is based on the interpretation of mesh polylines as discrete versions of curves $c(t)$ — the 2nd differences occurring in E_{fair} are discrete counterparts of 2nd derivatives $\frac{d^2}{dt^2} c$. As such they carry information on both the intrinsic (geodesic) curvature of mesh polylines within M', and the normal curvature of M' itself. The energy is not able to distinguish between the two. This has implications on our optimization procedures, since E_{fair} will not be able to achieve zero residual if the target shape M' is curved.

Tangential Fairness. A way to remedy this situation is to consider the tangential components of 2nd differences, which are discrete counterparts of *covariant* derivatives $\frac{D}{dt} \frac{d}{dt} c$, cf. [do Carmo 1976]. After estimating unit normal vectors n_i, n_i' in the vertices of M, M', we let $\Delta_{ijk}^{tang} = \Delta_{ijk} - \langle n_j, \Delta_{ijk} \rangle n_j$ and similar for $\Delta_{ijk}'^{tang}$.

It is well known that isometries respect covariant derivatives, i.e., we expect that $d\psi(\Delta_{ijk}^{tang}) = \Delta_{ijk}'^{tang}$ holds. We thus formulate a *tangential* fairness functional

$$E_{fair}^{tang} = \sum_{\text{triples } i,j,k} \|d\psi_j(\Delta_{ijk}^{tang}) - \Delta_{ijk}'^{tang}\|^2.$$
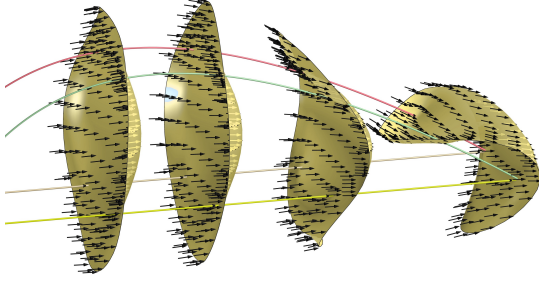
Fig. 6. *Infinitesimal isometries and continuous bending.* Here a surface undergoes time-continuous isometric bending such that four selected vertices move on prescribed trajectories. The computation involves i-vector fields.

Here $d\psi_j$ is a linearization of the isometry valid for the central vertex v_j. Like normal vectors, in our iterative procedures it is updated after each round of optimization. For all edges $v_j v_k$ we must have $d\psi_j(v_j - v_k) \approx v'_j - v'_k$. We also extend $d\psi_j$ to normal vectors by requiring $d\psi_j(n_j) \approx n'_j$. All these constraints are linear and can be written in the form $d\psi_j \cdot M_j = M'_j$ (the columns in matrices M_j, M'_j contain the vectors that are required to be mapped). This leads to the least-squares solution $d\psi_j = M'_j M_j^T (M_j M_j^T)^{-1}$.

2.2.3 The Optimization Procedure. Our procedure to compute isometric mappings is analogous to the one proposed by [Jiang et al. 2020], only we use E_{fair}^{tang} instead of E_{fair}. Therefore we do not describe it here. Furthermore, it is only a special case of the optimization procedure which in §4 is used for registration (in Equ. (10), let $\lambda_{orth} = 0$ and $\mathbf{w}' = 0$).

3 INFINITESIMAL ISOMETRIES OF MESHES

Vector fields representing infinitesimal isometries occur several times in our computations. Firstly a special case (Killing vector fields) is important in understanding the intrinsic geometry of surfaces. Secondly, we will use them for guiding isometries, in particular for local isometric registration (see §4). We start with describing the general case and then specialize to Killing vector fields.

3.1 Extrinsic Infinitesimal Isometries

Consider a time-continuous flexion $M(t)$ of a mesh M, where each vertex v_i moves on a smooth path and all meshes $M(t)$ are isometric in the sense that the parallelograms inscribed in faces move rigidly. Fig. 6 shows an example of such a continuous isometric bending. At a certain time $t = t_0$ of interest we consider the velocity vectors $w_i = \frac{dv_i}{dt}(t_0)$. The parallelograms inscribed in faces are required to move rigidly, implying that in the original mesh, for any face $f = (v_0 v_1 v_2 v_3)$, the diagonals' lengths and enclosed angle remain unchanged. Thus,

$$\|v_0 - v_2\|^2, \quad \|v_1 - v_2\|^2, \quad \langle v_0 - v_2, v_1 - v_3 \rangle$$

remain constant, and time-derivatives $w_i = \frac{d}{dt} v_i$ obey

$$\langle v_0 - v_2, \ w_0 - w_2 \rangle = 0, \quad \langle v_1 - v_3, \ w_1 - w_3 \rangle = 0,$$
$$\langle v_0 - v_2, \ w_1 - w_3 \rangle + \langle w_0 - w_2, \ v_1 - v_3 \rangle = 0. \tag{4}$$

3.1.1 Infinitesimal Isometries: i-Velocity Fields and r-Velocity Fields. We call any vector field w obeying (4) an i-velocity vector field. The linear space of i-fields has dimension ≥ 6, because it always contains the so-called r-velocity fields which are the time-derivatives of rigid body motions. These have the form

$$w_i = c \times v_i + \bar{c}, \tag{5}$$

where $c, \bar{c} \in \mathbb{R}^3$, with c indicating the vector of angular velocity. See Fig. 6 for a visualization of i-fields.

Remark 1. In most of the literature, i-velocity fields are called infinitesimal isometries, or *infinitesimal flexions* of the hinge-and-body mechanism formed by rigid parallelograms connected with spherical joints. It is known that generically every i-field is generated as time-derivative of an actual time-continuous isometry. The configuration space of isometries of the mechanism is algebraic and generically its dimension equals the dimension of the space of i-fields. Exceptions are possible for certain special vertex positions, see the survey [Connelly 1987].

Remark 2. Equ. (4) is symmetric in the variables v_j and w_j, and their roles could be exchanged. Analogously, for an isometric bending of a smooth surface $x(u, v)$ with time derivative $w(u, v) = \frac{d}{dt} x(u, v)$ we have $\langle x_u, w_u \rangle = \langle x_v, w_v \rangle = \langle w_u, x_v \rangle + \langle x_u, w_v \rangle = 0$. This is called the orthogonality relation between a surface and its infinitesimal isometry, and Equ. (4) is its discrete counterpart.

Remark 3. A parallelogram inscribed in a face f_i of the mesh moves rigidly, so all edge midpoints $m_{kl} = \frac{1}{2}(v_k + v_l)$ of f have velocity vectors that fit a certain r-field of the kind given by Equ. (5): There are $c_i, \bar{c}_i \in \mathbb{R}^3$ with $\frac{1}{2}(w_k + w_l) = c_i \times m_{kl} + \bar{c}_i$ whenever $v_k v_l$ is an edge of f_i. R-fields associated with faces are not independent, as

$$v_k v_l = f_i \cap f_j \implies c_i \times m_{kl} + \bar{c}_i = c_j \times m_{kl} + \bar{c}_j.$$

3.2 Intrinsic Infinitesimal Isometries (Killing Vector Fields)

3.2.1 Definition and Properties of Killing Vector Fields. A surface Φ may admit time-continuous isometries within itself, or even a rigid body motion within itself, e.g. if it is a complete surface of revolution. The time-derivatives of such isometries are i-fields tangential to Φ, or r-fields tangential to Φ. Also a kind of converse statement is true: The flow of a tangential i-field (a *Killing vector field*) moves
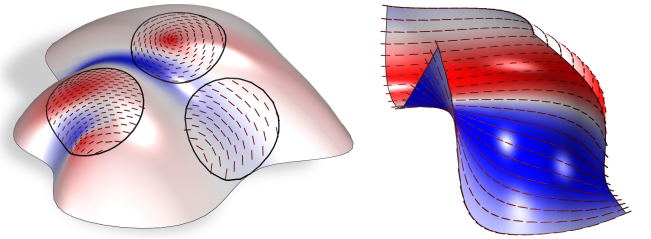

Fig. 7. Infinitesimal isometries tangential to surfaces (Killing fields) exist on surfaces isometric to rotational surfaces like the example at right. Otherwise vector fields that are as Killing as possible yield information on paneling. The color coding is according to Gaussian curvature — true Killing fields must follow its level sets.

any piece of Φ within Φ in an isometric way until the boundary is reached. For an r-field, this motion is even a rigid body motion.

Within Geometry Processing, several contributions deal with the computation and applications of Killing vector fields, see e.g. [Ben-Chen et al. 2010]. One notable feature of Killing fields is that they are always tangential to the isolines of Gaussian curvature K (which follows directly from the fact that Gaussian curvature is invariant under isometries, see Fig. 7). The Killing fields of a surface Φ constitute a linear space $\mathcal{K}_\Phi$ which reveals geometric information [Myers 1936]:

- $\dim \mathcal{K}_\Phi \geq 1$: Simply connected pieces of Φ can be isometrically mapped to a rotational surface, possibly with overlaps. This deformation transforms the Killing field to an r-field tangential to parallel circles. Its orthogonal trajectories are mapped to meridian curves, see Figures 7 and 10.
- $\dim \mathcal{K}_\Phi \geq 2$: Gaussian curvature K is constant. Simply connected pieces of Φ can be mapped isometrically, possibly with overlaps, to a plane if $K = 0$, to a sphere if $K > 0$, and to a pseudosphere if $K < 0$. If Φ is simply connected, we have $\dim \mathcal{K}_\Phi = 3$, see Fig. 8.

One should be aware that topological features of surfaces typically restrict existence of Killing fields. To obtain local information it is therefore advisable to make surfaces simply connected by cutting them open before computing Killings fields on them. We are going to use Killing vector fields to find an isometry or near-isometry of a surface to a surface of revolution, thus enabling paneling with a small number of molds (see §5.1.2 and Fig. 10).

3.2.2 Computing Near-Killing Vector Fields. Vector fields can be approximately Killing in different ways: An i-vector field might be only approximately tangential; or a tangential vector field might satisfy the constraints (4) only approximately. The following optimization procedure provides access to both versions.

Assume the surface of interest is given by a quad mesh (V, E, F), and in each vertex v_i, $i = 1, \ldots, |V|$ a unit normal vector n_i has been estimated. We let $\mathbf{n} = (n_1, n_2, \ldots) \in \mathbb{R}^{3|V|}$. A Killing vector field $\mathbf{w} = (w_1, w_2, \ldots) \in \mathbb{R}^{3|V|}$ consists of a vector w_i per vertex. It is subject to three constraints: Firstly, $\mathbf{w}$ is tangential, i.e.,

$$E_{tang}(\mathbf{w}) = \sum_i \langle w_i, n_i \rangle^2 = \mathbf{w}^T (\mathbf{n} \mathbf{n}^T) \mathbf{w} = 0.$$

Secondly, it obeys the orthogonality constraint (4), which can be written as $C\mathbf{w} = 0$, with a matrix $C \in \mathbb{R}^{3|F| \times 3|V|}$. Equivalently,

$$E_{orth}(\mathbf{w}) = (C\mathbf{w})^T (C\mathbf{w}) = 0. \quad (6)$$

Thirdly, a fairness energy of the form (3) is used, which is quadratic and can therefore be expressed as $E_{fair}(\mathbf{w}) = \mathbf{w}^T M \mathbf{w}$. Summing up,
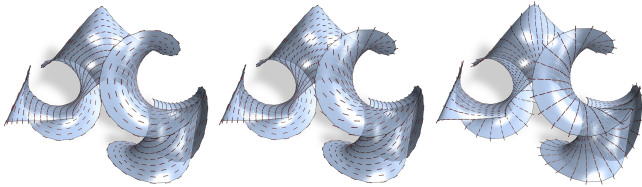
we require $\mathbf{w}$ to minimize $E_{Killing} = E_{tang} + \lambda_{orth} E_{orth} + \lambda_{fair} E_{fair}$:

$$E_{Killing}(\mathbf{w}) = \mathbf{w}^T K \mathbf{w} \to \min, \ K = \mathbf{n}\mathbf{n}^T + \lambda_{orth} C^T C + \lambda_{fair} M, \quad (7)$$

under the normalization constraint $\|\mathbf{w}\| = 1$. The parameter λ_{orth} is used to adjust the relative weight of constraints, while λ_{fair} is small. The solution consists of an eigenvector of K corresponding to the smallest eigenvalue. This procedure is very similar to the computation of r-vector fields by [Pottmann et al. 2001]. The parameter λ_{orth} in (7) determines whether we want more emphasis on the tangential property or the i-vector field property. §5.1.2 treats the more difficult problem of modifying a mesh so that it has an exact Killing field.

3.2.3 Computing the Dimension of $\mathcal{K}_\Phi$. It may happen that the matrix K of (7) has more than one small eigenvalue. The meaning of 'small' has to be clarified: For any normalized eigenvector $\mathbf{w}$ corresponding to the eigenvalue ω we have $E_{Killing}(\mathbf{w}) = \mathbf{w}^T K \mathbf{w} = \omega \mathbf{w}^T \mathbf{w} = \omega$. So ω is small if, seen as an energy expressing the Killing property, it is small. The occurrence of two or three such small eigenvalues is evidence for $\dim \mathcal{K}_\Phi \geq 2$, which can only happen if the Gaussian curvature is constant. Our numerical experiments confirmed this relation. Figures 7 and 10 show the case of a single smallest eigenvalue, while the example of Fig. 8 has three.

Remark 4. There is an alternative to the energy defined by (7) that does not require estimating normal vectors. Suppose $f = (v_p, v_q, v_r, v_s)$ is a face, and the inscribed parallelogram has normal vector n_f which is thought to be valid for the center $m_f = \frac{1}{4}(v_p + v_q + v_r + v_s)$. The vector field assigns to m_f the vector $\frac{1}{4}(w_p + w_q + w_r + w_s)$, and tangency is expressed by $E_{tang} = \sum_{f=(v_p, v_q, v_r, v_s)} \langle w_p + w_q + w_r + w_s, n_f \rangle^2 \to \min$. This modified energy E_{tang} yields a modified energy $E_{Killing}$.

4 ISOMETRIC REGISTRATION

For the purpose of paneling, we treat the following problem. Given are two surfaces Φ_1 (a panel) and Φ_2 (the mold as target surface). Isometrically deform Φ_1 so that the resulting deformed surface Φ_1' matches Φ_2. We do not require alignment of boundaries here. This process is called nonrigid, isometric registration. Algorithmically, one has to solve for an initial alignment first — this is a global registration problem and will depend on the specific application. In a second step one fine-tunes the alignment by numerical optimization — this is a local registration problem.

For local registration, we will be able to work within the framework of i-vector fields presented in §3. The surface Φ_1 is represented as a quad mesh M_1, while Φ_2 is represented in any way which allows us to compute closest points and distances.

The Distance Field of a Surface. We here build on prior work by Pottmann et al. [2006] on rigid registration. For simplifying the distance field of the target surface, we use the following property: If a point y is not far from its closest point $y^* \in \Phi_2$ (not far compared to curvature radii of Φ_2), then locally around y we may replace the squared distance from the target by the squared distance from the

Fig. 8. For the surface of constant Gaussian curvature in Fig. 3, the space $\mathcal{K}_\Phi$ of Killing vector fields has dimension 3. These images show a basis.
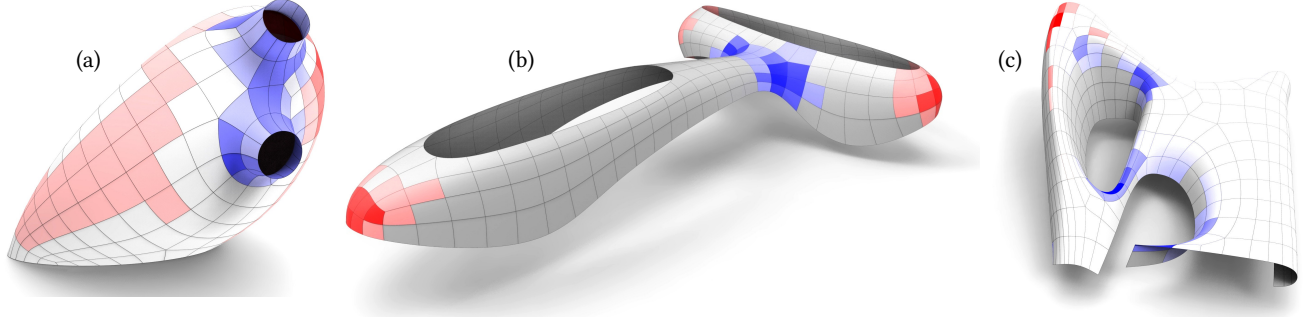
Fig. 9. *Paneling with surfaces of constant Gaussian curvature K.* 5 positive curvature molds corresponding to red panels and negative curvature molds corresponding to blue panels are used. The remaining panels are developable and correspond to one single flat virtual mold. The color coding indicates the value of Gaussian curvature. Subfigures (b) and (c) show actual architectural freeform skins, namely a 2009 glass roof by Asymptote Architecture in Abu Dhabi and a design by Zaha Hadid architects realized 2012 in concrete in Baku. See detailed statistics in Fig. 17.

tangent plane in y^*:

$$d(x, \Phi_2)^2 \approx \langle x - y^*, n^* \rangle^2, \quad \text{where } n^* = \frac{y - y^*}{\|y - y^*\|}. \quad (8)$$

The vector n^* is the normal vector of Φ_2 in the closest point y^*. This approximation is a 2nd order Taylor approximation in case that y actually lies in Φ_2. We point out that the well known ICP algorithm works on the basis $d(x, \Phi_2) \approx \|x - y^*\|$, which is a good approximation only far away from the target [Pottmann et al. 2006].

When the approximation (8) is used during iterative optimization, closest points and normal vectors are recomputed before every round of optimization.

Registration by Optimization. The local registration procedure itself is based on iterative optimization. Its aim is to find a mesh M'_1 isometric to the given mesh M_1 (the panel), but close to the target Φ_2 (the mold). We even allow the panel to change a little in the process so as to better fit the mold. Conceptually, we linearize the evolution of the mesh by computing an i-vector field and move vertices in this direction. Isometry is destroyed by this procedure (but not by much) and has to be restored using optimization. These two steps are combined into one, as described below.

We initialize by $M'_1 = M_1$ and update M'_1 in every round of optimization. Collecting the vertices of M'_1 in $\mathbf{v}' = (v'_1, \ldots) \in \mathbb{R}^{3 \times |V|}$, the update is via $\mathbf{v}' \leftarrow \mathbf{v}' + \mathbf{w}'$, where $\mathbf{w}'$ is an i-vector field minimizing the energy E_{orth} according to Equ. (6). This vector field linearizes the isometric evolution of M'_1 over time.

Based on (8), proximity of the updated mesh M'_1 to the target is governed by the penalty term

$$E^{mold}_{close} = \sum_i \langle v'_i + w'_i - v^*_i, n^*_i \rangle^2 + \varepsilon \sum_i \|v'_i + w'_i - v^*_i\|^2. \quad (9)$$

It is formulated in terms of closest point projections v^*_i of vertices v'_i onto the target. It includes a regularizer with small weight ε which might not be necessary, depending on the application. If the original panel M_1 is also a variable, an analogous energy E^{panel}_{close} penalizes deviation of M_1 from the reference surface Φ_1.

The energy E_{iso} of (2) expresses isometry of M_1 and the updated mesh M'_1, if applied to $\mathbf{v}' + \mathbf{w}'$. Combining the constraints listed so far with fairness, we create an energy which is to be minimized:

$$E_{reg} = \lambda_{iso} E_{iso}(\mathbf{v}' + \mathbf{w}') + \lambda^{panel}_{close} E^{panel}_{close} + \lambda^{mold}_{close} E^{mold}_{close} +$$
$$+ \lambda_{orth} E_{orth}(\mathbf{w}') + \lambda_{fair} E_{fair} \rightarrow \min. \quad (10)$$

The variables in this optimization are the vector field $\mathbf{w}'$ and, depending on the application, also the vertices of the original panel M_1. Fairness applies to all meshes that are variables, namely M'_1 and possibly also M_1. Therefore we do not use the fairness energy E^{tang}_{fair} (which is based on a comparison of these two meshes), but stick with E_{fair}.

After updating we repeat until either the optimization goal or the maximum number of iterations has been reached. A summary of the procedure is given by Alg. 1. We found that for our main application (paneling) it was not necessary to change the weights occurring in (10).

Isometric Registration up to Scale. It turned out that a very successful method of paneling is to use surfaces of constant Gaussian curvature K, see e.g. Fig. 9. We found it useful to have at our disposal a way to match a given surface patch with a sphere or pseudosphere, but where the value of K is still a variable. This is done by deciding the sign of K first, and subsequently registering to the unit sphere, or unit pseudosphere, in a way that is isometric up to scale.

Algorithm 1: Isometrically register mesh M_1 to target surface Φ_2.

Data: Mesh M_1 with vertices $\mathbf{v}$

$\mathbf{v}' \leftarrow \mathbf{v}$;

Fix energy threshold $E_{reg,\min}$ and maximum iterations;

Choose weights ε, λ_{orth}, λ_{fair}, λ^{panel}_{close}, λ^{mold}_{close} for (9) and (10);

repeat

    For all vertices v_i, compute closest point $v^*_i \in \Phi_2$;

    Estimate unit normal vectors $n^*_i = (v_i - v^*_i) / \|v_i - v^*_i\|$;

    Determine $\mathbf{v}$, $\mathbf{w}'$ such that $E_{reg} \rightarrow \min$;

    Update $\mathbf{v}' \leftarrow \mathbf{v}' + \mathbf{w}'$;

    Update weights;

until $E_{reg} \le E_{reg,\min}$ *or max. iterations reached*;

return *Vertex coordinates* $\mathbf{v}$, $\mathbf{v}'$

For that we modify Alg. 1, by replacing the energy E_{iso} by $E_{iso}^\kappa = \sum_f \sum_{j=0}^2 c_{iso,j}^\kappa(f)$. The constraints $c_{iso,j}^\kappa$ are based on the isometry constraints (1) and express isometry up to scaling by a factor κ:

$$c_{iso,0}^\kappa(f) = \kappa^2 \|v_0 - v_2\|^2 - \|v_0' - v_2'\|^2 = 0,$$
$$c_{iso,1}^\kappa(f) = \kappa^2 \|v_1 - v_3\|^2 - \|v_1' - v_3'\|^2 = 0,$$
$$c_{iso,2}^\kappa(f) = \kappa^2 \langle v_0 - v_2, \ v_1 - v_3 \rangle - \langle v_0' - v_2', \ v_1' - v_3' \rangle = 0.$$

Since κ^2 enters these constraints linearly, we use κ^2 as a variable. If meshes M, M' obey these constraints and M represents the unit sphere, then M' is isometric to a sphere of radius $1/\kappa$ and enjoys Gaussian curvature $K = \kappa^2$. In the pseudosphere case, $K = -\kappa^2$.

5 APPLICATIONS

5.1 Isometric Paneling

The main motivation for our work is paneling of architectural freeform skins by materials that allow isometric bending (sheet metal, certain plastics, and textiles). We start from a given panel layout on a surface and compute a set of molds for it, so that this set is as small as possible and the predefined panels can be manufactured, up to tolerances, by these molds. Quality criteria include deviation from the reference surface, and gaps and kink angles at seams. A second task is to guide the panel layout by geometric information. Note however that the panel layout, if at all visible on the finished architectural design, in a real-world application is going to be a design decision.

5.1.1 Paneling Algorithm for Constant Gaussian Curvature Molds. In the course of our work on the paneling problem we were surprised to find out to which extent it is possible to achieve a smooth paneling of freeform shapes by surfaces of constant Gaussian curvature. Figures 1, 9a–c, 15, and 19 show results obtained by this method. The panelization procedure starts with a given reference surface plus panel boundary layout, and it consists of the following steps.

- *Intrinsic patch analysis.* For this application, this means computing Gaussian curvature K, for which we use the method of osculating jets [Cazals and Pouget 2003].
- *Clustering.* This means assigning a type of mold to each panel. Panels where the Gaussian curvature does not exceed a threshold are declared intrinsically flat, exhibiting zero Gaussian curvature. On the other panels we employ k-means clustering on the average value of K of the panel, using the algorithm of [Lloyd 1982]. This leads to a small number of clusters centered around values K_i (we simply chose to use 5 clusters). K_i is the Gaussian curvature of the mold assigned to all panels in that cluster.
- *Construction of molds.* For each nonzero value K_i we construct a mold by scaling the unit sphere resp. pseudosphere with the factor $|K_i|^{-1/2}$. It is very convenient to have spheres as molds since the distance field is known. Similarly, the pseudospheres have rotational symmetry and their distance field is manageable. The third kind of mold is a flat plane, it is associated with zero Gaussian curvature.
- *Local panel optimization.* For a first alignment between panels and molds, rigidly move molds of Gaussian curvature K_i to each panel contained in the corresponding cluster. This global registration

task in case $K_i > 0$ consists of a standard least square fitting of a sphere of known radius to the panel. In case $K_i = 0$ we use PCA to fit a plane to the panel. In case $K_i < 0$ we find a pseudospherical piece of the same size as the panel and likewise use PCA for the first alignment. We subsequently for each panel M_j find an isometric panel M_j' that is close to the mold which currently is positioned close to M_j. This local registration procedure is initialized as $M_j' = M_j$ and is performed according to §4. In order to achieve a better fit between the design surface and the molds we extend the registration procedure by considering also the vertices of the original panels as variables except for the boundaries which are kept fixed.

- *Final optimization.* While this was not necessary for most of our examples, in order to achieve a smoother appearance (smaller kink angles at panel boundaries) one can perform a few steps of optimization of several panels (or even all panels) simultaneously. The target functional of optimization is the sum of the ones used in the local panel optimization above, cf. Fig. 14.

5.1.2 Paneling Algorithm Based on Rotational Surfaces. Paneling a surface of revolution Ψ needs only a small number of molds – choose molds as pieces of Ψ along a meridian and ensure that each panel can be rotated into at least one mold (see Fig. 10d). Any surface Φ admitting a near-Killing vector field can be modified in small ways to become isometric to a surface of revolution Ψ, so also Φ has a paneling that needs only a small number of molds. Figure 10 illustrates the procedure we apply to achieve this modification.

We start with the cross field determined by the near-Killing vector field and vectors orthogonal to it. We use the *libigl* implementation of mixed-integer quadrangulation [Bommes et al. 2009; Jacobson et al. 2018] to find a quad mesh M aligned with this cross field, see Fig. 10b. M becomes exactly isometric to a rotational surface, if any pair of faces that are neighbours in the Killing field direction become
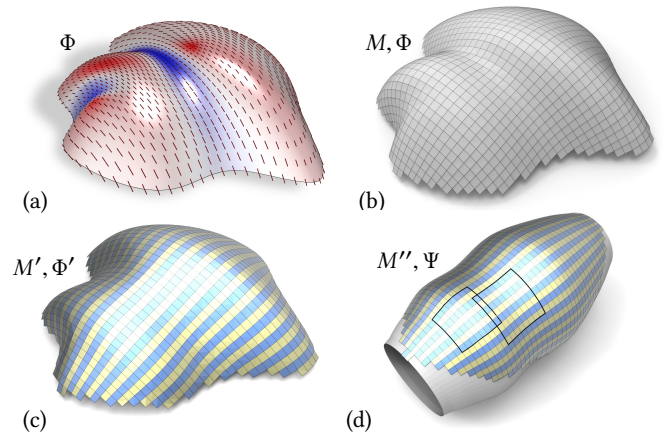

Fig. 10. We use near-Killing vector fields as the one in (a) to find a rotational surface Ψ isometric to a given surface Φ. Remeshing with edges are either tangential or orthogonal to the field yields the mesh (b). It is optimized so that quads that are neighbours in the Killing field direction become isometric. This yields the mesh Ψ' in (c) which is almost the same shape as (b) because the vector field was almost Killing. Mapping mesh polylines to parallel circles and meridians, we create an isometric rotational surface (d).
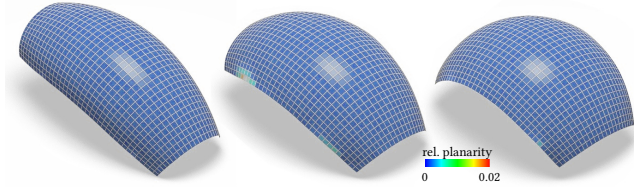
Fig. 11. A flexible quad mesh with planar faces. The design of such highly constrained objects is very interesting from the differential-geometric viewpoint. The color coding indicates the measure of nonplanarity, which is the distance of diagonals in a face divided by average edge length. Its average is about $7 \cdot 10^{-4}$. Isometry is very accurate (the edge length error according to the statistics in Fig. 17 does not exceed $5 \cdot 16^{-6}$).

isometric. These constraints are expressed analogous to Equ. (1) and are met by way of optimization. The result is a mesh M' (Fig. 10c) isometric to a mesh M'' with rotational symmetry (Fig. 10d). The construction of M'' is elementary and is described e.g. by [Wang et al. 2019]; it is unique after choice of an initial value.

5.2 Designing and Actuating Isometries

5.2.1 Flexible Quad Meshes with Planar Faces.
Certain special quad meshes with planar faces admit flexions where all faces move as rigid bodies and edges act as hinges. They have been a topic of interest already in the earlier works on discrete differential geometry [Sauer 1970]. In case of regular combinatorics, counting degrees of freedom reveals that flexibility is equivalent to flexibility of each 3 by 3 submesh, but the classification of flexible 3×3 meshes is rather involved [Izmestiev 2017].

It is remarkable that such highly constrained structures have already been realized as an actual transformable design: we point to the *Kinematic Sculpture* exhibit by S.O.M. at the 2018 Chicago Design week [Baldwin 2018]. So far no algorithmic treatment was available. We are able to numerically construct such meshes, as follows. Consider a quad mesh which enjoys fairness so that edges can be interpreted as derivatives and the mesh is a discrete version of a smooth surface Φ. Planar faces mean that edges are discrete versions of *conjugate* tangent vectors w_1, w_2, i.e., they obey $\mathrm{II}(w_1, w_2) = 0$ [Liu et al. 2006]. Here II is the second fundamental form of the surface [do Carmo 1976]. A flexion with rigid faces thus is a discrete counterpart of an isometric deformation $\psi : \Phi \to \Phi'$ where conjugate vectors w_j are mapped to conjugate vectors $w_j' = d\psi(w_j)$, i.e.,
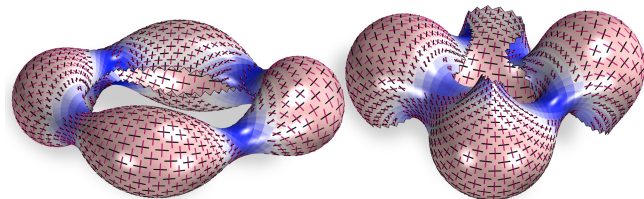

Fig. 12. Local analysis of isometries. We investigate the ratio $\kappa_n(v)/\kappa_n'(v')$ of normal curvatures before and after an isometric deformation and visualize the cross field of directions where it is extremal. This is geometrically meaningful in areas of positive Gaussian curvature.

we have $\mathrm{II}'(w_1', w_2') = 0$. Linear algebra tells us that in each point, w_1, w_2 can be uniquely found as the generalized eigenvectors corresponding to the generalized eigenvalue problem $\det(\mathrm{II}' - \lambda \mathrm{II}) = 0$ (where tangent planes of Φ, Φ' are identified via $d\psi$). Existence of a solution depends on the sign of $\det \mathrm{II}$, i.e., the sign of Gaussian curvature: If $K < 0$ there may be no solution; if $K \geq 0$ solutions exist; if $K > 0$ they are unique unless $\mathrm{II}, \mathrm{II}'$ happen to be multiples of each other.

This leads to the following procedure for the *design of flexible quad meshes with planar faces*. For a given isometric deformation $\psi : \Phi \to \Phi'$, vectors v, w and their images v', w' define cross fields on Φ, Φ'. We use the *libigl* implementation of mixed-integer quadrangulation [Bommes et al. 2009; Jacobson et al. 2018] to find quad meshes M, M' aligned with them. We also initialize intermediate positions M_t for selected values $t \in [0, 1]$ by linearly interpolating between $M_0 = M$ and $M_1 = M'$ (this crude method turned out to be sufficient for our examples). Subsequent optimization for isometry follows the principle of §2 with the difference that the isometry constraints now refer to faces moving rigidly. Thus the constraints of (1) are replaced by $\|v_i - v_j\|^2 - \|v_i' - v_j'\|^2 = 0$ for all pairs i, j where vertices v_i, v_j are contained in the same face. A result is shown by Fig. 11.

Remark 5. An easily understood class of flexible quad meshes are discrete *Voss surfaces* where mesh polylines enjoy a discrete geodesic property [Sauer 1970]. In each vertex the four angles $\omega_1, \ldots, \omega_4$ of faces obey $\omega_1 - \omega_3 = \omega_2 - \omega_4 = 0$. For Fig. 11, average and maximum of these differences equal $1.3°$ resp $10°$. This example therefore is not a Voss surface.

Remark 6. When trying to understand the isometric bending ψ of a surface Φ into another shape $\Phi' = \psi(\Phi)$, we might ask for directions where the normal curvature changes the most, or changes the least. This normal curvature in direction of a unit tangent vector is expressed as $\mathrm{II}(v, v)$. With $v' = d\psi(v)$ the normal curvature after bending is $\mathrm{II}'(v', v')$. When we ask for extremal values of the quotient, we are led to the very same generalized eigenvalue problem as above. Fig. 12 shows an example. Note that asking for extremal values does not make sense in case of negative Gaussian curvature, because if v' points in an asymptotic direction, we have $\mathrm{II}(v', v') = 0$ and the ratio will be infinite [do Carmo 1976, p. 148ff].

5.2.2 Isometric Bending Guided by Trajectories.
Figure 6 shows the time-continuous isometric bending of a surface such that selected vertices $v_{i_k}, k = 1, \ldots, r$ run on prescribed trajectories T_{i_k}. This is simulated by computing an i-vector field $\mathbf{w}$ where w_{i_k} is tangential to T_{i_k}, updating vertices by $v_i \leftarrow v_i + w_i$ and restoring isometry to the mesh we started with by invoking optimization according to §2. The movement of vertices along the prescribed trajectories is ensured by penalizing distances by an energy E_{close} analogous to (9). This small example illustrates the many degrees of freedom that are present in isometric bending.

5.2.3 Continuous Folding of Doubly-Curved Surfaces.
Geometric folding has produced a vast amount of literature. Specifically the folding of flat sheets along curves has been treated from the viewpoint of geometry processing only more recently. Double-curved smooth surfaces likewise admit isometric deformations to surfaces
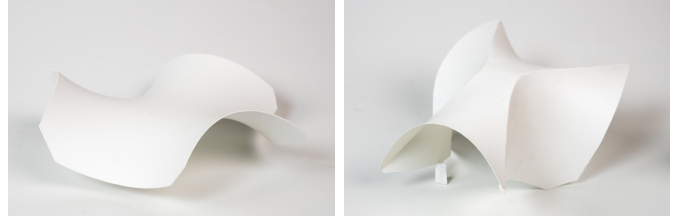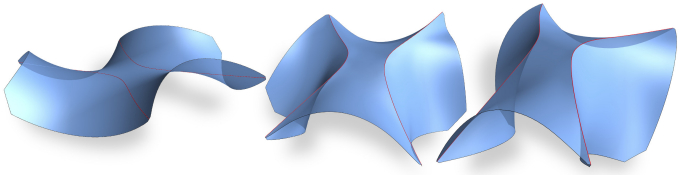
Fig. 13. *Time-continuous curved folding.* In contrast to flat sheets, time-continuous curved folding is impossible for positive Gaussian curvature and can happen only along asymptotic curves in case of negative Gaussian curvature. This example shows a smooth surface folding along two curves. It is an A-net exhibiting planar vertex stars, where all mesh polylines are asymptotic, cf. [Bobenko and Suris 2008]. The photos at right show both the original surface and the result of folding fabricated in polystyrene.

that are only piecewise smooth. Interestingly, time-continuous deformations of this kind are rather restricted. The reason for this is a well known geometric property occurring at the foldline. The foldline's osculating plane must be a local supporting plane [Rabinovich et al. 2019], which in the moment we start folding, where the surface is still smooth, means that the osculating plane equals the tangent plane. Unfortunately in non-flat surfaces such curves are rare. In case Gaussian curvature K is positive there are none, for developable surfaces there are only the rulings. In the case $K < 0$ we have the network of *asymptotic curves*, with two curves passing through each point. Curved folding is only possible along these asymptotic curves. Fig. 13 shows an example. We simulated folding by computing isometries under the side-condition that two selected points move towards each other, while not enforcing fairness along selected foldlines.

6 DISCUSSION

Implementation and Properties of the Optimization. For optimization, we employ a Levenberg-Marquardt method according to [Madsen et al. 2004, §3.2], using a damping parameter of 10^{-6}, but we could just as well have used other optimization methods. Our implementation in C++ uses the data structures of *OpenMesh* [Botsch et al. 2002] and the Taucs library for sparse linear solvers [Toledo 2003]. Detailed statistics on panelization examples are provided by the table in Fig. 17. The computation times refer to an Intel Xeon E5-2687W 3.0GHz processor without parallel processing or other acceleration techniques. The result can be summarized by saying that registering a single panel with 1024 vertices takes about 0.3 to 0.4 seconds. In comparison, computing an isometric mapping according to §2 takes only about 20% of that time.
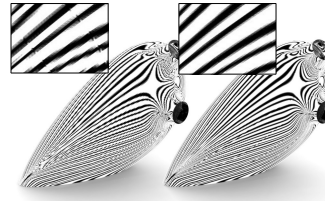

Fig. 14. Reflection lines reveal small tangent discontinuities in the result of Fig. 9a (left). Simultaneous optimization of several panels improves the surface quality (right).

The panelization procedure is essentially local; each panel is registered and optimized to a mold separately. We found that we can achieve a surface quality sufficient for applications in this way. However, a final round of global optimization, as described in the last paragraph of §5.1.1, can still improve surface quality. Fig. 14 shows an example where panels are optimized together with their immediate neighbourhood. Such a procedure is not difficult but rather time-consuming.

Convergence. We also want to add a comment on convergence. Convergence is not guaranteed, as is usual in nonlinear and nonconvex situations like ours, so a good initialization is very important. Our experiments revealed the optimization problems we solved as rather benign, with typically 5-10 iterations necessary to achieve convergence.

As to the *order* of convergence, in particular of optimization of isometries according to §2, we replaced the energy E_{fair} by E_{fair}^{tang} not just because of geometric considerations. This was done also because we were concerned about E_{fair} witnessing extrinsic curvature and thus never achieving zero residual. E_{fair}^{tang} can do that but in practice does not. In consequence, the order of convergence was not improved from one to two by switching to E_{fair}^{tang}. This switch nevertheless improves the behaviour of optimization. This is because
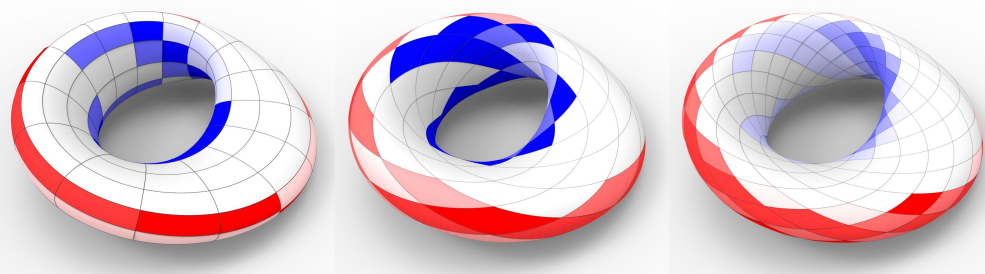

Fig. 15. Changing the panel layout typically hardly influences the approximation quality of the paneling, but increasing the number of panels does so in a significant way. The statistics of Fig. 17 suggest higher than linear approximation order.
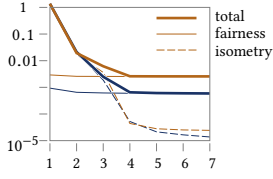
Fig. 16. A typical logarithmic plot of energies expressing fairness and isometry as iterative optimization performs 7 rounds. The orange plot corresponds to using E_{fair}, the blue plot to tangential fairness E_{fair}^{tang}.

E_{fair}^{tang} regularizes like E_{fair} while having significantly smaller value, thus not dominating other energies as easily, see Fig. 16.

Accuracy. We distinguish two kinds of accuracy, one being the quality of approximation of a reference surface by the paneling. It is expressed by the distances $\delta_{...}$ in Fig. 17. Examples like the one of Fig. 15 suggest that those distances behave like $N^{-\alpha}$, where N is the number of panels and $\alpha > 1$. Unfortunately also computation time increases faster than linearly. Our experiments, e.g. by comparing Figs. 15a with 15b further show that the concrete panel layout has little influence on approximation quality. Secondly we measure the quality of the isometry by the change of intrinsic distances. Constant distances are expressed by smallness of the L-error defined in Fig. 17, and by the success of practical examples.

Practical Feasibility. The materials and processes employed in freeform architecture do not scale well to the kind of small models we are able to build for experimental verification. We nevertheless checked the feasibility of our paneling procedure by means of molds created by CNC machining and panels made from high impact polystyrene. Molds and panels are shown by Figures 18, 19.

Limitations. Our method of representing isometries has some limitations. E.g. the importance of fairness makes it unsuitable for

Fig.	1	9a	9b	9c	15a	15b	15c	19
# panels	190	295	302	408	165	144	330	9
$K = 0$	91	117	221	320	104	76	168	4
$K > 0$	56	113	52	16	35	39	93	2
$K < 0$	43	65	29	72	26	29	69	3
# molds	11	11	11	11	11	11	11	3
T [sec]	52	115	108	148	54	48	231	3.2
$\delta_{panel,max}$	$3.5 \cdot 10^{-3}$	$2.5 \cdot 10^{-2}$	$1.9 \cdot 10^{-2}$	$2.3 \cdot 10^{-2}$	$4.5 \cdot 10^{-3}$	$4.4 \cdot 10^{-3}$	$4.9 \cdot 10^{-5}$	$1.0 \cdot 10^{-2}$
$\delta_{panel,avg}$	$2.0 \cdot 10^{-4}$	$3.3 \cdot 10^{-4}$	$2.0 \cdot 10^{-3}$	$3.6 \cdot 10^{-4}$	$2.5 \cdot 10^{-5}$	$1.8 \cdot 10^{-5}$	$2.9 \cdot 10^{-6}$	$7.4 \cdot 10^{-4}$
$\delta_{mold,max}$	$1.2 \cdot 10^{-3}$	$3.7 \cdot 10^{-3}$	$5.1 \cdot 10^{-3}$	$9.1 \cdot 10^{-3}$	$1.1 \cdot 10^{-3}$	$7.0 \cdot 10^{-4}$	$2.2 \cdot 10^{-4}$	$8.8 \cdot 10^{-4}$
$\delta_{mold,avg}$	$1.3 \cdot 10^{-4}$	$2.9 \cdot 10^{-4}$	$2.2 \cdot 10^{-4}$	$2.4 \cdot 10^{-4}$	$2.5 \cdot 10^{-5}$	$4.1 \cdot 10^{-5}$	$1.7 \cdot 10^{-5}$	$8.9 \cdot 10^{-5}$
L-err.	$1.5 \cdot 10^{-3}$	$2.5 \cdot 10^{-4}$	$4.6 \cdot 10^{-4}$	$7.4 \cdot 10^{-4}$	$1.4 \cdot 10^{-3}$	$1.6 \cdot 10^{-3}$	$3.9 \cdot 10^{-4}$	$6.2 \cdot 10^{-4}$

Weights: $\varepsilon = 0$, $\lambda_{fair} = \lambda_{close}^{panel} = 0.01$, $\lambda_{close}^{mold} = \lambda_{orth} = 0.1$, $\lambda_{iso} = 1$.

Fig. 17. Optimization statistics for nonrigid isometric registration according to Alg. 1. For each of the freeform design surfaces referred to in the first line we give the number of patches (of 32×32=1024 vertices each), the number of patches which enjoy zero, positive, and negative Gaussian curvature, and the number of molds. The number of iterations of our optimization procedure is 10 throughout, the time (in seconds) is shown in the table. The data verifying the quality of the result are the distance between patches and design surface, the distance between isometrically deformed patches and mold surface. Distances are normalized such that a patch bounding box has length 1 in average. The last number is the L^2 error of edge lengths, defined as $\|L' - L\|_2 / \|L\|_2$, where L, L' refer to the vector of all edge lengths occurring in panels, resp. deformed panels.

Fig. 18. *Paneling free forms.* The rendering at left shows a covering of the design surface by panels categorized according to Gaussian curvature K. We fabricated them by minimally altering the shape via optimization so that we need only two molds (at right): a spherical mold corresponding to 2 panels exhibiting $K > 0$; a mold with constant negative Gaussian curvature corresponding to 3 panels; and a virtual flat "mold" corresponding to 4 developable panels. See Fig. 19 for the final result.

modeling wrinkles and creases that might occur in isometric bending. This might lead to unrealistic behaviour in applications like the one shown in Fig. 6. However this is not an issue with paneling.

Future Research. Several directions of research can build on our work. Especially interesting are those having to do with continuous flexions. Flexible meshes with rigid faces have been a source of deep Mathematics and also actual transformable designs, but have hardly been explored algorithmically. Also mechanisms consisting of nonrigid parts are an interesting direction, maybe in connection with bionic technology. We may even go one step further and leave the domain of isometric mappings. Mappings which obey constraints, e.g. on length distortion, have already been noticed to play a role in our understanding of auxetic and other nonconforming micromaterials which are actually mechanisms. Mappings subject to length-related constraints are obviously relevant for the modeling of the behaviour of more general sheet-like materials. We believe that in this area, well-informed geometric modeling can significantly speed up simulations.

ACKNOWLEDGMENTS

REFERENCES

Dmitrij V. Alekseevskij, Ernest B. Vinberg, and Aleksandr S. Solodovnikov. 1993. Geometry of spaces of constant curvature. In *Geometry II*. Springer, 1–138.

Niccolo Baldassini, Nicolas Leduc, and Alexander Schiftner. 2013. Construction aware design of curved glass facades: The Eiffel Tower Pavilions. In *Glass Performance Days Finland (Conference Proceedings)*. 406–410.

Eric Baldwin. 2018. SOM Designs Kinematic Sculpture for Chicago Design Week. ArchDaily (Jan 19). https://www.archdaily.com/904506

Mirela Ben-Chen, Adrian Butscher, Justin Solomon, and Leonidas Guibas. 2010. On Discrete Killing Vector Fields and Patterns on Surfaces. *Comp. Graph. Forum* 29, 5 (2010), 1701–1711.

Alexander Bobenko and Yuri Suris. 2008. *Discrete differential geometry: Integrable Structure*. American Math. Soc.

David Bommes, Henrik Zimmer, and Leif Kobbelt. 2009. Mixed-integer Quadrangulation. *ACM Trans. Graph.* 28, 3 (2009), 77:1–10.

Mario Botsch, Stephan Steinberg, Stephan Bischoff, and Leif Kobbelt. 2002. OpenMesh: A Generic and Efficient Polygon Mesh Data Structure. Proc. OpenSG Symposium. *https://graphics.uni-bielefeld.de/publications/openmesh.pdf*.

Sofien Bouaziz, Mario Deuss, Yuliy Schwartzburg, Thibaut Weise, and Mark Pauly. 2012. Shape-Up: Shaping Discrete Geometry with Projections. *Comp. Graph. Forum* 31, 5 (2012), 1657–1667.

Alexander M. Bronstein, Michael M. Bronstein, and Ron Kimmel. 2008. *Numerical geometry of non-rigid shapes.* Springer.

Frédéric Cazals and Marc Pouget. 2003. Estimating differential quantities using polynomial fitting of osculating jets. In *Proc. Symp. Geometry Processing.* 177–178.

Albert Chern, Felix Knöppel, Ulrich Pinkall, and Peter Schröder. 2018. Shape from Metric. *ACM Trans. Graph.* 37, 4 (2018), 63:1–17.

Sebastian Claici, Mikhail Bessmeltsev, Scott Schaefer, and Justin Solomon. 2017. Isometry-Aware Preconditioning for Mesh Parameterization. *Comp. Graph. Forum* 36, 5 (2017), 37–47.

Robert Connelly. 1987. Infinitesimal Rigidity. In *Theory of rigid structures (unpublished collection).* http://pi.math.cornell.edu/~connelly/rigidity.chapter.2.pdf

Manfredo do Carmo. 1976. *Differential Geometry of Curves and Surfaces.* Prentice-Hall.

Lionel du Peloux, Olivier Baverel, Jean-François Caron, and Frédéric Tayeb. 2013. From shape to shell: a design tool to materialize freeform shapes using gridshell structures. In *Rethinking Prototyping.* Proc. Design Modelling Symposium Berlin.

Michael Eigensatz, Martin Kilian, Alexander Schiftner, Niloy Mitra, Helmut Pottmann, and Mark Pauly. 2010. Paneling Architectural Freeform Surfaces. *ACM Trans. Graph.* 29, 4 (2010), 45:1–10.

Konstantinos Gavriil, Ruslan Guseinov, Jesús Pérez, Davide Pellis, Paul Henderson, Florian Rist, Helmut Pottmann, and Bernd Bickel. 2020. Computational Design of Cold Bent Glass Façades. *ACM Trans. Graph.* 39, 6 (2020), 208:1–16.

Ruslan Guseinov, Eder Miguel, and Bernd Bickel. 2017. CurveUps: Shaping Objects from Flat Plates with Tension-Actuated Curvature. *ACM Trans. Graph.* 36, 4 (2017), 64:1–12.

David W. Henderson and Daina Taimina. 2001. Crocheting the Hyperbolic Plane. *Math. Intelligencer* 23, 2 (2001), 17–28.

Qi-Xing Huang, Bart Adams, Martin Wicke, and Leonidas Guibas. 2008. Non-Rigid Registration under Isometric Deformations. *Comp. Graph. Forum* 27, 5 (2008), 1449–1457.

Ivan Izmestiev. 2017. Classification of flexible Kokotsakis polyhedra with quadrangular base. *Int. Math. Res. Not.* 3 (2017), 715–808.

Alec Jacobson, Daniele Panozzo, et al. 2018. *libigl*: A simple C++ geometry processing library. https://libigl.github.io

Caigui Jiang, Cheng Wang, Florian Rist, Johannes Wallner, and Helmut Pottmann. 2020. Quad-mesh based isometric mappings and developable surfaces. *ACM Trans. Graph.* 39, 4 (2020), 128:1–13.

L. Klein, To. Wagner, C. Buchheim, and D. Biermann. 2014. A procedure for the evaluation and compensation of form errors by means of global isometric registration with subsequent local reoptimization. *Prod. Eng.* 8 (2014), 81–89.

Mina Konaković-Luković, Julian Panetta, Keenan Crane, and Mark Pauly. 2018. Rapid Deployment of Curved Surfaces via Programmable Auxetics. *ACM Trans. Graph.* 37, 4 (2018), 106:1–13.

Michael Lewis. 1973. Roof Cladding of the Sydney Opera House. *J. & Proc. Royal Soc. New South Wales* 106 (1973), 18–32.

Julian Lienhard, Simon Schleicher, Simon Poppinga, Tom Masselter, Markuks Milwich, Thomas Speck, and Jan Knippers. 2011. Flectofin: a hingeless flapping mechanism inspired by nature. *Bioinspir. Biomim.* 6, Article 045001 (2011).

Ligang Liu, Lei Zhang, Yin Xu, Craig Gotsman, and Steven J. Gortler. 2009. A local/global approach to mesh parametrization. *Comp. Graph. Forum* 27, 5 (2009), 1495–1504.

Yang Liu, Helmut Pottmann, Johannes Wallner, Yong-Liang Yang, and Wenping Wang. 2006. Geometric modeling with conical meshes and developable surfaces. *ACM Trans. Graph.* 25, 3 (2006), 681–689.

Stuart P Lloyd. 1982. Least Squares Quantization in PCM. *IEEE Trans. Information Th.* 28 (1982), 129–137.

Kaj Madsen, Hans Bruun Nielsen, and Ole Tingleff. 2004. *Methods for non-linear least squares problems* (2nd ed.). Technical Univ. Denmark.

Luigi Malomo, Jesús Pérez, Emmanuel Iarussi, Nico Pietroni, Eder Miguel, Paolo Cignoni, and Bernd Bickel. 2018. FlexMaps: Computational Design of Flat Flexible Shells for Shaping 3D Objects. *ACM Trans. Graph.* 37, 6 (2018), 231:1–14.

Tom Masselter, Simon Poppinga, Julian Lienhard, Simon Schleicher, and Thomas Speck. 2012. The flower of Strelitzia reginae as concept generator for the development of a technical deformation system for architectural purposes. In *Proc. 7th. Plant Biomechanics Int. Conf.* INRIA, 389–392.

Sumner B. Myers. 1936. Isometries of 2-dimensional Riemannian manifolds into themselves. *Proc. Nat. Acad. Sc. USA* 22 (1936), 297–300.

Julian Panetta, Mina Konaković-Luković, Florin Isvoranu, Etienne Bouleau, and Mark Pauly. 2019. X-Shells: A New Class of Deployable Beam Structures. *ACM Trans. Graph.* 38, 4 (2019), 83:1–15.

Chi-Han Peng, Caigui Jiang, Peter Wonka, and Helmut Pottmann. 2019. Checkerboard Patterns with Black Rectangles. *ACM Trans. Graph.* 38, 6 (2019), 171:1–13.

Nico Pietroni, Marco Tarini, and Paolo Cignoni. 2010. Almost Isometric Mesh Parameterization through Abstract Domains. *TVCG* 16, 4 (2010), 621–635.

Fig. 19. *Paneling free forms with an isometrically deforming material*. This is an actual model built with high impact polystyrene panels. For the design surface and a photo of the molds see Fig. 18.

Helmut Pottmann, Michael Eigensatz, Amir Vaxman, and Johannes Wallner. 2015. Architectural Geometry. *Computers & Graphics* 47 (2015), 145–164.

Helmut Pottmann, Qi-Xing Huang, Yong-Liang Yang, and Shi-Min Hu. 2006. Geometry and convergence analysis of algorithms for registration of 3D shapes. *Int. J. Computer Vision* 67, 3 (2006), 277–296.

Helmut Pottmann, Johannes Wallner, and Stefan Leopoldseder. 2001. Kinematical methods for the classification, reconstruction and inspection of surfaces. In *SMAI 2001: Congrès national de mathématiques appliquées et industrielles.* 51–60.

Michael Rabinovich, Tim Hoffmann, and Olga Sorkine-Hornung. 2019. Modeling Curved Folding with Freeform Deformations. *ACM Trans. Graph.* 38, 6 (2019), 170:1–12.

Idzhad Kh. Sabitov. 1992. Local Theory of Bendings of Surfaces. In *Geometry III.* Springer, 179–256.

Alexei Sacharow, Jonathan Balzer, Dirk Biermann, and Tobias Surmann. 2011. Non-rigid isometric ICP. *Computer-Aided Design* 43 (2011), 1758–1768.

Josua Sassen, Behrend Heeren, Klaus Hildebrandt, and Martin Rumpf. 2020. Geometric optimization using nonlinear rotation-invariant coordinates. *Computer Aided Geom. Des.* 77, Article 101829 (2020).

Robert Sauer. 1970. *Differenzengeometrie.* Springer.

Alexander Schiftner, Michael Eigensatz, Martin Kilian, and Gery Chinzi. 2013. Large scale double curved glass facades made feasible – The Arena Corinthians West Facade. In *Glass Performance Days Finland (Conference Proceedings).* 494 – 498.

Justin Solomon, Mirela Ben-Chen, Adrian Butscher, and Leonidas Guibas. 2011a. As-Killing-As-Possible Vector Fields for Planar Deformation. *Comp. Graph. Forum* 30, 5 (2011), 1543–1552.

Justin Solomon, Mirela Ben-Chen, Adrian Butscher, and Leonidas Guibas. 2011b. Discovery of Intrinsic Primitives on Triangle Meshes. *Comp. Graph. Forum* 30, 2 (2011), 365–374.

Olga Sorkine and Mark Alexa. 2007. As-rigid-as-possible surface modeling. In *Proc. Symposium Geometry Processing.* 109–116.

Sivan Toledo. 2003. TAUCS, A Library of Sparse Linear Solvers. Tel Aviv University. www.tau.ac.il/~stoledo/taucs

Michael Wand, Philipp Jenke, Qixing Huang, Martin Bokeloh, Leonidas Guibas, and Andreas Schilling. 2007. Reconstruction of Deforming Geometry from Time-Varying Point Clouds. In *Proc. Symp. Geometry Processing.* 49–58.

Hui Wang, Davide Pellis, Florian Rist, Helmut Pottmann, and Christian Müller. 2019. Discrete Geodesic Parallel Coordinates. *ACM Trans. Graph.* 38, 6 (2019), 173:1–13.

Walter Wunderlich. 1951. Zur Differenzengeometrie der Flächen konstanter negativer Krümmung. *Sitzungsber. Österr. Ak. Wiss. II* 160 (1951), 39–77.