

Curvature-based Offset Distance: Implementations and Applications

Wei Zhuo^a, Jarek Rossignac^a

^awzhuo3, jarek@cc.gatech.edu
College of Computing, Georgia Institute of Technology

Abstract

We address three related problems. The first problem is to change the volume of a solid by a prescribed amount, while minimizing Hausdorff error. This is important for compensating volume change due to smoothing, subdivision, or advection. The second problem is to preserve the individual areas of infinitely small chunks of a planar shape, as the shape is deformed to follow the gentle bending of a smooth spine (backbone) curve. This is important for bending or animating textured regions. The third problem is to generate consecutive offsets, where each unit element of the boundary sweeps the same region. This is important for constant material-removal rate during numerically controlled (NC) machining. For all three problems, we advocate a solution based on normal offsetting, where the offset distance is a function of local or global curvature measures. We discuss accuracy and smoothness of these solutions for models represented by triangle or quad meshes or, in 2D, by spine-aligned planar quads. We also explore the combination of local distance offsetting with a new selective smoothing process that reduces discontinuities and preserves curvature sign.

1. Introduction

In this paper, we discuss the use of normal offsetting [1] for volume or area preservation, where the offset distance is computed globally or locally from curvature measures. Specifically, we address the following three problems.

1.1. Adjust volume while minimizing Hausdorff error

We are given a base solid P with volume V_P . Typically, P is obtained by applying a small deformation to some starting solid S , which has volume V_S . The deformation may be the result of subdivision [2], smoothing [3], or advection of a fluid/swimmer interaction [4]. We want to obtain an offset solid O that is similar to P , but has volume V_S . Specifically, we define O as the shape that minimizes the Hausdorff distance, $\delta(P, O)$, between P and O , with O constrained to having volume V_S . Maintaining the volume is important in manufacturing applications where weight matters [5] and in physically based simulations where incompressibility matters [6]. The solution proposed here defines O as the constant distance offset (CDO) of P : $O = P^h$. We explain how to compute the correct distance h , both in two and three dimensions. We discuss accuracy in cases where P and O are represented by piecewise linear boundaries. In Fig. 1, we compare this solution to

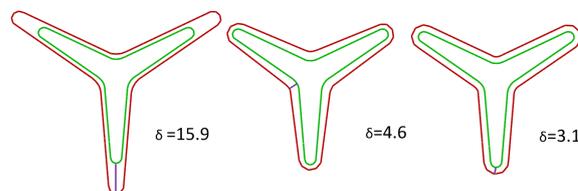


Figure 1: The original 3-branch-star base shape P (green) is shown with three offset shapes O (red) that enclose regions of the same area: global scaling (left), variable distance offsetting (center), and constant distance offsetting (right). The respective Hausdorff distances are: 15.9, 4.6, and 3.1. A line segment connecting P and O indicate where the Hausdorff distance is reached. On the right, all points are at the Hausdorff distance from the other set.

global scaling and to variable distance normal offsetting (discussed in Sec. 1.3).

1.2. Preserve local area during spine bending

We are given a portion of an image R . R roughly aligned along a smooth spine curve P . Note that P does not need to be the medial axis of R and that the width of R may vary along P . We are also given a bent version \tilde{P} of P . We assume that \tilde{P} and P have identical length and are both parameterized by arc-length. Assume that each point O of R has a unique closest projection on P . We want a locally area-preserving homeomorphism H that maps point $O = P(s) + rN(s)$ to point $\tilde{P}(s) + h\tilde{N}(s)$, where

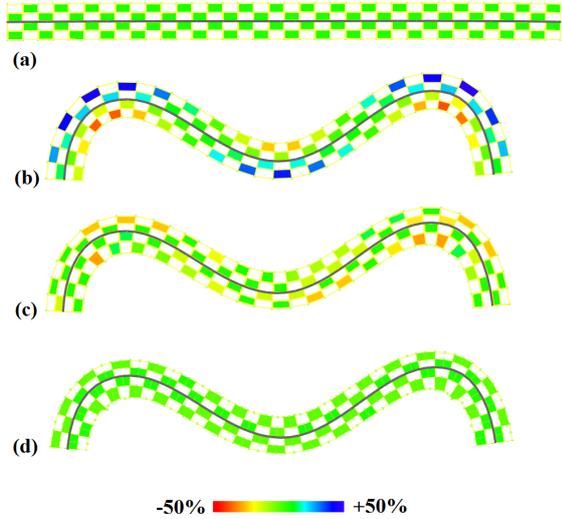


Figure 2: On the top (a), we show a texture region painted with an axis-aligned checkerboard pattern along a straight spine curve P . Below (b), we show a deformed version \bar{P} of the spine and the result of a mapping where $h = r$. The squares of the checkerboard are colored to indicate area preservation (more green), compression (more red), or dilation (more blue). Below (c), we show the proposed corrected mapping. At the bottom (d), we show the proposed corrected mapping while doubling the sampling density. Notice that this increased sampling reduces area errors significantly.

$N(s)$ is the normal to P at $P(s)$ and $\bar{N}(s)$ is the normal to \bar{P} at $\bar{P}(s)$. By locally area-preserving, we mean that any subset Q of R has same area as its image $H(Q)$.

The approach that we advocate here defines h in terms of r and the curvature $k(s)$ of P at $P(s)$ and the curvature $\bar{k}(s)$ of \bar{P} at $\bar{P}(s)$. For an exact solution h to exist, r must fall within a specific range defined by $k(s)$ and $\bar{k}(s)$. In Fig. 2, we compare this “fleshing” solution to the common skinning solution with $h = r$. We also discuss the computational and accuracy advantages of the spine-aligned grid, as shown in Fig. 2, over an axis aligned grid.

1.3. Generate contours for constant material removal

We are given the planar boundary P of a pocket to be machined, and we want to compute a series, $\{O_j\}$, of concentric variable-distance normal offset contours. For each contour, we want to adjust the offset distance locally, so that the area of a segment of the corridor between two consecutive contours is proportional to the length of that segment. More precisely, consider an animation that moves all points of O_j along their normal until they reach their offset point on O_{j+1} . For any connected subset S of O_j , let u denote its length. Our objective is to ensure that the region swept by S during this

animation has area ur , where r is a given nominal depth. This is important because NC machining is most efficient when the cutter advances at constant speed (tangentially along a contour O_i) and removes a constant amount of material per unit of time [7]. Our solution combines two steps: (1) a variable distance offset where the local offset distance h is computed from the nominal distance r and the local curvature k of O_j using a simple variation of the formulation discussed above, and (2) a selective smoothing, which reduces the sharp features introduced by step (1) and ensures that the curvature at a point does not change sign during offsetting. In Fig. 3, we compare constant distance offsetting, variable distance offsetting, and the proposed solution which combines steps (1) and (2).

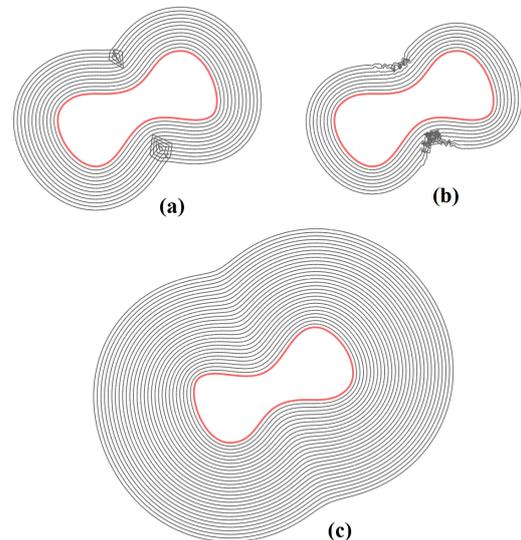


Figure 3: We show a series of contours produced by constant distance offsetting (a), curvature-based distance offsetting (b), and curvature-based distance offsetting with selective smoothing (c). The successive constant distance offsets (a) do not preserve a constant area-to-length ratio and produce self-intersections for larger offset distances. Successive curvature-based offsets (b) preserve that ratio, but exhibit an increasing amount of discontinuities where the curvature of the previous offset changes rapidly (we only render the first few contours). The proposed combination of curvature-aware offsetting and selective smoothing (c) produces concentric offset contours that are smooth and approach a constant area-to-length ratio. The selective smoothing ensures that the curvature at each point maintains its sign or becomes zero. Hence, the process converges towards a convex shape, as can be extrapolated from the drawing.

1.4. Summary of contributions

The solutions to all three problems are based on a curvature-based distance correction, which maps a

nominal distance r to a distance h . In two dimensions, assuming that k is the curvature, h is a specific root of

$$\frac{1}{2}kh^2 + h - r = 0 \quad (1)$$

In three dimensions, assuming that g is the Gaussian and m the mean curvature, h is a specific root of

$$\frac{1}{3}gh^3 + mh^2 + h - r = 0 \quad (2)$$

The derivation of these equations and their prior use for constant area or volume offsetting is discussed in the next section. Our contributions comprise the following:

1) To solve the first problem of constant distance offsetting for a desired volume change, we generalize the Steiner formula [8] for the volume change under constant distance offsetting to non-convex solids as well as to higher genus solids, and we describe an efficient implementation. We also analyze the error sensitivity of our formula, study the impact of sampling density on its accuracy, and report the results on benchmark curves and surfaces.

2) To solve the second problem of local area preservation during skeletal bending, we have adapted the formulation (Equ. 1) originally developed by Chirikjian [9] for locally area-preserving bending. Chirikjian discusses divergence-free deformation for continuous models. We explore its use for deforming discrete, texture-mapped quads to follow the bending of a polygonal spine. Specifically, we propose the use of a spine-aligned grid, and argue its advantages over axis-aligned grids.

3) To address the third problem of constant material removal modeling, we build upon the solution proposed by Moon [7], but show that it produces sharp discontinuities of the offset curve near concave features. We propose a novel selective smoothing technique which eliminates these sharp features while preserving the curvature sign between the original points and their offsets.

2. Prior Art

In this section, we discuss relevant prior work in constant distance offsetting, variable distance offsetting, volume correction, and skeleton-driven shape deformations.

2.1. Constant distance offsetting

The constant-distance offset (CDO) S^r of a solid S by distance r [10], also called dilation, is formulated as

the Minkowski sum [11] of S with a ball of radius r centered at the origin. It may also be expressed as the union of all balls of radius r with center in S . S^r contains all points at distance r or less from S . Steiner [8] has derived formulae for the area change and volume change under constant distance offsetting for the special cases of convex sets of genus zero. Here in Sec. 4 we prove its generalization to non-convex solids and to higher genus solids.

CDO operations are important in planning and simulating NC-machining processes [12], where they are used to generate constant thickness layers of material to be removed by successive machining passes, and for creating fillets and blends [13] by offsetting the solid and then its complement or vice versa. In 2D, CDO preserves the domain of shapes bounded by piecewise-circular curves [14]. In 3D, we obtain our approximation by offsetting each vertex by a constant distance along an estimated vertex normal. Numerical and topological accuracy issues of CDOs of solids bounded by triangle meshes and polyhedral surfaces have been investigated in various applications [12] [15].

2.2. Variable distance offsetting

Variable-distance offsetting (VDO) is specified by assigning a distance $h(s)$ to each point $P(s)$ of the base shape P (curve in 2D or surface in 3D). Three different interpretations of this specification have been compared in [16]. The radial offset is the union of balls $(P(s), h(s))$. The ball offset [17] is the union of balls of diameter $h(s)$ that are tangent to P at $P(s)$. Finally, the normal offset [1] is the union of all line segments of length $h(s)$ that are normal to P at $P(s)$. In all three cases, under sufficient assumptions on the smoothness and curvature of P , there is a bijective mapping between P and a portion of the boundary of the offset shape, which may be formulated as an envelope of a set of line segments or balls. (Note that each formulation imposes a different set of constraints on the relation between the offset distance function and the curvature of P [1].) The shape and curvature of these envelopes may be computed efficiently [16]. Here, we restrict our attention to the normal offset, hoping that the other two interpretations will be investigated later. One issue addressed in this paper is the computation of the offset distance field $h(s)$ that distributes the “invaded” space uniformly. Let P be a surface in 3D. Let, Q be a subset of P , and R be the region swept by Q during the offset. We want to compute a variable offset distance function $h(s)$ such that the ratio r of R 's volume over the area of Q is a constant. If P is a curve in 2D, r is the ratio of the area of R over the arc-length of Q . This equi-volumetric offsetting

has been investigated by Moon [7] [18] for NC machining, so as to ensure a constant material-removal rate, rather than constant depth of removal. Moon has shown that, in valid situations where the curvature is smaller than some limit defined in terms of r , $h(s)$ may be formulated as the root of a quadratic equation, for the 2D case, and of a cubic equation, for the 3D case. Specifically, in 2D, h is the root of $\frac{1}{2}k(s)h(s)^2 + h(s) - r = 0$, where $k(s)$ is the curvature of P at $P(s)$. In 3D, $h(s)$ is the root of $\frac{1}{3}g(s)h(s)^3 + m(s)h(s)^2 + h(s) - r = 0$ where $g(s)$ is the local Gaussian curvature and $m(s)$ is the local mean curvature of P at $P(s)$. These curvature based distance functions have been studied by Hagen and Hahmann as generalized focal surfaces [19] as a tool for surface interrogation. We build our local offsetting solutions to the volume compensation and to the area-preserving bending on these equations.

2.3. Skeleton-driven deformations

Consider the planar shape S to be the union of an infinite set of disjoint line segments intersected at their midpoints by a continuous spine P . Let $2h(s)$ and $a(s)$ define the length of the line segment and its angle to the tangent to P at $P(s)$. Cavalieri's principle [20] implies that, when bending P , the area of the convex hull of two infinitely close line segments remains constant regardless of the shape of S , as long as we preserve $h(s)$ and $a(s)$ and do not bend $P(s)$ excessively (ensuring that the radius of curvature at $P(s)$ does not exceed $h(s)$). Although this solution preserves the area of each convex hull of consecutive two line segments, it does not preserve the local area on each side of the spine, as discussed in the introduction. Several approaches have been proposed to maintain a constant local area of a region as its spine is bent. Chirikjian [9] has derived the quadratic equation mentioned above by constraining the Jacobian of the deformation to be 1, so as to make it locally area preserving. When the spine bend exceeds the local limit, the normal offsetting is no longer appropriate. More general techniques for skinning and fleshing with locally-preserving bending have been proposed by Rohmer and colleagues [21]. They adjust both the direction and distance of the offsetting and solve for an optimal solution that favors locality.

3. Curvature-based Offset Distance Computation

In this section, we discuss implementation and accuracy issues of computing the curvature-based offset distance. For implementation simplicity, we define a function f in 2D and in 3D, which returns the proper offset

distance, when it exists within the allowable range, or the appropriate range bound otherwise. We use a subscript (f_{2D} and f_{3D}) to distinguish the 2D and 3D versions of f . We also discuss how to select the proper root in each case.

3.1. Function interface and capping

f_{2D} takes as input the signed curvature k and the reference distance r respectively. The output $h = f_{2D}(k, r)$ is the quadratic root $\frac{-1 + \sqrt{1+2kr}}{k}$ of Equ. 1 when $1+2kr > 0$. Otherwise, f caps the value of h and returns the limit $-1/k$ so as to prevent a local self-intersection.

f_{3D} takes as input the signed Gaussian curvature g , the mean curvature m and the reference distance r . The output $h = f_{3D}(g, m, r)$ is the valid cubic root of Equ. 2. Notice that if $g = 0$, then h is computed via the 2D solution discussed above, as $f_{2D}(2m, r)$. Otherwise, we need to select the proper real root and to ensure that the solution is capped to an allowable bound. Moon [18] has derived the existence condition and the monotonic region where the valid root exists. In our implementation, we use a change of variables: $h^* = \frac{h}{r}$, $g^* = gr^2$ and $m^* = mr$. Then if $2\sqrt{m^{*2} - g^*} - m^* > 3(m^* - \sqrt{m^{*2} - g^*})$, there is a unique positive real root in $[0, \frac{1}{\sqrt{m^{*2} - g^*} - m^*}]$. Otherwise, no valid real root exists and we output the maximum offset distance that is free from a local self intersection.

3.2. Error Sensitivities

Estimating curvature from a sampling of a smooth curve will produce an incorrect offset distance. Below we show that the error in h is a linear function of the errors in the curvature estimation, both in 2D and in 3D.

Let ϵ_x represent a small variation in the variable x . Assume that r is a constant. For 2D, we take the derivative of Equ. 1 and arrive at

$$\frac{h^2}{2}\epsilon_k + kh\epsilon_h + \epsilon_h = 0$$

From this, we conclude that ϵ_h is proportional (\propto) to ϵ_k :

$$\epsilon_h \propto \frac{h^2}{1 + kh}\epsilon_k$$

Similar for 3D, we take the derivative of Equ. 2 and obtain

$$\epsilon_h \propto \frac{\epsilon_g h^3 + \epsilon_m h^2}{1 + 2mh + gh^2}$$

Therefore, the numerical error in the output of f is linear in the errors of its inputs when $kh > 0$ in 2D, or $2mh + gh^2 > 0$ in 3D.

3.3. Curvature approximation

Densely sampled polylines and polygonal meshes are often used in modeling solids with smooth boundaries whose parametric expression may not be conveniently available. Hence, we adopt discrete formulas to evaluate the curvatures.

3.3.1. Local curvatures

Let P denote a watertight quad or triangle mesh and P_i a vertex of P . The local curvature at P_i can be evaluated from its one-ring neighbors $\{Q_j\}$. In 2D, the discrete curvature k_i may be conveniently calculated by fitting a parabola to P_i and its neighbors. In 3D, we use the discrete formulas proposed by Meyer, et. al. [22]. Specifically, the local area A_i associated with P_i is approximated by the area sum of incident Voronoi cells. The gradient of A_i with respect to P_i , also known as the discrete Laplace Beltrami operator, has the following closed form [23]:

$$\nabla A_i = \frac{1}{2} \sum_j \left(\frac{P_i Q_{j-1} \cdot Q_{j-1} Q_j}{|P_i Q_{j-1} \times Q_{j-1} Q_j|} + \frac{P_i Q_{j+1} \cdot Q_{j+1} Q_j}{|P_i Q_{j+1} \times Q_{j+1} Q_j|} \right) P Q_j \quad (3)$$

Then, the local mean curvature is approximated by a scaled dot product of ΔA_i with the unit normal at P_i . The local Gaussian curvature is approximated by the angle deficit at P_i [22].

3.3.2. Global curvatures

Let A_P denote the total surface area of P . We refer to the surface integral of Gaussian curvature divided by A_P as the global Gaussian curvature (g_P) and the surface integral of mean curvature divided by A_P as the global mean curvature (m_P). The integrated Gaussian is intrinsic to P and equals $2\pi\chi_P$, where χ_P is the Euler characteristic of P . ($\chi_P = V - E + F$ where V, E, F are numbers of vertices, edges and faces.) Therefore,

$$g_P = \frac{2\pi\chi_P}{A_P} \quad (4)$$

The surface integral of mean curvature is related to the bending energy [24], which we denote as E_P . Note that E_P can be approximated by the scaled sum of $|\nabla A_i|$ at each vertex. Therefore,

$$m_P = \frac{E_P}{A_P} \quad (5)$$

In 2D when P denotes a Jordan curve, its integrated curvature is intrinsic and equals 2π [25]. Let L_P denote the length of P . The global curvature of P , k_P , is defined as

$$k_P = \frac{2\pi}{L_P} \quad (6)$$

Note that a global curvature has the same unit as its local counterpart.

4. Dilation with Prescribed Volume Change

Consider a 3D shape P with volume V_P . We want to compute O from P by a single step of dilation, so that the enclosed volume is increased by a prescribed amount ΔV . We first discuss methods that are not based on curvature measures. Then we present our solution.

4.1. Uniform scaling

The work of Desbrun et. al. [23] introduces a simple approach of rescaling P around its barycenter C by a uniform amount s :

$$O = C + s(P - C) \quad (7)$$

where $s = \sqrt[3]{\frac{V_P + \Delta V}{V_P}}$. Uniform scaling guarantees that the enclosed volume is increased exactly by ΔV . However, this approach generates unbounded Hausdorff error between O and P (Fig. 1).

4.2. Linearized solution

In contrast, when a constant distance normal offset by a distance h is used, the Hausdorff error is exactly h (assuming that h is smaller than the least feature size of the shape). When using a constant distance offset (CDO), to increase the volume of a solid by ΔV , one must compute the proper offset distance h . One approach [21] is to use $h = \frac{\Delta V}{A_P}$. We compare below this approximate solution to the one proposed here.

4.3. Normal offset based on the global curvature

The correct solution defines h as the appropriate root computed by f_{2D} or f_{3D} as explained earlier in Sec. 3.1. We include below the derivation of this result.

4.3.1. 2D

Let P denote a Jordan curve of length L_P . Let, $k(s)$ and $N(s)$ be the signed curvature and the unit normal of P at $P(s)$. The curvature $k(s)$ is the derivative of the unit normal. Hence, we have the following expression of the area increase ΔA associated with offsetting P by a constant distance h :

$$\begin{aligned} \Delta A &= \iint_{\gamma \in [0, h]} \left| \frac{\partial(P(s) + \gamma N(s))}{\partial s} \right| d\gamma ds \\ &= hL_P + \frac{h^2}{2} \int k(s) ds \end{aligned}$$

By the Total Curvature Theorem [25], we have

$$\int k(s)ds = 2\pi$$

Therefore we arrive at,

$$\frac{\pi}{L_P}h^2 + h - \frac{\Delta A}{L} = 0 \quad (8)$$

Hence to compensate for the area change ΔA , we need to offset the curve P by a constant distance h computed by $h = f_{2D}(\frac{2\pi}{L_P}, \frac{\Delta A}{L_P})$. Or equivalently, $h = f_{2D}(k_P, \frac{\Delta A}{L_P})$ using the global curvature defined in Equ. 6.

4.3.2. 3D

Let $P(u, v)$ denote a point on a surface P parameterized by u and v . We derive the exact expression of the volume increase when offsetting $P(u, v)$ by a constant distance h . Let $m(u, v)$ and $g(u, v)$ represent the local mean and Gaussian curvature of P at (u, v) . Since the mean curvature is the divergence of the unit normal and the Gaussian curvature is the determinant of its Hessian, the volume increase ΔV can be expressed as follows:

$$\begin{aligned} \Delta V &= \iiint_{\gamma \in [0, h]} |\nabla(P(u, v) + \gamma N(u, v))| dydudv \\ &= h \iint |\nabla P| dvdu + \frac{1}{2}h^2 \iint \nabla \cdot N dvdu \\ &\quad + \frac{1}{3}h^3 \iint |\nabla N| dudv \\ &= hA_P + h^2 \iint m(u, v) dudv \\ &\quad + \frac{1}{3}h^3 \iint g(u, v) dudv \end{aligned}$$

By the Gauss-Bonnet Theorem [25], we have

$$\iint g(u, v) dudv = 2\pi\chi_P$$

where χ_P is the Euler characteristic of P which is $2 - g$ for a genus- g surface. The other integral term is the total integral of the mean curvature: $E_P = \iint m(u, v) dudv$. Therefore, we arrive at:

$$\frac{2\pi\chi_P}{3A_P}h^3 + \frac{E_P}{A_P}h^2 + h - \frac{\Delta V}{A_P} = 0 \quad (9)$$

Hence to increase the current volume by ΔV , we offset P by $h = f_{3D}(\frac{2\pi\chi_P}{A_P}, \frac{E_P}{A_P}, \frac{\Delta V}{A_P})$. Notice that the definition of global curvatures in Equ.4 and Equ. 5, the solution can also be written as $h = f_{3D}(g_P, m_P, r)$.

4.4. Proof of minimizing Hausdorff error

Let P, O and Q either be regularized planar regions or solids. Assume that $O = P^d$ for some positive distance d . (If instead we want a negative d , the argument below will hold for the complements of P, O and Q and still support our conclusion.) We will prove that $\forall Q \neq O, V_Q = V_O \Rightarrow H(Q, P) > H(O, P)$, where H defines Hausdorff distance and V_X denotes the area or volume of X .

Assume that $V_Q = V_O$. First, we note that Q cannot be a proper subset of O , otherwise we would have $V_Q < V_O$. Second, we note that Q cannot contain any point q outside of O , otherwise we would have the distance from q to P , $d(q, P) > d$ (Since O includes all points at distances less or equal to d from P) and hence $H(Q, P) > d$. From these two observations (Q is not a proper subset of O and Q is a subset of O), we conclude that if $Q \neq O$ then $H(Q, P) > H(O, P)$. Hence, O is Hausdorff distance minimized. \square

4.5. Implementation

We have implemented the three volume correction schemes (Uniform scaling, Linearized, and Curvature-based solutions) on quad as well as triangle meshes. Our implementation uses a Corner Table [26] representation and the associated corner operators. The whole process is only a few lines of code. First, to compute the global mean curvature m_P we sum the area gradient at each vertex and divide it by 3 for triangle meshes or 2 for quad meshes. Then, the normal at each vertex is the weighted sum of the normals of the incident triangles scaled by their areas. Then, we compute the surface area A_P of P (as the sum of triangle areas), the volume V_P (as a sum of signed volumes of the tetrahedron formed by each triangle with the origin). For a quad mesh, we treat each quad face as a bi-linear patch interpolating the four face vertices. We compute the total volume and the total surface area as the sums of the sub-volume and the sub-area associated with each bi-linear patch, using formulae presented in [21]. The extraction of the proper root of the cubic polynomial was discussed in Sec. 3.1. Although we have not optimized the code, the whole process of computing the corrected offset distance and of performing the offsetting is instantaneous (it takes a very small fraction of a second for all models tested).

We evaluate P 's barycenter C as the area-weighted sum of geometric centers of all faces of P divided by A_P . The Hausdorff distance between P and O is approximated by

$$\max\{\max\{d(p, O), p \in P\}, \max\{d(o, P), o \in O\}\}$$

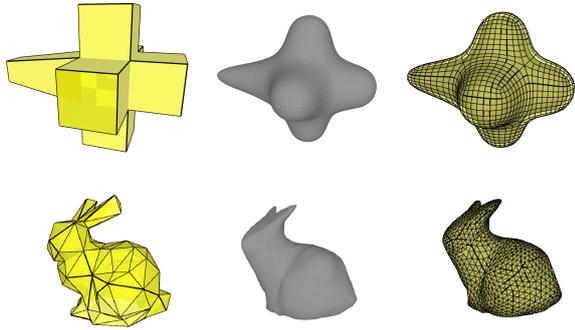


Figure 4: Steps of volume compensation through dilation. Left: original control meshes of volume V_S ; Center: fair and subdivided meshes with volume V_P ; Right: meshes after dilation with volume V_O

where $d(x, Y)$ calculates the distance from a vertex x to mesh Y .

4.6. Results

We present our experiment results on 11 meshes shown in Fig. 5. In these examples, 9 (Cross, Holes, Bunny, Horse, Donut, Spikes, Sphere-(coarse, fine)) are obtained from coarse solids by Catmull [2] or Butterfly [27] subdivision and smoothing [3] steps shown in Fig. 4. Mesh “Horse-noise” is obtained by adding random noises to the subdivided horse model. We prescribe the desired volume change ΔV , and want to offset P to produce a solid $O = P^h$ with volume $V_P + \Delta V$. We report in Tab. 1 the number of vertices n_V , volume V_P , area A_P for each mesh P . The volume after correction is denoted as V_O . It is not exactly $V_P + \Delta V$ due to numerical errors. We measure the discrepancies between $V_P + \Delta V$ and V_O in terms of ϵ defined as follows:

$$\epsilon = \frac{|V_P + \Delta V - V_O|}{V_P} \quad (10)$$

Tab. 1 shows the errors of the linearized solution (ϵ_{linear}) where $h = \frac{\Delta V}{A}$ and the errors of our solution based on the global curvatures ($\epsilon_{curv.}$). The results show that in general the curvature-based solution is about 3 times more accurate than the linearized solution. We also report the Hausdorff error between P and O . For meshes that contain parts that are long and thin, the Hausdorff error ($\delta_{scaling}$) produced by uniform scaling is much larger than the Hausdorff error ($\delta_{curv.}$) produced by our solution based on global curvatures. For spheres, $\delta_{scaling}$ and $\delta_{curv.}$ are roughly the same. We also observe that for all models tested, repeating the offsetting with the correct solution (Equ. 9) for h (each time using the remaining volume error as inputs) three or four times reduces the relative error to 0.00003% or less.

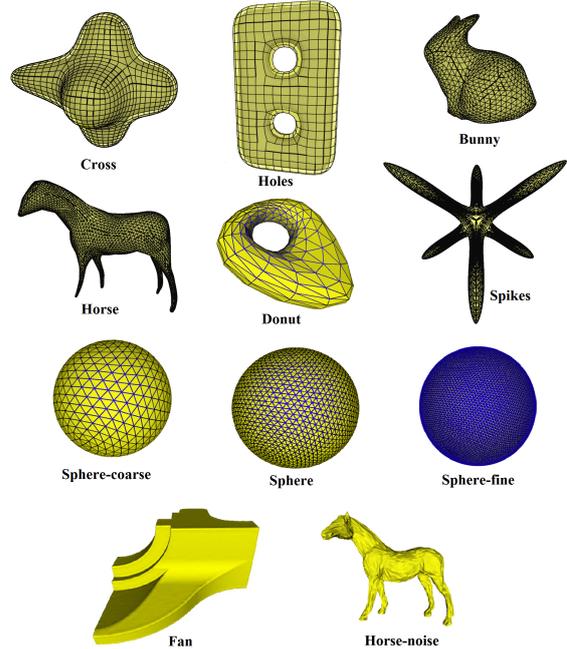


Figure 5: Mesh models used in our experiments: Cross, Holes, Bunny, Horse, Donut, Spikes, Sphere-noise, Sphere, Sphere-fine, Fan, Horse-noise

5. Spine Bending with Local Area Preservation

Volume and area preserving deformation are often keys to simulations with physical realism. The fundamental idea for locally volume/area-preservation is to make the deformation field divergence-free, which implies that the Jacobian determinant is 1.

Here we consider the problem in 2D. The spine is represented by a polygonal curve produced by subdivision or by a dense sampling of a smooth curve. Applications of bending curves range from rendering brush strokes with variable thickness and textures [28] to image and shape manipulation [29]. We notice that a ribbon-style framework suitable for bending an open continuous curve was first proposed by Alan Barr [30]. The framework provides an efficient method for a planar deformation controlled by a skeletal curve. We present below a locally area-preserving shape manipulation application based on this framework.

5.1. Continuous model

We include here a derivation of Equ. 1 for bending with a continuous curve. Given a skeletal curve which we denote as $P(s)$, a nearby point O is expressed as:

$$O(s, r) = P(s) + rN(s)$$

Model	n_V	V_P	A_P	ϵ_{linear}	$\epsilon_{curv.}$	$\delta_{scaling}$	$\delta_{curv.}$
Cross	3198	2.15e7	7.05e5	2.5%	0.12%	12.5	2.8
Holes	1922	4.26e7	7.26e5	1.9%	0.11%	14.8	5.7
Bunny	1522	5.19e6	1.73e5	2.4%	0.065%	6.3	2.9
Horse	4002	8.29e6	3.12e5	2.95%	0.21%	12.3	2.5
Donut	256	1.43e7	4.03e5	2.2%	0.18%	7.4	4.8
Spikes	3842	7.69e6	5.165e5	3.20%	0.65%	17.6	1.4
Sphere-coarse	194	3.01e7	4.73e5	3.1%	0.62%	5.8	5.7
Sphere	770	3.08e7	4.78e5	3.8%	0.83%	6.4	6.4
Sphere-fine	3074	3.08e7	4.78e5	3.9%	1.0%	6.4	6.4
Fan	25895	5.022e7	1.07e6	2.2%	1.3%	8.7	4.5
Horse-noise	4002	1.21e7	4.69e6	2.7%	0.96%	11.5	2.3

Table 1: Mesh statistics and results of different volume-correction schemes corresponding to the models in Fig. 5

here r is the distance from O to its orthogonal projection on P . We denote the skeletal curve after length-preserving bending as \bar{P} with its unit normal and curvature denoted as \bar{N} and \bar{k} . The deformed position \bar{O} is then:

$$\bar{O}(s, r) = \bar{P}(s) + h\bar{N}(s)$$

Setting $h = r$ produces an approximate solution as previously discussed in Sec. 1.2. However, the deformation is not locally area-preserving as the local rate of expansion varies depending on the curvatures at $P(s)$ and $\bar{P}(s)$. Hence, $h \neq r$. By the chain rule, we have:

$$\frac{\partial \bar{O}}{\partial O} = \frac{\partial \bar{O}}{\partial(s, h)} \frac{\partial(s, h)}{\partial(s, r)} \frac{\partial(s, r)}{\partial O}$$

By setting the determinant of the above transformation to 1, we have:

$$\frac{dh}{dr} (1 + h\bar{k}(s))(1 + rk(s))^{-1} = 1$$

Therefore,

$$\frac{\bar{k}(s)}{2}h^2 + h - (r + \frac{r^2}{2}k(s)) = 0$$

The solution for h is a curvature-based distance which can be computed by $h = f_{2D}(\bar{k}(s), r + \frac{k(s)}{2}r^2)$.

5.2. Discretization

To bend an image, the designer specifies the initial and final spine curves. We use a grid of quads and paint the bent image as a texture onto the deformed quads. One could do this using an axis-aligned grid, but such an approach has two drawbacks: (1) there is an expense of computing the closest projection of each grid point onto the initial spine curve, and (2) aliasing artifacts occur

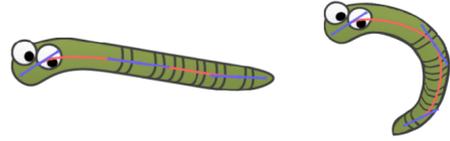


Figure 6: The user draw a initial curve (left) over an image and a deformed curve (right). The deform image is rendered as a texture mapping over the spine-aligned grid. We preserve the length of the spine by keeping the number of samples and the distance between consecutive samples as constants, when sampling from a curve manipulated by the user.

when the spine curve is not sufficiently sampled, as several grid points that would project on different points of a continuous spine may have, as closest projection, the same vertex of a polygonal approximation. To alleviate these drawbacks, we advocate using a spine-aligned grid, as shown in Fig. 2. For simplicity, we sample the smooth spine curve so that all edges of its polygonal approximation have the same length. We generate the initial grid by estimating the normal at each vertex P_i of the initial spine (as being orthogonal to the line passing by its neighbors) and by generating offset points in both directions by jr , with j being an integer in some desired range. At each such grid-point, we record its coordinates in the image as texture coordinates. To display the deformed image, we use the same process to establish the normal at each vertex of the bent spine, and generate the corresponding grid points, but instead of offsetting them by jr , we offset them by $f_{2D}(\bar{k}, jr + \frac{k}{2}(jr)^2)$, where k and \bar{k} are the local curvatures before and after bending. Then we render the grid quads with texture mapping. An example of this bending process is shown in Fig. 6.

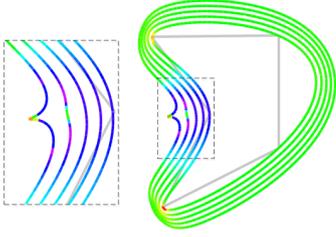


Figure 7: A family of curvature-based distance offsets. Notice that the offset curve may contain sharp pointy protrusion at concave side of the spine curve when r approaches the limit $-\frac{1}{2k(s)}$.

5.3. Limitations

As the half-width of the grid approaches the validity limit discussed above, the corrected offset distance increases rapidly, creating a spike, as shown in Fig. 7. Hence, in practice, we must limit the width of the area of the picture upon which we operate or the amount of curvature change at every point between the initial and final spines. Specifically, we limit $|k|$ to $[0, \frac{1}{2r}]$ where k is the local curvature and r is the half-width of the grid. In practice, to avoid spikes, we limit $|k|$ to $[0, \frac{1}{2.5r}]$.

6. Constant Material Removal Rate

6.1. Machining

We recall the quadratic formula proposed by Hwan Pyo Moon [7] in the context of machining:

$$\frac{1}{2}k(s)h(s)^2 + h(s) - r = 0$$

where $k(s)$ is the local curvature of the progenitor curve P , $h(s)$ is the depth of cut, and r is the material removal rate to feedrate ratio. Moon argues its importance in NC milling with constant power consumption. General milling tools have sufficient degrees of freedom which allow them to follow arbitrary planar paths. One of the challenges is to define a tool path that lead to constant material removal rate in milling for a target shape modeled by P . Since we want to keep the translational speed of the milling tool as constant as possible, the removed area per unit length should also be constant in order to achieve stable power consumption. Let this constant be r , solving the above equation gives the offset distance that defines the tool path with removed area per unit length equal to r .

6.2. Successive offsets

In practice, the tool path could consist of a set of concentric offsets from P . They form a set of *successive*

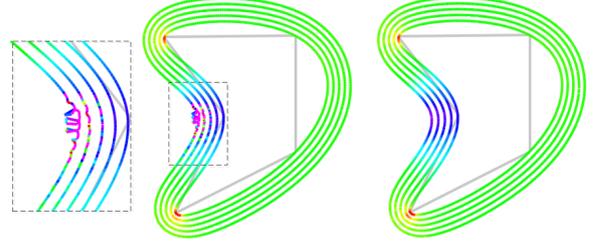


Figure 8: A set of successive curvature-based distance offsets. Left: direct offset curves without fairing; Right: the same set of offsets with selective smoothing

offsets $\{O^j\}$, $j = 1, 2, \dots$ from P :

$$\begin{aligned} O^1(s) &= P(s) + f(k_{P(s)}, r)N_{P(s)} \\ O^{j+1}(s) &= O^j(s) + f(k_{O^j(s)}, r)N_{O^j(s)} \end{aligned}$$

6.3. Loss of smoothness

It is known in differential geometry that the curvature transformation $k_{P(s)}$ is a second-order operator on the parametric curve $P(s)$. Naturally, the curvature-based distance function $f(k_{P(s)}, r)$ is second order as well. Hence only C^{d-2} continuity is observed in the offset when $P(s)$ is C^d continuous. To verify this loss of smoothness when P is approximated by dense polyloop, we show a set of successive offsets on a dense polyloop P produced by the $J_{1.5}$ subdivision scheme [31] whose limiting curve is of C^4 continuity.

Fig. 8 (Left) shows the result of directly applying f_{2D} to discrete curvatures evaluated at points of P and $\{O^j\}$. The first two offset curves appear smooth. However, the third appears jaggy and the fourth contains self-intersections. These discontinuities result from large differences of curvature estimates between neighboring vertices. Variances in evaluating the discrete curvatures could cause the offset to contain unwanted local convexities and concavities, and further increase the curvature variances in the offset curve. Therefore, we propose below an iterative algorithm called *selective smoothing*, for successively generating visually smooth offset curves.

6.4. Selective Smoothing

We observe that changes in the sign of the curvature are undesirable in generating a smooth offset curve. Hence, our smoothing strategy focuses on producing a curvature-compatible offset curve, where a point with non-negative curvature is mapped to a offset point with non-negative curvature, and the same for non-positive curvature.



Figure 9: The 1st, 3rd, 6th, 12th, 14th iteration of selective smoothing. Points with incompatible curvatures are shown in red.

Selective Smoothing is similar to the Laplacian smoothing except that only points with non-compatible curvatures are subject to the operation. It consists of two steps in each iteration (Fig. 9): *Select* and *Smoothen*. Let k_i^o denote the discrete curvature at the i -th vertex on the offset curve O ; k_i and N_i denote the signed curvature and the unit normal at P .

- **Select:** Check each vertex O_i in O and put i into a smoothing list L if k_i and k_i^o are of different signs.
- **Smoothen:** Compute a list of Laplacian vectors $\{V_i\}$ at vertices of L ; Move each vertex of L along the unit normal N_i by the dot product of V_i and N_i , and then empty L .

Typically, as shown in Fig. 9, there are only a few incompatible points along the initial offset curve. As these are made compatible by a step of the selective smoothing, some of their immediate neighbors may become incompatible. However, the process converges rapidly. Fig. 8 (Right) and 3 (Right) show results of applying selective smoothing: in Fig. 8, unwanted noise is smoothed out while the rest of a curve is not modified; in Fig 3, we are able to generate a large series of consecutive offsets using this combination of curvature-based distance and selective smoothing.

6.5. Discussion and limitations

Consider now selective smoothing as a separate process. It could be used to smoothen a polygonal curve so that each vertex is either flat (has zero curvature) or has a curvature with a prescribed sign. Selective smoothing identifies incompatible vertices—those where the curve makes the wrong turn—and moves them to the average of their immediate neighbors. When a chain of incompatible vertices has the same prescribed curvature sign, repeating the process is essentially equivalent to Laplacian smoothing and converges to a straight line. However, selective smoothing can fail if the curve becomes self-crossing. When used as a smoothing to curvature-based normal offsetting, we restrict the motion of each vertex to be along the normal to the original curve. Furthermore, the extent of that motion is constrained by the cap on the corrected offset value ($|h|$ is

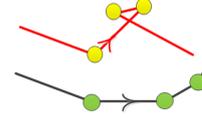


Figure 10: The yellow vertices are having compatible curvature signs with the green vertices on the black curve.

confined to $[0, -1/k]$ if $k < 0$). Hence, allowable vertex motions cannot create local loops. Therefore, we conjecture that our Selective Smoothing process will converge to a compatible curve. Of course, the offset curve may exhibit global self-intersections, which can be detected and should be prevented or resolved by trimming, if topological changes are desired. But such a global post-processing is necessary regardless of the smoothing step.

Finally, due to the discretization and numerical errors when evaluating k , an offset contour may still contain a local self-intersecting loop (Fig. 10). To detect these situations, we detect self-crossing along the offset curve and flag, as incompatible, all vertices between two consecutive self-crossing points. This heuristic works correctly only when the loops are isolated.

7. Discussion

This section discusses the impact of sampling density on the accuracy of locally area/volume distribution computed by the curvature-based normal offset. We compute variable distance normal offset from prototypical curve and surface patch (denoted as P). In order to show the error on both local and global scales, we divide P into a constant number of portions and define the following measures:

In 2D, we compute the sub-area a_k swept by offsetting the k th portion of P with length l_k . The local relative error for each portion is defined as $\delta_k = \frac{a_k}{l_k r} - 1$. We report the maximum absolute value, δ_{max} , and the mean absolute value, δ_{mean} , of the local relative errors for all portions of P . We also report the global relative error as $\delta_{global} = \frac{\sum_k a_k}{r \sum_k l_k} - 1$. δ_{global} measures the relative difference from the total-increased-area to perimeter ratio from the user-input reference distance r . In 3D, we

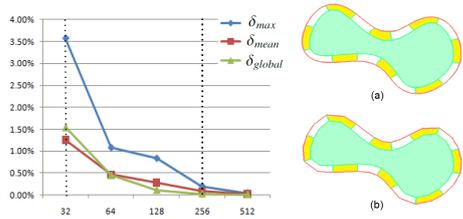


Figure 11: Dependence of the local and the global error on sampling density: (a) finely sampled curve that consists of 256 points. (b) coarsely sampled curve that consists of 32 points.

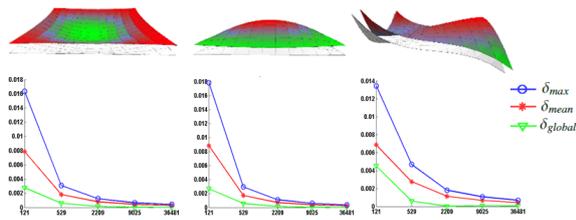


Figure 12: Dependence of the local and the global error on mesh resolutions

define similar measures which we use to analyze the errors associated with different types of surface patches. Fig. 11 shows values of δ_{max} , δ_{mean} and δ_{global} at 5 different sampling densities of a polygonal curve. Both the local and the global relative errors converge to zero as the subdivision depth increases. For example, the relative errors are less than 0.5% when there are 256 sample points on P . Fig. 12 shows values of δ_{max} , δ_{mean} and δ_{global} at different subdivision levels of bi-cubic surface patches. We collect statistics from three types of surface patches to avoid biases. Again, both the global and the local relative errors fall quickly as the sampling density increases. The relative errors are less than 0.5% when there are 529 sample points on each surface patch.

These results show that in general, the accuracy of even-area/volume distribution can be improved by increasing the sampling density.

8. Conclusion

In this paper, we have presented our study and implementation on the curvature-based offset distance for several applications. Specifically, we present a simple formulation of the offset distance and discuss its accuracy and smoothness, when computed on discrete models. We provide an exact formulation of the offset distance for adjusting the offset of 3D shapes by a constant distance offset. Our solution generalizes prior art which was limited to convex, zero-genus shapes. For bend-

ing images, we propose the use of an axis-aligned grid and the formulation of the offset mapping between two curved spines. Finally, for machining, we propose combining curvature-based local offsetting with an iterative selective smoothing process.

Acknowledgment

This project was funded by NSF grant #0811485, Collaborative Rsch: CPA-G&V-T: Aquatic Propulsion Lab. The authors would like to thank PhD students Mark Luffel, Jason Williams and Topraj Gurung for their help, Professors David Rosen and Yingjie Liu for their guidance and reviewers for their suggestions.

References

- [1] F. Chazal, A. Lieutier, J. Rossignac, Normal-map between normal-compatible manifolds, *Int. J. Comput. Geometry Appl.* 17 (5) (2007) 403–421.
- [2] E. Catmull, J. Clark, Recursively generated b-spline surfaces on arbitrary topological meshes, *Computer-Aided Design* 10 (6) (1978) 350–355.
- [3] G. Taubin, A signal processing approach to fair surface design, in: *SIGGRAPH*, 1995, pp. 351–358.
- [4] J. Tan, Y. Gu, G. Turk, C. K. Liu, Articulated swimming creatures, in: *ACM SIGGRAPH 2011 papers*, *SIGGRAPH '11*, 2011, pp. 58:1–58:12.
- [5] W. Nadir, I. Y. Kim, Structural shape optimization considering both performance and manufacturing cost, in: *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, AIAA '04, 2004, p. 4593.
- [6] B. Kim, Y. Liu, I. Llamas, X. Jiao, J. Rossignac, Simulation of bubbles in foam with the volume control method, *ACM Trans. Graph.* 26 (3) (2007) 1919–1927.
- [7] H. P. Moon, Equivolumetric offsets for 2d machining with constant material removal rate, *Computer Aided Geometric Design* 25 (6) (2008) 397 – 410.
- [8] J. Steiner, über parallele flächen, *Monatsberichte der Akademie der Wissenschaft zu Berlin* (Monthly Report of the Academy of Sciences, Berlin) (1840) 114–118.
- [9] G. S. Chirikjian, Closed-form primitives for generating locally volume preserving deformations, *Journal of Mechanical Design* 117 (3) (1995) 347–354.
- [10] J. Rossignac, A. A. G. Requicha, Offsetting operations in solid modelling, *Computer Aided Geometric Design* 3 (2) (1986) 129–148.
- [11] J. Serra, *Image Analysis and Mathematical Morphology*, Ac. Press, 1988.
- [12] T. Maekawa, An overview of offset curves and surfaces, *Computer-Aided Design* 31 (3) (1999) 165 – 173.
- [13] J. Rossignac, A. Requicha, Constant-radius blending in solid modeling, *ASME Computers In Mechanical Engineering (CIME)* 3 (1984) 65–73.
- [14] J. Rossignac, A. Requicha, Piecewise-circular curves for geometric modeling, *IBM Journal of Research and Development* 3 (1987) 129–148.
- [15] Y. Chen, H. Wang, D. W. Rosen, J. Rossignac, A point-based offsetting method of polygonal meshes (2005).
- [16] J. Rossignac, Ball-based shape processing, in: *DGCI*, 2011, pp. 13–34.

- [17] F. Chazal, A. Lieutier, J. Rossignac, B. Whited, Ball-map: Homeomorphism between compatible surfaces, *Int. J. Comput. Geometry Appl.* 20 (3) (2010) 285–306.
- [18] H. P. Moon, Equivolumetric offset surfaces, *Computer Aided Geometric Design* 26 (2009) 17 – 36.
- [19] H. Hagen, S. Hahmann, Generalized focal surfaces: A new method for surface interrogation, in: *IEEE Visualization, 1992*, pp. 70–76.
- [20] R. L. Foote, The volume swept out by a moving planar region, *Mathematics Magazine* 79 (2006) 289–297.
- [21] D. Rohmer, S. Hahmann, M.-P. Cani, Local volume preservation for skinned characters, *Computer Graphics Forum* 27 (7) (2008) 1919–1927, proceedings of Pacific Graphics.
- [22] M. Meyer, M. Desbrun, P. Schröder, A. H. Barr, Discrete differential-geometry operators for triangulated 2-manifolds (2002).
- [23] M. Desbrun, M. Meyer, P. Schröder, A. H. Barr, Implicit fairing of irregular meshes using diffusion and curvature flow, in: *Proceedings of the 26th annual conference on Computer graphics and interactive techniques, SIGGRAPH '99, 1999*, pp. 317–324.
- [24] D. Zorin, Curvature-based energy for simulation and variational modeling, in: *SMI, 2005*, pp. 198–206.
- [25] J. M. Lee, *Riemannian Manifolds: An Introduction to Curvature*, Springer, 1997.
- [26] J. Rossignac, A. Safonova, A. Szymczak, 3d compression made simple: Edgebreaker on a corner-table, in: *Shape Modeling International Conference, 2001*, pp. 278–283.
- [27] N. Dyn, D. Levine, J. A. Gregory, A butterfly subdivision scheme for surface interpolation with tension control, *ACM Trans. Graph.* 9 (1990) 160–169.
- [28] S. C. Hsu, I. H. H. Lee, N. E. Wiseman, Skeletal strokes, in: *Proceedings of the 6th annual ACM symposium on User interface software and technology, UIST '93, 1993*, pp. 197–206.
- [29] S. Schaefer, T. McPhail, J. Warren, Image deformation using moving least squares, in: *ACM SIGGRAPH 2006 Papers, SIGGRAPH '06, 2006*, pp. 533–540.
- [30] A. H. Barr, Global and local deformations of solid primitives, in: *Proceedings of the 11th annual conference on Computer graphics and interactive techniques, SIGGRAPH '84, 1984*, pp. 21–30.
- [31] J. Rossignac, S. Schaefer, J-splines, *Computer Aided Design* 40 (2008) 1024 – 1032.