Fleshing: Spine-driven Bending with Local Volume Preservation

Wei Zhuo* and Jarek Rossignac *

* Graphics Visualization and Usability Center, College of Computing, Georgia Tech

Abstract
Several design and animation techniques use a one-dimensional proxy C (a spine curve in 3D) to control the deformation or behavior of a digital model of a 3D shape S. We propose a modification of these “skinning” techniques that ensures local volume preservation, which is important for the physical plausibility of digital simulations. In the proposed “fleshing” techniques, as input, we consider a smooth spine C₀, a model S₀ of a solid that lies “sufficiently close” to C₀, and a deformed version C₁ of C₀ that is “not overly bent”. (We provide a precise characterization of these restrictions.) As output, we produce a bijective mapping M, that maps any point X of S onto a point M(X) of M(S). M satisfies two properties: (1) The closest projection of X on C₀ and of M(X) on C₁ have the same arc length parameter. (2) U and M(U) have the same volume, where U is any subset of S. We provide three different closed form expressions for radial, normal and binormal fleshing and discuss the details of their practical real-time implementation.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric transformations F.2.2 [Theory of Computation]: Nonnumerical Algorithms and Problems—Geometrical problems and computations

1. Introduction
Often, the design of a three-dimensional model or of its animation involves bending elongated parts. Models of humans and of various animals are often defined in terms of an articulated skeleton with a few rigid bones connected at (possibly spherical) joints. As the joint angles change, points on the surface or inside the model are moved so as to preserve their relative position with respect to nearby bones. Such “skinning” techniques typically use the arc-length of the closest projection onto individual bones to define relative coordinates and track bone twists (around the bone axis) to fully define the new location of a point after skeletal bending. Points close to a joint may project on more than one bone. The displacements suggested by these different bones are often blended using linear combinations of locations or weighted combinations of rigid motions [LCF00].
Successful techniques have been proposed to increase physical realism of the skin deformations near the joints, so as to more accurately reflect the behavior of skin during the bending of a human elbow [KCZO07]. Some strive to preserve the total volume of the solid near the joint. Other strive to preserve the volume of each slice by possibly tilting the cross-sections [RHC08].

We focus here on deformations that follow the gentle three-dimensional bending of a “spine” that is a smooth curve. Such a tool seems appropriate for bending models of tubes, hoses, wires, ducts, and for approximating the spinal bending of vertebrates (reptiles, fish), and the bending of muscles with no skeletal support, such as mammalian tongues, elephant trumps, octopus arms, or nautilus tentacles. Although such “spine-driven bending” may not be sufficient to model exactly the physically correct behavior of these vertebrae and muscles, it provides an important tool that facilitates the design of useful approximations of their behavior and may have computational advantages over more expensive finite element simulations.

Our main contribution is to propose an explicit mathematical model of spine-driven bending that preserves local volume exactly. By “local” we mean that any solid portion of the initial shape preserves its volume during bending. This objective is more challenging than the preservation of the overall volume (which may for example be achieved by a global scaling or constant distance offsetting [ZR12]) and the preservation of the overall volume of each arbitrary cross-sectional slice using an extension of the Cavalier’s Principle [HS97]. Unfortunately, such global or per-slice volume preservation approaches do not provide a volume preserving mapping (homeomorphism) from the initial shape to the final shape. Hence, in these prior approaches, either it is not clear where exactly in a slice a particular chunk of muscle of the initial shape will end up in the bent model, or, when an exact mapping is defined, the volume of the chunk is not preserved.

When the shape is planar and the bending is in that plane, the 2D problem amounts to preserving the local area. An exact solution to local area preserving bending in 2D has been proposed in [ZR12]. It is based on a local correction, which, after the standard bending, adjusts the normal offset (from the spine) of a point based on the curvatures of the initial and the bent versions of the spine at the corresponding (closest projection) point. Our contribution is to extend this prior 2D solution to 3D, where the spine is a possibly non-planar curve and where the goal is to preserve the local volume instead of the area. The extension to 3D is far from trivial. As illustrated in Fig. 2, the solution proposed in [ZR12] is only valid for a point X0 that lies in the osculating plane of the closest projection Q0 of X0 on the initial spine C0. The solution proposed here does not have this limitation.

1.1. Problem statement and our solutions

The designer starts with a shape S0 in 3D. Our solution is a mapping from a subset of three-space to another and, as such, it operates on any shape (point cloud, bundle of curves, surfaces, solids, meshes, or cell complexes). Still, because our focus is on volume preservation, for clarity, we say that S0 is a solid.

The designer first specifies an initial spine C0. The spine is a smooth curve in 3D that may pierce the solid S0 or not. In fact, an important benefit of our solution is that the spine may be positioned fully outside of the initial solid. Then the designer specifies a new (“bent”) version C1 of the spine.

We restrict our attentions to formulations that are defined by a mapping M which maps each point X0 of S0 to its image X1 = M(X0) in S1. We say that M is a “fleshing” if it satisfies the following conditions:

1. M is fully defined by C0 and C1, and hence independent of S0. This is essential in applications where different versions of S0 may be used with different resolutions or levels of detail.
2. M is an homeomorphism between S0 and S1. This is important because we want the mapping to be invertible: \( M^{-1}(X_1) = X_0 \), where \( M^{-1} \) is defined by the initial spine as \( C_1 \) and the final spine as \( C_0 \).
3. M maps C0 to C1 (i.e., \( M(C_0) = C_1 \)). Because the spines can be adjusted precisely by the user, a solution that ignores such a constraint may be surprising and unnatural.
4. M preserves the arc length along the spine of the closest projection (i.e., \( s_0 = s_1 \), where \( s_i \) is the arc-length parameter of the closest projection of \( P_i \) onto curve \( C_i \)). This constraint restricts the mapping to respect cross-sections. Although this constraint seems natural, it may
not be physically correct for some materials that are capable of stretching along the spine [ACWK04]. Nevertheless, this constraint is key to the effectiveness of our solution and a proper assumption if the spine is rigid (only capable of bending and twisting, but not stretching).

5. $M$ preserves volume locally (i.e., $\text{vol}(U) = \text{vol}(M(U))$ for any subset $U$ of $S_0$). This is important for the physical plausibility of digital simulations, especially when they involve interactions between evolving solids (swimming creature) and surrounding, incompressible fluids.

1.2. Contributions

We propose three different fleshings that satisfy all of the constraints defined in the previous subsection. We call them “radial”, “normal”, and “bi-normal”. We provide the explicit and mathematically exact expression for each one of these fleshings and explain its derivation.

During interactive manipulation or animation, these fleshings may be computed in real time, at each frame, and animated as the user manipulates the spines. Hence, we advocate their use for gaming and medical simulations where live animation of bending shapes are desired.

Our three solutions produce results that are qualitatively different. To clearly illustrate these differences and help the reader decide which one is appropriate for a particular application, we show in Fig. 4 and 5 comparisons of their effects when $S_0$ is a tube or extruded cross-section around $C_0$. We also show in Fig. 1, 7, 6 and 8 their effect on a solid bounded by an arbitrary triangle or quad meshes. We require that $C_0$ and $C_1$ be smooth. We provide, for each fleshing, the precise formulation of a valid space in which $S_0$ must be contained for $M$ to exist as a valid fleshing.

2. Prior Art

The basic deformation operations proposed by Barr [Bar84] extend the conventional operations of affine transformation and CSG to include planar curve-based bending, which preserves the normal offset distance from the spine curve. The resulting mapping is not locally volume preserving as there are local expansion on the convex side and contraction on the concave side of the bent spine. To address this shortcoming, Chirikjian [Chi95] presents a mathematically precise, closed-form solution: for locally area preserving bending in 2D, the offset distance is computed as a root of a quadratic equation with curvature-based coefficients. This variable offset distance allows the shape in the concave side of the spine to grow in the normal direction in order to compensate for the area loss. Moon [Moo08] derives the same quadratic formula for milling with constant material removal rate.

In character animation, Lewis et al. [LCF00] propose generalized forms of skeleton-driven deformations as scattered interpolations. Kavan and colleagues [KCZO07] present the dual quaternion blending as an effective approach to preserve the skinning mesh’s rigidity and roughly its local volume around the joint. Constant volume deformations may be defined by a divergence-free vector field as proposed by von Funck et al. [vFTS06]. Angelidis and Singh [AS07] present the computation of divergence-free vector fields induced by skeletal motion. Their framework requires time integration as physically based rigging [CBC*07] and may have computational disadvantages for high resolution meshes. Rohmer and colleagues [RHC08] compute the offset distance scaled by the skinning weight based on affinity and bone-length. To avoid self-intersection, they detect if an offset point is within its region determined by automatic segmentation. Their subsequent work [RHC99] allows the user to specify the locality of the compensation through 1D profile curves that represent isotropic inflation, bulging, or rippling effects.

A classic theorem due to Steiner [Ste40] establishes the relationship between the differential properties of the surface and the volume enclosed. Thus, if we wish to preserve the total volume, we can grow or shrink the shape uniformly (via constant distance normal offsetting rather than global scaling) in one step (without iteration) [ZR12]. Note that this approach minimizes Hausdorff error and may hence be preferred over global rescaling [DMSB99]. It provides an more efficient algorithm for preserving the total volume of a solid undergoing free-form deformation [HML99], or for compensating the volume change due to advection [KLL*07]. To preserve the details of a shape during deformation, one may use registration with the extracted skeleton [STG97], or with a lower level subdivision model or base surface. Botsch and Kobbelt [BK03] propose to keep the displacement volumes locally constant through relaxation during a deformation of the base surface. Moon [Moo09] presents a closed-form solution for the variable offset distance from a surface that preserves the local volume.

3. Preliminaries

3.1. Locally Volume-preserving Mapping

We consider a bijective mapping $M : X_0 \rightarrow X_1$ that maps any point $X_0 = P_0(x_0)$ onto $X_1 = P_1(x_1)$, where $P_0$ and $P_1$ are themselves mappings from local parameters $x_0$ and $x_1$ onto Cartesian coordinates. (We use $P^{-1}$ to denote the inverse of a mapping $P$). The local parameters can be the arc length, radial offset distance and the angle between the offset direction and the Frenet normal. The mapping $M$ is volume-preserving (i.e. divergence-free) if the Jacobian determinant, $\text{det}(J(M))$, equals 1 [Chi95]. We compute the Jacobian of $M$ by the following equation:

$$ J(M) = \frac{\partial X_1}{\partial X_0} = \frac{\partial P_1}{\partial x_1} \frac{\partial x_1}{\partial x_0} \frac{\partial x_0}{\partial P_0}. \quad (1) $$
In order to perform skinning, one computes the local frame $3.2. \text{Frenet versus twist-compensated local frames}$

In order to perform skinning, one computes the local frame $F_0$ of $C_0$ at the point $C_0(s)$ that is closest to $X_0$, registers $X_0$ to $F_0$, which amounts to computing the local coordinates, and then computes the local frame $F_1$ of $C_1$ at the point $C_1(s)$ and constructs $X_1$ from $C_1(s)$ using the local coordinates.

Typically, we select frames that are aligned with the tangent to the spine. Hence, we pick $T_0(s)$ as the tangent to $C_0$ at $C_0(s)$ and $T_1(s)$ as the tangent of $C_1$ at $C_1(s)$. The remaining issue is how to determine the other two basis vectors, or their twist around the tangent. A natural candidate for the local frame is the Frenet frame $\{T(s), N(s), B(s)\}$ at $C(s)$ where $N(s)$ and $B(s)$ are the normal and binormal. By Frenet-Serret theorem [dC92], the derivative of the Frenet frame at $C(s)$ is related to the frame itself through the curvature $\kappa$ and the torsion $\tau$ at $C(s)$.

\[
\begin{bmatrix}
T'(s) \\
N'(s) \\
B'(s)
\end{bmatrix} = 
\begin{bmatrix}
0 & \kappa & 0 \\
-\kappa & 0 & \tau \\
0 & -\tau & 0
\end{bmatrix}
\begin{bmatrix}
T(s) \\
N(s) \\
B(s)
\end{bmatrix}.
\tag{2}
\]

Although the Frenet frame provides a convenient local frame along the curve, it is not appropriate as the tool for skinning, because it contains undesired twists, as shown in Fig. 3 (left). For example, the Frenet frame has an orientation discontinuity along a piecewise circular curve [RR87] at the $C^1$ continuous junction between two adjacent, but not coplanar circular arcs. To address this problem, we use a “twist-compensated local frame”, as shown in Fig. 3 (right). Its rotation with respect to the Frenet frame is defined by the integral of the torsion [Sab99] [FHH03]. We construct the twist-compensated normal $I'(s+ds)$ at $C(s+ds)$ by projecting $I(s)$ to the normal plane of $C(s+ds)$. Therefore, given an initial normal $I(0)$, the twist-compensated normal $I'(s)$ is obtained by propagation from $I(0)$. Then for each point $X_0$ of $S_0$, we register it with the twist-compensated cross-sectional frame $W_0(s) = \{I_0(s), J_0(s)\}$ on $C_0(s)$.

### 3.3. Overview of the fleshing algorithm

We are given a solid $S_0$, an initial spine $C_0$, and a final spine $C_1$. We are also given an initial normal vector $I_0(0)$ to $C_0$ at $C_0(0)$ and an initial normal vector $I_1(0)$ to $C_1$ at $C_1(0)$. Alternatively, we compute $I_0(0)$ and $I_1(0)$ automatically, using an agreed upon rule for generating a vector normal to a tangent direction, and let the designer control the global twist angle $\omega$ which we use to adjust $I_1(0)$ by rotating it around the tangent to $C_1$ at $C_1(0)$. We assume that each point of $S_0$ has a unique closest projection on $C_0$ and that $C_1$ satisfies our validity conditions. We compute the best version $S_1$ of $S_0$ by applying a fleshing to every vertex or control point $X_0$ of $S_0$ to obtain its image $X_1 = M(X_0)$. Our approach involves the following steps:

1. **Registration:** For each vertex $X_0$ of $S_0$, we compute the arc-length parameter $s_0$ of its closest projection $Q_0 = C_0(s_0)$ and the corresponding Frenet cross-sectional frame $F_0 = \{N_0, B_0\}$ and twist-compensated frame $W_0 = \{I_0, J_0\}$. We compute the local parameters $s_0$ of $X_0$ in $F_0$.

2. **Unbending:** We compute the image $x_0$ of $X_0$ produced by a local volume preserving unbending.

3. **Transfer:** We compute the parameters of $x_0$ in $W_0$, compute the corresponding twist compensated frame $W_1$ at $s_0$ on $C_1$, use these parameters in $W_1$ to produce a point, and compute the parameters $x_1$ of that point in the corresponding cross-sectional Frenet frame $F_1$ on $C_1$.

4. **Bending:** We compute the image $x_1$ of $x_0$ produced by bending a straight line to obtain curvature $k_1$ of $C_1$ at $Q_1 = C_1(s_0)$. Then we use the parameters $x_1$ in $F_1$ to construct the image $X_1 = M(X_0)$.

In Sec. 4, we provide three versions of unbending $x_u = unbend(x_0, k_0)$ and of bending $x_1 = bend(x_u, k_1)$.

### 3.4. Validity conditions

To express the validity conditions under which our approach produces a fleshing, we define a valid space $S(C_0, C_1)$ which must contain $S_0$. To do so, we define the “reach” $R(C)$ of a curve $C$ as the locus of all points that have a unique normal projection onto that curve. The reach may be computed as the space obtained by radially inflating the curve at each point and in all orthogonal directions until we reach the corresponding curvature axis (which is the axis of the osculating circle). We define $S(C_0, C_1)$ as the intersection $R(C_0) \cap M^{-1}(R(C_1))$ of the reach of $C_0$ with the pre-image of the reach of $C_1$. In Sec. 4, we provide explicit formulae for testing, during unbending and bending whether a point $X_0$, is in the valid space.

### 4. Fleshings

We present three fleshings: radial, normal and binormal. For each, we discuss a general derivation satisfying \(\det(J(M)) = 1\), and special cases for unbending and bending.
4.1. Radial Fleshing

Radial fleshing is denoted by \( M_r \). We start with the point \( X \) parameterized by \( (s, r, \theta) \), \( s \) is the arc length parameter of the closest projection \( C(s) \) of \( X \) onto \( C \). \( r \) and \( \theta \) are the polar coordinates of \( X \) on the normal plane of \( C(s) \):

\[
X = P(s, r, \theta) = C(s) + r \cos \theta N(s) + r \sin \theta B(s).
\]

We take the derivative of \( X \) with respect to its parameters and substitute \( T', N' \) and \( B' \) by using Frenet-Serret equation (Eq. 2), and reduce the result to

\[
\frac{\partial X}{\partial (s, r, \theta)} = \begin{bmatrix}
(1 - \kappa r \cos \theta) & -\tau r \sin \theta & \tau r \cos \theta \\
0 & \cos \theta & \sin \theta \\
0 & -\tau r \sin \theta & r \cos \theta
\end{bmatrix} \begin{bmatrix}
T(s) \\
N(s) \\
B(s)
\end{bmatrix}
\]

(3)

Therefore, we have:

\[
det\left( \frac{\partial X}{\partial (s, r, \theta)} \right) = r(1 - \kappa r \cos \theta).
\]

(4)

In \( M_r : X_0 \rightarrow X_1 \), we assume that only the radial offset distance \( r \) is updated from \( r_0 \) to \( r_1 \) while other parameters remain the same. We solve for \( r_1 \) under the constraint that \( det(J(M_r)) = 1 \). We next show that there exists an closed-form solution for \( r_1 \), and hence an analytic solution for \( M_r \).

Specifically, from Eq. 1 we have:

\[
J(M_r) = \frac{\partial X_1}{\partial (s, r_1, \theta_1)} \frac{\partial X_0}{\partial (s, r_0, \theta_0)} = det(J(M_r))^{-1}.
\]

(5)

Given Eq. 4, we compute the determinant of the Jacobian in Eq. 5 as:

\[
det(J(M_r)) = det\left( \frac{\partial X_1}{\partial (s, r_1, \theta_1)} \right) \frac{dr_1}{dr_0} \left( \frac{\partial X_0}{\partial (s, r_0, \theta_0)} \right) = r_1 \left( 1 - \kappa_1 r_1 \cos \theta_1 \right) \frac{dr_1}{dr_0} = r_1 \left( 1 - \kappa_0 \cos \theta_0 \right) \frac{dr_0}{dr_0}
\]

In order to let \( det(J(M_r)) = 1 \), we solve the following ODE:

\[
\frac{dr_1}{dr_0} - \kappa_1 r_1^2 \cos \theta_1 = \frac{dr_0}{dr_0} - \kappa_0 r_0^2 \cos \theta_0,
\]

and integrate from \( 0 \) to \( 1 \) on both sides of the above equation to obtain:

\[
-\frac{2}{3} \kappa_1 \cos \theta_1 r_1^3 + r_1^2 = -\frac{2}{3} \kappa_0 \cos \theta_0 r_0^3 + r_0^2.
\]

(6)

Therefore, \( r_1 \) is a cubic root of Eq. 6 with coefficients specified by \( r, \kappa_0, \kappa_1 \) and \( \cos \theta \). The solution provided by Eq. 6 assumes that the bending (change of curvature) does not change the local Frenet frame. To support more general bending, as explained in Sec. 3.3, we split the fleshing into several steps which include unbending (locally at \( C_0(s) \), the spine becomes a straight line) and bending (the spine becomes curved again with the new curvature and normal). Here, we provide the formulae for the unbending and bending steps.

**Radial Unbending:** We first assume that \( C_0(s) \) is straightened into a line (i.e. \( \kappa_1 = 0 \)) and solve for a temporary value \( r_* \):

\[
r_* = r_0 \sqrt{1 - \frac{2}{3} \kappa_0 \cos \theta_0 r_0^3}.
\]

(7)

In order for \( r_* \) to exist, \( \frac{2}{3} \kappa_0 \cos \theta_0 r_0^3 < 1 \). As \( \cos \theta \) varies in \([-1, 1]\), an sufficient condition for \( r_* \) to exist is \( |\kappa_0 r_0| \leq \frac{3}{2} \).

**Radial Bending:** We then bend the straight spine into \( C_1 \) and solve for \( r_1 \) using \( r_* \):

\[
-\frac{2}{3} \kappa_1 \cos \theta_1 r_1^3 + r_1^2 = r_*^2.
\]

(8)

We normalize the unknown and the coefficients in Eq. 8. Specifically, let \( \lambda = \frac{1}{r_*} \) and \( \alpha = -\frac{2}{3} \kappa_1 \cos \theta_1 \), then Eq. 8 becomes \( \alpha \lambda^3 + \lambda^2 = 1 \). Let \( f(\lambda) = \alpha \lambda^3 + \lambda^2 - 1 \), which has two local extrema (minimum at \( \lambda_1 = 1 \) and maximum at \( \lambda_2 = -\frac{2}{\alpha} \)). If \( \alpha > 0, \lambda_2 < 0 \), then \( f(0) f(1) < 0 \) and \( f' > 0 \in [0,1] \), and hence there exists a valid solution in \([0,1] \). If \( \alpha > 0, \lambda_2 > 0 \), then a valid solution exists only if \( f(\lambda_2) > 0 \), or equivalently \( \alpha^2 < \frac{2}{9} \). Again since \( \cos \theta \) varies in \([-1, 1]\), an sufficient condition for \( r_1 \) to exist is \( |\alpha| < \frac{2}{3\sqrt{3}} \), or

\[
|\kappa_1 r_*| \leq \frac{1}{\sqrt{3}}.
\]

(9)

and when \( \kappa_1 \) reaches this curvature limit, \( r_1 = \sqrt{3} r_* \).

Fig. 4 illustrates the bending of a straight spine into a circular arc. Fig. 4 (a) is the original spine with two layers of cylindrical tube surfaces. Fig. 4 (b) shows the application of Radial Bend to the original tube. Intuitively, the radial distance increases for points on the inner side of the curved spine in order to compensate for local compression. Excessive bending over the curvature limit in Eq. 9 leads to self-intersection of the tube surface.

4.2. Normal Fleshing

Here, we define the normal fleshing \( M_n \). We consider expressing a point \( X \) in the local Frenet frame as follows:

\[
X = P(s, x, y) = C(s) + xN(s) + yB(s).
\]

We take the derivative of \( P \) with respect to its parameters and substitute the derivatives using Eq. 2:

\[
\frac{\partial X}{\partial (s, x, y)} = \begin{bmatrix}
1 - \kappa x \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
T(s) \\
N(s) \\
B(s)
\end{bmatrix}.
\]

Therefore,

\[
det\left( \frac{\partial X}{\partial (s, x, y)} \right) = 1 - \kappa x.
\]

During normal fleshing, we change the parameter \( x \) from \( x_0 \) to \( x_1 \) while keeping \( s \) and \( y \) constant. Given \( \kappa_0, \kappa_1, x, y \)
and $x_0$, we solve for $x_1$ under the constraint $\det(J(M)) = 1$. Specifically, we have:

$$\det(J(M)) = \det\left(\frac{\partial P_1}{\partial (s,x_1,y)}\frac{dx_1}{dx_0}\right)\det\left(\frac{\partial P_0}{\partial (s,x_0,y)}\right)$$

$$= \left(1 - \kappa_1 x_1\right) dx_1 \left(1 - \kappa_0 x_0\right) dx_0.$$ 

Setting $\det(J(M)) = 1$ gives that:

$$dx_1 - \kappa_1 dx_1 = dx_0 - \kappa_0 x_0 dx_0.$$ 

Integrate on both sides of the above equation and we have:

$$x_1 - \frac{1}{2}\kappa_1 x_1^2 = x_0 - \frac{1}{2}\kappa_0 x_0.$$ 

Therefore, $x_1$ is a quadratic root of Eq. 10 with coefficients specified by $x_0$, $\kappa_0$, $\kappa_1$.

As for Normal Fleshing, Eq. 10 is limited to cases where the local curvature is changed, but the Frenet frame remains constant. To support more general fleshing, as explained in Sec. 4.1, we provide below its decomposition into normal unbending and bending maps, which may be combined with the twist-compensated rotation, as discussed in Sec. 3.3. To solve $x_1$, we break Eq. 10 into two steps:

**Normal Unbending**: Assume that $C_0(s)$ is first straightened ($\kappa_1 = 0$) and we solve for a temporary value $x_s$,

$$x_0 = x_0(1 - \frac{1}{2}\kappa_0 x_0).$$ (11)

As $\frac{dx_0}{ds} \geq 0$, the condition for a valid solution of $x_s$ to exist is $|\kappa_0 x_0| \leq 2$.

**Normal Bending**: We then bend the straight spine into $C_1$ and solve for $x_1$ using $x_s$:

$$\frac{1}{2}\kappa_1 x_1^2 + x_1 = x_s.$$ (12)

Hence, the closed-form solution for $x_1$ is

$$x_1 = \frac{1 - \sqrt{1 - 2\kappa_1 x_s}}{\kappa_1}.$$ (13)

In order for $x_1$ to be valid, we have:

$$\kappa_1 x_s \leq \frac{1}{2},$$

and when $\kappa_1$ reaches this curvature limit, $x_1 = 2x_s$.

Fig. 4 (c) shows the application of Normal Bending to the cylindrical tube surfaces in Fig. 4 (a). As shown in the cross-sectional plot, $M_B$ slides points in the normal direction. Intuitively, the tube surface stretches towards the inner side and shrinks from the outer side of the circular spine in order to compensate for local compression and expansion. When reaching the curvature limit in Eq. 13, the tube surface starts to intersect itself. Note that $M_B$ has a more stringent curvature limit than $M_F$ for the same initial tube surface.

### 4.3. Binormal Fleshing

During the Binormal Fleshing, we adjust the coordinate from $y_0$ to $y_1$ while keeping $s$ and $x$ constant. We then solve for $y_1$ under the constraint $\det(J(M)) = 1$:

$$\det(J(M)) = \frac{(1 - \kappa_1 y_1) dy_1}{(1 - \kappa_0 y_0) dy_0}.$$ (14)

We set $\det(J(M)) = 1$ to obtain,

$$(1 - \kappa_1 y_1) dy_1 = (1 - \kappa_0 y_0) dy_0.$$ (15)

Therefore,

$$(1 - \kappa_1 x) y_1 = (1 - \kappa_0 x) y_0.$$ (16)

This result shows that $y_1$ is linearly related to $y_0$ with the coefficient defined by $\kappa_0$, $\kappa_1$, and $x$.

**Binormal Unbending**: Let $\kappa_1 = 0$, $y_s = y_1$ and we have,

$$y_s = (1 - \kappa_0 x) y_0.$$ (17)

In order for $y_s$ to be valid, we have $\kappa_0 x \leq 1$.

**Binormal Bending**: Let $\kappa_0 = 0$, $y_0 = y_s$, and we have,

$$y_1 = \frac{1}{1 - \kappa_1 y_s}.$$ (18)

In order for $y_1$ to be valid, we have

$$\kappa_1 y_s < 1.$$ (19)

When $\kappa_1$ reaches this curvature limit, $y_1$ becomes unbounded.

Fig. 4 (d) shows the application of Binormal Bending to the cylindrical tube surfaces in Fig. 4 (a). As shown in the cross-sectional plot, $M_B$ only allows stratification in the binormal direction: points on the tube surface expand or shrink bilaterally on the inner side or the outer side of the circular spine. When reaching the curvature limit in Eq. 17, the tube surface becomes flat on the inner side. Note that $M_B$ has the least stringent curvature limit among the three solutions.

**Discussion**: Note that Eq. 6, Eq. 10 and Eq. 14 are symmetric in the initial and final states of the spine and the space point. Hence, the mappings are homeomorphisms between $S_0$ and $S_1$. Fig. 4 presents a qualitative comparison of the three fleshing solutions by showing their effects on tubular surfaces: Bi-normal fleshing is closest to what happens when a tube is bent horizontally a bit too much: the flesh is pushed vertically up or down (as in the crack of a bent elbow). The normal fleshing is the reverse: the flesh moves horizontally, hence it moves more quickly in the direction of the center of curvature. The radial is a compromise, the flesh moves radially away or towards the spine.

### 5. Experiments and Results

This section shows the results of our three fleshing solutions.
5.1. Extrusion Models

We first present the application of fleshing to models of solids produced by sweeping a user specified planar cross-section along a smooth 3D spine curve. To better show the different effects of our three solutions, we apply them to extrusions along spines that are circular arcs. In the 2D view as shown in Fig. 5, the user draws a contour and indicates the point $Q = Q_0 = Q_1$ at which the initial and the final spines are aligned and have the same tangent, but different radii and osculating planes. The centers of the arcs are specified by locations $O_0$ and $O_1$. We show the initial cross-section in blue, then for each fleshing, we show the result of unbending in green and in red the result of bending the green in Fig. 5(b). The vectors $O_0 Q$ and $O_1 Q$ define the Frenet frames and curvatures. We assume here no twist (i.e., $\theta_1 = \theta_0$). Notice that the radial fleshing nearly preserves straight lines (even though it is not an affine map). In Fig. 5(c), we show the results in 3D and the corresponding statistics in Tab. 1.

We compute the exact volumes of the extrusion models in all cases using the following approach. Let $W$ be the centroid of a planar region $R$ and let $Q$ be the point passed by the arc $C$ with length $l$ and center $O$. Then the volume of the solid $S$ swept out by $R$ along $C$ is computed as [Foo06],

$$\text{vol}(S) = \text{area}(R) \rho l$$

where $\rho = \frac{\text{OW} \cdot OQ}{\text{OW}}$ is the ratio of the actual distance traveled by the centroid and $l$. If $C$ is a line, $\rho = 1$. We compute the relative error $\epsilon$ of the solid $S$ as

$$\epsilon = \frac{\text{vol}(S_1) - \text{vol}(S_0)}{\text{vol}(S_0)}$$

where $S_i$ is the solid swept out by $R_i$ along $C_i$, $i = 0, 1$. As shown in Tab. 1, models without fleshing have relatively large volumetric errors (3%-5%). The other 6 models with fleshing have nearly the same volume with very small (less than 0.03%) volumetric errors.

<table>
<thead>
<tr>
<th></th>
<th>area($R$)</th>
<th>$\rho$</th>
<th>vol($S$)</th>
<th>$\epsilon$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original</td>
<td>0.278</td>
<td>1.032</td>
<td>1.436</td>
<td>0</td>
</tr>
<tr>
<td>Unbend</td>
<td>0.278</td>
<td>1</td>
<td>1.391</td>
<td>-0.031</td>
</tr>
<tr>
<td>Bend</td>
<td>0.278</td>
<td>0.982</td>
<td>1.366</td>
<td>-0.049</td>
</tr>
<tr>
<td>Radial Unbend</td>
<td>0.287</td>
<td>1</td>
<td>1.436</td>
<td>-1.36E-5</td>
</tr>
<tr>
<td>Radial Bend</td>
<td>0.296</td>
<td>0.969</td>
<td>1.436</td>
<td>-3.97E-6</td>
</tr>
<tr>
<td>Normal Unbend</td>
<td>0.287</td>
<td>1</td>
<td>1.436</td>
<td>-5.63E-5</td>
</tr>
<tr>
<td>Normal Bend</td>
<td>0.298</td>
<td>0.963</td>
<td>1.436</td>
<td>-1.03E-4</td>
</tr>
<tr>
<td>Binormal Unbend</td>
<td>0.287</td>
<td>1</td>
<td>1.436</td>
<td>1.17E-4</td>
</tr>
<tr>
<td>Binormal Bend</td>
<td>0.293</td>
<td>0.979</td>
<td>1.437</td>
<td>2.83E-4</td>
</tr>
</tbody>
</table>

Table 1: Statistics of the cross-sectional areas, ratios of the centroid traveled distance to $l$, solid volumes and their relative errors for extruded models in Fig. 5(c).

Figure 4: The deformation of two layers of tube surfaces driven by bending a straight spine into a circular arc. We show three types of fleshing to the original tube surfaces. Each shows the transverse (left) and the frontal (right) views of the bent tubes, and the cross section is dynamically plotted on the top-right. The red marks show the selected mapping of points.
5.2. Quad and Triangle Meshes

We show fleshing applications to general shapes and report total volume changes of less than 0.3%. Fig. 1 shows bending a triangle mesh representing a bunny.

Fig. 6 shows bending a genus-2 quad mesh first with a vertical spine into the frontal plane, then with a horizontal spine out of the frontal plane. Initially, the red and the blue spines are on the plane that divides \( S_0 \) into identical halves. Without the volume-preserving mapping, the volume remain unchanged after the first bending, but increases by 9% after the second bending. In comparison, the volume deviates little (<0.1%) from the orginal one if using fleshing. However, the binormal stretch (Fig. 6(b) right) causes the mesh to grow unexpectedly in the horizontal direction. Fig. 7 shows
an application in animation, where a dolphin mesh is sliding along a 3D curve (produced by changing the origin of the arc-length for $C_1$) with its orientation and deformation determined by the curve. Here the curve, $C_1$, represents an animator-specified path for the digital model to follow. The path may be curved so that the model may bend. Without fleshing, this causes unexpected changes of volume perceivable by the viewer. The volume of the digital model is preserved by one of our fleshing mappings. As shown in the figure, the volume deviation is above 5% without fleshing and reduced to 0.01% with the radial method.

5.3. Resolution and Accuracy

This section discusses the impact of sampling density on the accuracy of (local) volume preservation implemented by fleshing. Fig. 8 shows bending a subdivision mesh at different resolutions. The increase of the subdivision depth greatly decreases the relative volume errors of the three fleshing mappings (from 2.5% to 0.001%). On the contrary for skinning, radial, normal and binormal on different levels of a subdivision mesh. From top to bottom, the number of vertices are 32, 482, 1922.

![Figure 8: From left to right we show the bending results of skinning, radial, normal and binormal on different levels of a subdivision mesh. From top to bottom, the number of vertices are 32, 482, 1922.](image)

![Figure 9: Bending a cloud of cubes of uniform size. From left to right, the original cube sizes are 15, 22, 30.](image)

of fleshing over skinning is to report the average of the absolute volume errors of the small chunks. Fig. 9 shows bending a cloud of cubes at different sizes. We compute the volume of each cube deformed by the spine. The relative error for each cube is computed as $\epsilon = (v - v_0)/v_0$. We report the percentage mean absolute value, $\epsilon_{\text{mean}}$, of the relative errors for all cubes. Fig. 10 shows the plot of $\epsilon_{\text{mean}}$ versus the cube size. In general, the relative error scales with the cube size in all three fleshing mappings. The volume error reported for large cubes comes from approximating the curved shape of the bent cube by a polyhedron that interpolates the images of the vertices of the initial cube.

6. Discussion

Fleshing presents three closed-form volume preserving mappings, which depend on a proper local parameterization of the 3D shape along the 1D spine. The spine may have a simple parametric expression, such as circular or helical arc. Then the closed-form parameterization along the curve is easy to obtain. However, this limits the designer’s ability to bend the spine by manipulating control points. If we represent the spine by a interpolating polynomial, we must be able to compute the arc-length parameter of the closest projection of a point onto the spine efficiently.

The choice of representation for the spine is orthogonal to our contribution. Nevertheless, we support two formulations for the spine $C$: (1) a low degree, interpolating polynomial, which we evaluate using Neville’s algorithm, as shown in Fig. 1, and (2) quintic NUBS, which we evaluate using de Casteljau’s algorithm, as shown in Fig. 7. $C_0$ and $C_1$ are polygonal approximations of such smooth curves. We sample them so as to ensure a constant distance, $d$, between consecutive samples. Again, let $Q_0$ be the closest projection of $X_0$ on $C_0$. Assume that $Q_0$ is on the edge $C_0[k][k+1]$: $Q_0 = C_0[k] + (aC_0[k][k+1])$: $Q_0 = C_0[k] + (aC_0[k][k+1])$: $0 < a < 1$.

The arc-length parameter $s$ of $Q_0$ is $s = (k + a)d$. We use the same arc-length parameter to compute $Q_1$ on $C_1$: $Q_1 =$
$\alpha C_1[k] + \alpha C_1[k + 1]$, which is used as the anchor point for computing $X_j$, as described in the overall fleshing transformation. The influence of this polyline approximation depends on the sampling step size ($d$). When $d$ is not overly small, decreasing $d$ improves the precision for locally volume preservation. Angular distortion exists in fleshing mappings due to that the mappings are not conformal. Also, it is not possible to deform B-spline surfaces with locally volume preservation by applying the mapping only to its control points as the mappings are not affine. However, fleshings preserve smoothness, and hence also sharp features. In fact, the radial fleshing is nearly line preserving, as shown in Fig. 5(b) (left). The result is guaranteed to be free from self-crossing (when we are within the validity conditions), and hence it will not produce new sharp features.

7. Conclusion

We have proposed three formulations for deforming a shape driven by bending a spine. Our fleshing solutions ensure that the local volume of any subset of a valid space is preserved during the bending. Furthermore, our solution is based on a closed form mapping of space and depends neither on the initial shape nor on the given global coordinate system. Hence the fleshing may be applied to any shape topology (point cloud, watertight surfaces or cell complexes). Furthermore, we extend our approach to free form spines, by propagating a twist compensated local frame and by letting the user or an application control the bending and twisting. We hope that the simplicity, accuracy and performance of the proposed fleshing approach will make it a standard bending tool for many modeling and animation applications where local volume preservation is desired.

References


© 2013 The Author(s)
© 2013 The Eurographics Association and Blackwell Publishing Ltd.