Variational Skinning of an Ordered Set of Discrete 2D Balls

Greg Slabaugh[1], Gozde Unal[2], Tong Fang[1], Jarek Rossignac[3], Brian Whited[3]

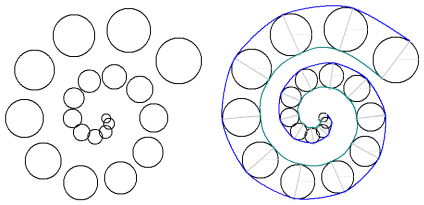
[1] Siemens Corporate Research, Princeton NJ 08540 USA
[2] Sabanci University, Istanbul Turkey
[3] Georgia Institute of Technology, Atlanta, GA 30332 USA

Abstract. This paper considers the problem of computing an interpolating skin of a ordered set of discrete 2D balls. By construction, the skin is constrained to be C^1 continuous, and for each ball, it touches the ball at a point and is tangent to the ball at the point of contact. Using an energy formulation, we derive differential equations that are designed to minimize the skin's arc length, curvature, or convex combination of both. Given an initial skin, we update the skin's parametric representation using the differential equations until convergence occurs. We demonstrate the method's usefulness in generating interpolating skins of balls of different sizes and in various configurations.

1 Introduction

In this paper, we consider the geometric problem of *ball skinning*, which we define to be the computation of a continuous interpolation of a discrete set of balls; an example appears in Figure 1. This problem arises in numerous applications, including character skinning, molecular surface model generation, and modeling of tubular structures. The balls can have different radii, be configured in different positions, and may or may not overlap. In our formulation of the problem, we require that there exist a series of contact points arranged along the skin, so that each ball has a point of contact with the skin and that the skin and ball are tangent to each other at the point of contact. The skin then rests on and interpolates the underlying balls.

For a given configuration of balls, there exist an infinite number of possible solutions to this problem as expressed above. To formulate the problem so that it is well-posed, we seek the skin that has minimal arc length, curvature, or combination of both. We achieve this by deriving, and solving, differential equations that minimize an energy, composed of arc length and curvature terms, based on this constrained variational problem. By minimizing this energy, the method provides an optimal constrained interpolation of the balls. In this paper we consider both *one-sided* and *two-sided* skins. A one-sided skin is a contour that rests on one side (left or right) of a collection of balls such as that portrayed in Figure 2 (a), while a two-sided skin defines an interpolating region that is composed of both left and right skins, and has an inside and outside, as demonstrated in Figure 1.



(a) (b)

Fig. 1. An example ball skinning. Given an ordered sequence of balls (a), we produce a skin that optimally interpolates the balls (b). This skin consists of two splines (green and blue) and is computed using differential equations.

1.1 Related Work

The problem of skinning appears in various contexts. In computer animation, often an articulated object or character is constructed using a layered representation consisting of a skeletal structure and a corresponding geometric skin [1]. The skeleton has fewer degrees of freedom and is simpler to adjust by an animator. Given a new skeletal pose, the skinning algorithm is responsible for deforming the geometric skin to respond to the motion of the underlying skeleton. The skinning problem is a special case of the problem of computing the envelopes of families of quadrics, which have been investigated by Paternell [2] via the use of cyclographic maps.

The problem of ball skinning appears frequently in the context of computational chemistry and molecular biology, when generating surface meshes for molecular models [3] [4] [5]. Several algorithms exist to skin a molecular model to produce a C^1 continuous surface that is tangent smooth and has high mesh quality. These methods are typically either based on Delaunay triangulation [3] or by finding the isosurface of an implicit function [5]. The work of [5] derives a special subset of skins that is piece-wise quadratic. When dealing with a continuous family of balls, the skin is the envelope of the infinite union of the circles of intersection of two consecutive pearls of infinitely close center. While the surfaces generated by these methods are tangent to the balls and have smoothness at the point of tangency, none of these existing methods provide an optimally smooth skin, unlike the method we present here.

In our application, we are interested in modeling the geometry of a blood vessel that has been identified using a 2D variant of Pearling [6], a ball packing algorithm that places numerous balls of different radii so that they fit snugly inside an imaged blood vessel. Given these balls, we would like to find an smooth, C^1 skin that smoothly interpolates the balls. This surface can then be used for visualization of the blood vessel as well as measurements such as volume or surface area.

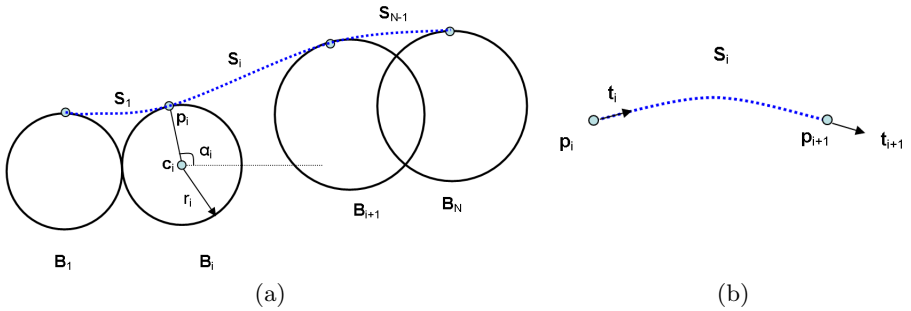


Fig. 2. Depiction of the problem. In (a), we would like to find the skin (dotted curve) that interpolates the ordered set of balls. In (b), we show a depiction of a segment of the skin.

1.2 Our Contribution

We model the skin as a C^1 spline, which, by construction, must touch each ball at a point of contact and be tangent to the ball at the point of contact. We then provide two novel derivations, one for deforming this constrained spline to minimize its arc length; and a second derivation for minimizing its curvature. The result of these derivations are differential equations, which we then solve to update a given spline to its optimal position. We then show experimental examples of how these differential equations are used perform *optimally smooth* skinning of balls.

2 Skin representation

In this paper we consider the problem of *smoothly* interpolating between a discrete ordered set of balls. Our objective is to find a skin $\mathbf{S}$, that satisfies several geometric criteria:

1. The skin should be modeled by a point of contact with each ball.
2. The skin should be tangent to a ball at the point of contact.
3. The skin should optimize an energy functional composed of arc length and curvature terms.

A depiction of the problem is presented in Figure 2 (a), where the desired skin is rendered as a dotted line. The skin $\mathbf{S}$ is composed of a set of segments, $\mathbf{S}_i$, for $i = 1 \ldots N-1$, where N is the total number of balls.

2.1 Segment representation

Various spline representations (such as Catmull Rom, 4-point, etc.) are possible for modeling the segments; in this paper we choose to model each segment $\mathbf{S}_i$

using a spline that starts at point $\mathbf{p}_i$ in direction $\mathbf{t}_i$, and ends at point $\mathbf{p}_{i+1}$ in direction $\mathbf{t}_{i+1}$, as depicted in Figure 2 (b). We model the segment using a cubic polynomial curve

$$\mathbf{S}_i = \mathbf{A}_i t^3 + \mathbf{B}_i t^2 + \mathbf{C}_i t + \mathbf{D}_i, \tag{1}$$

since the four constraints require four degrees of freedom. For the ith segment, $\mathbf{A}_i$, $\mathbf{B}_i$, $\mathbf{C}_i$, and $\mathbf{D}_i$ are coefficients, and $t \in [0, 1]$ is a time variable that parameterizes the curve.

Each segment $\mathbf{S}_i$ of the skin is defined by the Hermite interpolation of the boundary conditions defined by $\mathbf{p}_i$, $\mathbf{t}_i$, $\mathbf{p}_{i+1}$, and $\mathbf{t}_{i+1}$, specifically, $\mathbf{S}_i|_{t=0} = \mathbf{p}_i$, $\frac{d\mathbf{S}_i}{dt}|_{t=0} = \mathbf{t}_i$, $\mathbf{S}_i|_{t=1} = \mathbf{p}_{i+1}$, and $\frac{d\mathbf{S}_i}{dt}|_{t=1} = \mathbf{t}_{i+1}$. With these constraints, and the derivative of the segment,

$$\mathbf{S}'_i = \frac{d\mathbf{S}_i}{dt} = 3\mathbf{A}_i t^2 + 2\mathbf{B}_i t + \mathbf{C}_i, \tag{2}$$

we obtain a system of four equations for the four coefficients: $\mathbf{D}_i = \mathbf{p}_i$, $\mathbf{C}_i = \mathbf{t}_i$, $\mathbf{A}_i + \mathbf{B}_i + \mathbf{C}_i + \mathbf{D}_i = \mathbf{p}_{i+1}$, and $3\mathbf{A}_i + 2\mathbf{B}_i + \mathbf{C}_i = \mathbf{t}_{i+1}$, which is easily solved, yielding

$$\begin{aligned}
\mathbf{A}_i &= -2\mathbf{p}_{i+1} + 2\mathbf{p}_i + \mathbf{t}_i + \mathbf{t}_{i+1} \\
\mathbf{B}_i &= 3\mathbf{p}_{i+1} - 3\mathbf{p}_i - 2\mathbf{t}_i - \mathbf{t}_{i+1} \\
\mathbf{C}_i &= \mathbf{t}_i \\
\mathbf{D}_i &= \mathbf{p}_i
\end{aligned} \tag{3}$$

2.2 Endpoints

We now have a way to model each segment of the skin. But we have not yet described how to determine the endpoints $\mathbf{p}_i$, $\mathbf{p}_{i+1}$ and their respective tangents, $\mathbf{t}_i$, $\mathbf{t}_{i+1}$ of each segment. As shown in Figure 2 (a), we can represent the point of contact $\mathbf{p}_i$ on the ith ball as

$$\mathbf{p}_i = \mathbf{c}_i + \begin{bmatrix} r_i \cos \alpha_i \\ r_i \sin \alpha_i \end{bmatrix}, \tag{4}$$

where r_i is the radius of ball, $\mathbf{c}_i$ is its center, and α_i is an angle. In addition, we can represent the tangent

$$\mathbf{t}_i = \begin{bmatrix} -a_i \sin \alpha_i \\ a_i \cos \alpha_i \end{bmatrix}, \tag{5}$$

where a_i is a stiffness coefficient that controls the influence of the tangential constraint. Each a_i is fixed to be half the distance between the next and previous ball centers (for the first and last balls, it is the distance between the ball center and its neighbor ball center). Note that both the point $\mathbf{p}_i$ and the tangent $\mathbf{t}_i$ are only a function of the angle α_i, since the radius, center, and stiffness coefficient of the ball are fixed.

We now have a way to represent the skin $\mathbf{S}$ as a set of segments $\mathbf{S}_i$, where each segment $\mathbf{S}_i$ interpolates between the points of contact $\mathbf{p}_i, \mathbf{p}_{i+1}$ with balls $\mathbf{B}_i, \mathbf{B}_{i+1}$, subject to tangent conditions $\mathbf{t}_i, \mathbf{t}_{i+1}$ respectively.

By construction of the problem, the angle α_i affects only the segment $\mathbf{S}_i$ as well as the segment $\mathbf{S}_{i-1}$, as can be easily seen in Figure 2 (a). Finally, we note that the skin is fully parameterized by the balls and the spline angles α_i. Since the balls are fixed, our objective will be to compute the angles α_i that form the optimal skin.

3 Energy minimization

There are an infinite number of skins that are modeled by a contact point on each ball and have a direction tangent to the ball at the point of contact. To further constrain the problem, we require that the skin have minimal arc length and/or be smooth. We achieve this by finding the angles α_i that optimize an energy functional. First, we derive equations used to compute the skin with minimal arc length, then we consider curvature.

3.1 Arc length minimization

In this section, we consider the minimization of arc length. This will result in the shortest skin that satisfies the geometric constraints imposed by our ball representation. That is, we would like to find the angles α_i that minimize

$$E_a = \int |\mathbf{S}'| dt, \tag{6}$$

where $\mathbf{S}'$ is the derivative of $\mathbf{S}$ with respect to t. Since the skin is represented as a set of segments, this is equivalent to

$$E_a = \sum_{i=1}^{N-1} \int |\mathbf{S}_i'| dt, \tag{7}$$

Next, we take the derivative of the energy with respect to the angle α_i. As stated above, the ith angle only affects the segments $\mathbf{S}_{i-1}$ and $\mathbf{S}_i$. Therefore,

$$\frac{\partial E_a}{\partial \alpha_i} = \frac{\partial}{\partial \alpha_i}\left(\int |\mathbf{S}_i'| dt\right) + \frac{\partial}{\partial \alpha_i}\left(\int |\mathbf{S}_{i-1}'| dt\right) \tag{8}$$

First term Let us consider the first term of Equation 8. Propagating the derivative with respect to α_i through the integral, it is easy to show that

$$\begin{aligned}\frac{\partial}{\partial \alpha_i}\left(\int |\mathbf{S}_i'| dt\right) &= \int \frac{\partial}{\partial \alpha_i} < \mathbf{S}_i', \mathbf{S}_i' >^{\frac{1}{2}} dt \\ &= \int < \mathbf{S}_i', \mathbf{S}_i' >^{-\frac{1}{2}} < \mathbf{S}_i', \frac{\partial \mathbf{S}_i'}{\partial \alpha_i} > dt,\end{aligned}$$

where <> denotes an inner product. Next, we derive an expression for the $\frac{\partial \mathbf{S}'_i}{\partial \alpha_i}$ terms using Equation 2, yielding

$$\frac{\partial \mathbf{S}'_i}{\partial \alpha_i} = 3t^2 \frac{\partial \mathbf{A}_i}{\partial \alpha_i} + 2t \frac{\partial \mathbf{B}_i}{\partial \alpha_i} + \frac{\partial \mathbf{C}_i}{\partial \alpha_i} \tag{9}$$

The derivatives $\frac{\partial \mathbf{A}_i}{\partial \alpha_i}$, $\frac{\partial \mathbf{B}_i}{\partial \alpha_i}$ and $\frac{\partial \mathbf{C}_i}{\partial \alpha_i}$ can be derived using Equation 3, as

$$\begin{aligned} \frac{\partial \mathbf{A}_i}{\partial \alpha_i} &= 2\frac{\partial \mathbf{p}_i}{\partial \alpha_i} + \frac{\partial \mathbf{t}_i}{\partial \alpha_i} \\ \frac{\partial \mathbf{B}_i}{\partial \alpha_i} &= -3\frac{\partial \mathbf{p}_i}{\partial \alpha_i} - 2\frac{\partial \mathbf{t}_i}{\partial \alpha_i} \\ \frac{\partial \mathbf{C}_i}{\partial \alpha_i} &= \frac{\partial \mathbf{t}_i}{\partial \alpha_i} \end{aligned} \tag{10}$$

Finally, the derivatives $\frac{\partial \mathbf{p}_i}{\partial \alpha_i}$ and $\frac{\partial \mathbf{t}_i}{\partial \alpha_i}$ can be derived from Equations 4 and 5 as

$$\begin{aligned} \frac{\partial \mathbf{p}_i}{\partial \alpha_i} &= \begin{bmatrix} -r_i \sin \alpha_i \\ r_i \cos \alpha_i \end{bmatrix} \\ \frac{\partial \mathbf{t}_i}{\partial \alpha_i} &= \begin{bmatrix} -a_i \cos \alpha_i \\ -a_i \sin \alpha_i \end{bmatrix} \end{aligned} \tag{11}$$

We now have all the derivatives needed to compute the first term in Equation 8.

Second term Now let us consider the first term of Equation 8, which has a very similar derivation. Propagating the derivative with respect to α_i through the integral

$$\frac{\partial}{\partial \alpha_i} \left(\int |\mathbf{S}'_{i-1}| dt \right) = \int < \mathbf{S}'_{i-1}, \mathbf{S}'_{i-1} >^{-\frac{1}{2}} < \mathbf{S}'_{i-1}, \frac{\partial \mathbf{S}'_{i-1}}{\partial \alpha_i} > dt, \tag{12}$$

As before, we derive an expression for the $\frac{\partial \mathbf{S}'_{i-1}}{\partial \alpha_i}$ terms using Equation 2, yielding

$$\frac{\partial \mathbf{S}'_{i-1}}{\partial \alpha_i} = 3t^2 \frac{\partial \mathbf{A}_{i-1}}{\partial \alpha_i} + 2t \frac{\partial \mathbf{B}_{i-1}}{\partial \alpha_i} + \frac{\partial \mathbf{C}_{i-1}}{\partial \alpha_i} \tag{13}$$

Next, the derivatives $\frac{\partial \mathbf{A}_{i-1}}{\partial \alpha_i}$, $\frac{\partial \mathbf{B}_{i-1}}{\partial \alpha_i}$ and $\frac{\partial \mathbf{C}_{i-1}}{\partial \alpha_i}$ can be derived using Equation 3, as

$$\begin{aligned} \frac{\partial \mathbf{A}_{i-1}}{\partial \alpha_i} &= -2\frac{\partial \mathbf{p}_i}{\partial \alpha_i} + \frac{\partial \mathbf{t}_i}{\partial \alpha_i} \\ \frac{\partial \mathbf{B}_{i-1}}{\partial \alpha_i} &= 3\frac{\partial \mathbf{p}_i}{\partial \alpha_i} - \frac{\partial \mathbf{t}_i}{\partial \alpha_i} \\ \frac{\partial \mathbf{C}_{i-1}}{\partial \alpha_i} &= \mathbf{0} \end{aligned} \tag{14}$$

We now have all the derivatives needed to compute the second term of Equation 8.

3.2 Curvature minimization

We now consider the problem of minimizing curvature. Since curvature can be positive or negative, we choose to minimize the squared curvature, i.e, we would like to find the angles α_i that minimize

$$E_c = \int \kappa^2(t)dt, \tag{15}$$

where $\kappa(t)$ is the curvature of $\mathbf{S}$ at point t. Since the skin is represented as a set of segments, this is equivalent to

$$E_c = \sum_{i=1}^{N-1} \int \kappa_i^2(t)dt, \tag{16}$$

where $\kappa_i(t)$ is the curvature at point t along segment $\mathbf{S}_i$. Next, we take the derivative of the energy with respect to the angle α_i. As stated above, the ith angle only affects the segments $\mathbf{S}_{i-1}$ and $\mathbf{S}_i$. Therefore,

$$\frac{\partial E_c}{\partial \alpha_i} = \frac{\partial}{\partial \alpha_i}\left(\int \kappa_i^2(t)dt\right) + \frac{\partial}{\partial \alpha_i}\left(\int \kappa_{i-1}^2(t)dt\right) \tag{17}$$

Recall that the curvature is given by

$$\kappa_i = \frac{|\mathbf{S}_i^{'} \times \mathbf{S}_i^{''}|}{|\mathbf{S}_i^{'}|^3} = \frac{< \mathbf{S}_i^{'}, J\mathbf{S}_i^{''} >}{< \mathbf{S}_i^{'}, \mathbf{S}_i^{'} >^{\frac{3}{2}}}, \tag{18}$$

where $J = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ is a 90 degree rotation matrix. Using these equations, Equation 17 becomes

$$\frac{\partial E_c}{\partial \alpha_i} = \frac{\partial}{\partial \alpha_i}\left(\int \left[\frac{< \mathbf{S}_i^{'}, J\mathbf{S}_i^{''} >}{< \mathbf{S}_i^{'}, \mathbf{S}_i^{'} >^{\frac{3}{2}}}\right]^2 dt\right) + \frac{\partial}{\partial \alpha_i}\left(\int \left[\frac{< \mathbf{S}_{i-i}^{'}, J\mathbf{S}_{i-i}^{''} >}{< \mathbf{S}_{i-i}^{'}, \mathbf{S}_{i-i}^{'} >^{\frac{3}{2}}}\right]^2 dt\right) \tag{19}$$

First term Let us consider the first term of Equation 19. Propagating the derivative with respect to α_i through the integral, we see that

$$\frac{\partial}{\partial \alpha_i}\left(\int \left[\frac{< \mathbf{S}_i^{'}, J\mathbf{S}_i^{''} >}{< \mathbf{S}_i^{'}, \mathbf{S}_i^{'} >^{\frac{3}{2}}}\right]^2 dt\right) =$$
$$\int 2\left[\frac{< \mathbf{S}_i^{'}, J\mathbf{S}_i^{''} >}{< \mathbf{S}_i^{'}, \mathbf{S}_i^{'} >^{\frac{3}{2}}}\right]\left(\frac{\frac{\partial}{\partial \alpha_i} < \mathbf{S}_i^{'}, J\mathbf{S}_i^{''} >}{< \mathbf{S}_i^{'}, \mathbf{S}_i^{'} >^{\frac{3}{2}}} - \frac{3}{2}\frac{< \mathbf{S}_i^{'}, J\mathbf{S}_i^{''} > \frac{\partial}{\partial \alpha_i} < \mathbf{S}_i^{'}, \mathbf{S}_i^{'} >}{< \mathbf{S}_i^{'}, \mathbf{S}_i^{'} >^{\frac{5}{2}}}\right) dt$$

For this, we need the derivatives $\frac{\partial}{\partial \alpha_i} < \mathbf{S}_i^{'}, J\mathbf{S}_i^{''} >$ and $\frac{\partial}{\partial \alpha_i} < \mathbf{S}_i^{'}, \mathbf{S}_i^{'} >$. It is easy to show that these derivatives are

$$\frac{\partial}{\partial \alpha_i} < \mathbf{S}_i^{'}, J\mathbf{S}_i^{''} > = < \frac{\partial \mathbf{S}_i^{'}}{\partial \alpha_i}, J\mathbf{S}_i^{''} > + < \mathbf{S}_i^{'}, J\frac{\partial \mathbf{S}_i^{''}}{\partial \alpha_i} >$$
$$\frac{\partial}{\partial \alpha_i} < \mathbf{S}_i^{'}, \mathbf{S}_i^{'} > = 2 < \mathbf{S}_i^{'}, \frac{\partial \mathbf{S}_i^{'}}{\partial \alpha_i} > \tag{20}$$

Equation 9 gives an expression for $\frac{\partial \mathbf{S}_i^{'}}{\partial \alpha_i}$, and from this we see $\frac{\partial \mathbf{S}_i^{''}}{\partial \alpha_i} = 6t\frac{\partial \mathbf{A}_i}{\partial \alpha_i} + 2\frac{\partial \mathbf{B}_i}{\partial \alpha_i}$. The derivatives $\frac{\partial \mathbf{A}_i}{\partial \alpha_i}$, $\frac{\partial \mathbf{B}_i}{\partial \alpha_i}$ and $\frac{\partial \mathbf{C}_i}{\partial \alpha_i}$ are given in Equation 10. We now have all the derivatives needed to compute the first term in Equation 19.

Second term The first term of Equation 19 is very similar the second term derived above. Propagating the derivative with respect to α_i through the integral, we see that

$$\frac{\partial}{\partial \alpha_i}\left(\int \left[\frac{< \mathbf{S}_{i-1}^{'}, J\mathbf{S}_{i-1}^{''} >}{< \mathbf{S}_{i-1}^{'}, \mathbf{S}_{i-1}^{'} >^{\frac{3}{2}}}\right]^2 dt\right) = \int 2\left[\frac{< \mathbf{S}_{i-1}^{'}, J\mathbf{S}_{i-1}^{''} >}{< \mathbf{S}_{i-1}^{'}, \mathbf{S}_{i-1}^{'} >^{\frac{3}{2}}}\right] \cdot \tag{21}$$
$$\left(\frac{\frac{\partial}{\partial \alpha_i} < \mathbf{S}_{i-1}^{'}, J\mathbf{S}_{i-1}^{''} >}{< \mathbf{S}_{i-1}^{'}, \mathbf{S}_{i}^{'} >^{\frac{3}{2}}} - \frac{3}{2}\frac{< \mathbf{S}_{i-1}^{'}, J\mathbf{S}_{i-1}^{''} > \frac{\partial}{\partial \alpha_i} < \mathbf{S}_{i-1}^{'}, \mathbf{S}_{i-1}^{'} >}{< \mathbf{S}_{i-1}^{'}, \mathbf{S}_{i-1}^{'} >^{\frac{5}{2}}}\right) dt$$

For this, we need the derivatives $\frac{\partial}{\partial \alpha_i} < \mathbf{S}_{i-1}^{'}, J\mathbf{S}_{i-1}^{''} >$ and $\frac{\partial}{\partial \alpha_i} < \mathbf{S}_{i-1}^{'}, \mathbf{S}_{i-1}^{'} >$. It is well known that the derivative of the cross product is

$$\frac{\partial}{\partial \alpha_i} < \mathbf{S}_{i-1}^{'}, J\mathbf{S}_{i-1}^{''} > = < \frac{\partial \mathbf{S}_{i-1}^{'}}{\partial \alpha_i}, J\mathbf{S}_{i-1}^{''} > + < \mathbf{S}_{i-1}^{'}, J\frac{\partial \mathbf{S}_{i-1}^{''}}{\partial \alpha_i} >$$
$$\frac{\partial}{\partial \alpha_i} < \mathbf{S}_{i-1}^{'}, \mathbf{S}_{i-1}^{'} > = 2 < \mathbf{S}_{i-1}^{'}, \frac{\partial \mathbf{S}_{i-1}^{'}}{\partial \alpha_i} > \tag{22}$$

Equation 13 gives an expression for $\frac{\partial \mathbf{S}_{i-1}^{'}}{\partial \alpha_i}$, and from this we see $\frac{\partial \mathbf{S}_{i-1}^{''}}{\partial \alpha_i} = 6t\frac{\partial \mathbf{A}_{i-1}}{\partial \alpha_i} + 2\frac{\partial \mathbf{B}_{i-1}}{\partial \alpha_i}$. The derivatives $\frac{\partial \mathbf{A}_{i-1}}{\partial \alpha_i}$, $\frac{\partial \mathbf{B}_{i-1}}{\partial \alpha_i}$ and $\frac{\partial \mathbf{C}_{i-1}}{\partial \alpha_i}$ are given in Equation 14. Thus, all the derivatives needed to compute the second term of Equation 19 have been derived.

In these equations, we evaluate the integrals in Equations 8 and 19 for each angle α_i. However, for the first ball, $i = 1$, there is no segment $\mathbf{S}_{i-1}$, so we ignore the integral this term. Likewise, for the last ball, $i = N$, there is no segment $\mathbf{S}_i$, so we ignore the integral this term.

3.3 Discussion

To summarize, we have derived the gradient of energy functionals E_a and E_c with respect to angles, α_i. The derivation inherently consisted of several steps

via the chain rule, as the energy is a combination of the arc length and squared curvature, which in turn are functions of the skin, which in turn is a function of the segment coefficients $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i, \mathbf{D}_i$ and $\mathbf{A}_{i-1}, \mathbf{B}_{i-1}, \mathbf{C}_{i-1}, \mathbf{D}_{i-1}$, which in turn are functions of the angles α_i.

3.4 Implementation

We combine the energies E_a and E_c together, as

$$E = (1-k)E_a + kE_c \tag{23}$$

where k is a constant used to weight the arc length minimization relative to the curvature minimization. Convex combinations of the two can be selected using $k \in [0,1]$. Therefore, the combined energy minimization is given by

$$\frac{\partial E}{\partial \alpha_i} = (1-k)\frac{\partial E_a}{\partial \alpha_i} + k\frac{\partial E_c}{\partial \alpha_i} \tag{24}$$

where $\frac{\partial E_a}{\partial \alpha_i}$ is given in Equation 8 and $\frac{\partial E_c}{\partial \alpha_i}$ is provided in Equation 17. In all of the experiments in this paper, we fix $k = 0.9$, to encourage smoother solutions.

These equations are a set of differential equations that can be used in a gradient descent procedure to optimize the skin by manipulating the angles $\alpha = [\alpha_1, \ldots \alpha_N]^T$. Let α_i^n be the ith angle at iteration n. We can then update the angles by moving them in the negative gradient direction, i.e.,

$$\alpha^{n+1} = \alpha^n - \Delta t \nabla E_{\alpha^n}, \tag{25}$$

where Δt is a time step and $\nabla E_{\alpha^n} = [\frac{\partial E}{\partial \alpha_1^n}, \frac{\partial E}{\partial \alpha_2^n}, \ldots, \frac{\partial E}{\partial \alpha_N^n}]^T$.

The computational complexity of the algorithm depends on the number of balls N and the number of points L on a segment where the points and derivatives are evaluated. For each iteration of the gradient descent procedure, the computational complexity is $O(NL)$. The number of iterations required depends on the time step Δt as well as how close the initial skin is to the final solution.

4 Results

A simple example is provided in Figure 3. Here, four balls of radius 50, 75, 50, and 25 pixels, respectively were set along the x-axis. The initial angles for this experiment were 0.57, 1.07, 1.57 and 2.07 radians, respectively; the initial skin is shown in part (a) of the figure. The angles were iteratively updated using Equation 25. An intermediate solution after 50 iterations in shown in (b), at this stage, the skin is considerably smoother while still satisfying the constraints of the problem. We show the result after 100 iterations in (c), at which point the energy has reached a minimum and the angles have converged. The solution (all 100 iterations) is computed in 47 milliseconds using C++ code compiled on a machine with a 3.0 GHz single-core processor.

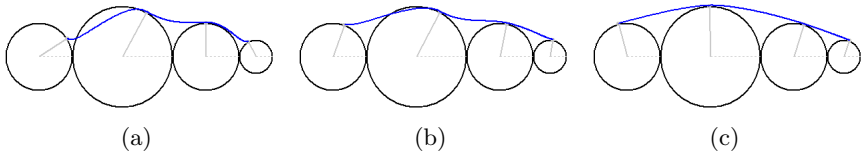


Fig. 3. Example ball skinning. The initialization is shown in (a), and the result after 50 iterations is shown in (b), and the converged result after 100 iterations is shown in (c). The skin is rendered in a blue color.

Figure 4 shows a slightly more complicated example for which some balls overlap and others do not. The initial skin is shown in (a), an intermediate result after 70 iterations in (b), and the final result upon convergence after 140 iterations in (c). The solution (all 140 iterations) is computed in 143 milliseconds.

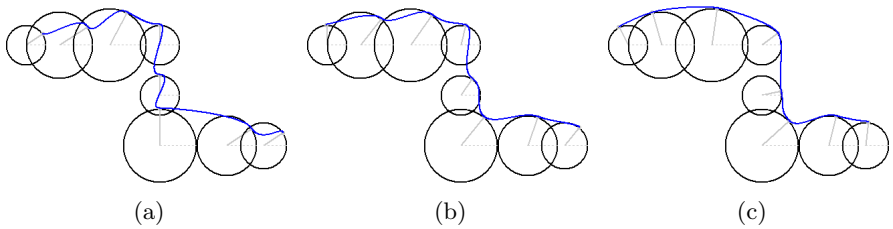
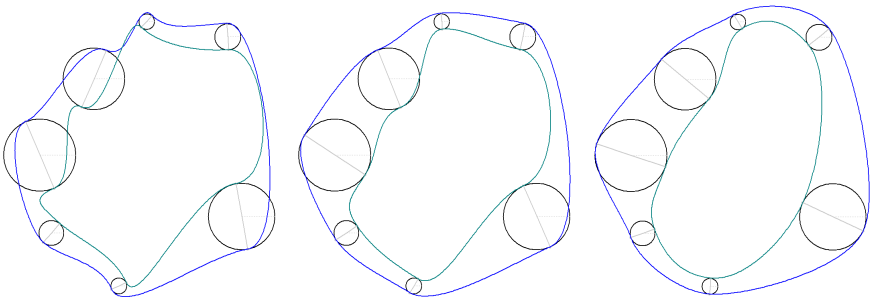


Fig. 4. Another skinning of a set of balls. Initialization (a), intermediate result (b) after 70 iterations, and final result upon convergence (c) after 140 iterations.

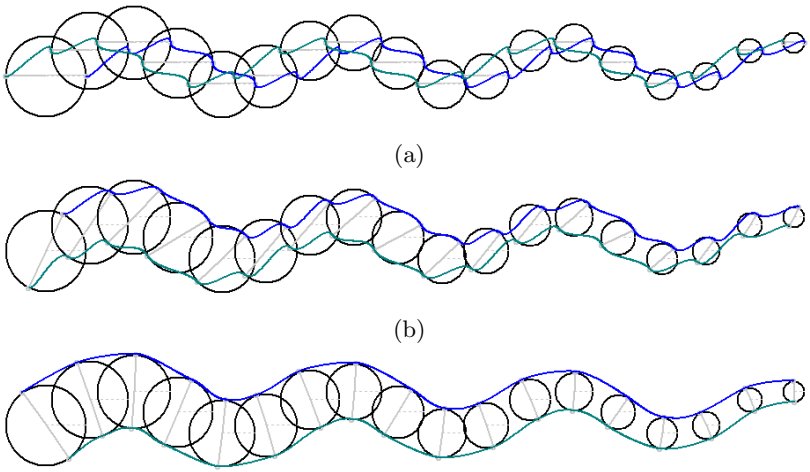
In Figure 5 we show an example of generating an interpolating region for a collection of balls. In this case, we have two skins, one defining the interior boundary of the region (rendered in green), and another defining the exterior boundary (rendered in blue). For each ball, there are two points of contact: one from the interior skin and one for the exterior skin; however, we constrain these points of contact to be separated by 180 degrees. Therefore, for each ball there is only one angle α_i to be determined as in the examples above. We solve for the angle for all the balls, with each skin contributing a term in Equation 25. In part (a) of Figure 5 we show an initialization, in (b) and intermediate result after 50 iterations, and in (c) the final converged result after 100 iterations. Convergence for this example occurs in 190 milliseconds.

More examples are provided in Figures 6 and 1. In Figure 6, the balls are arranged on a sine wave and have a variable radius. Convergence of the skinning algorithm, starting from a set of angles far from the optimal result, takes 775 milliseconds. In Figure 1, the variable radius balls are arranged in a spiral. The skin is generated in 2.5 seconds.



(a) (b) (c)

Fig. 5. Generating a smooth interpolating region between a set of balls. Initialization (a), intermediate result (b) after 50 iterations, and final result upon convergence (c) after 100 iterations.



(a)

(b)

(c)

Fig. 6. Generating a smooth interpolating region between a set of balls. Initialization (a), intermediate result (b) after 80 iterations, and final result upon convergence (c) after 160 iterations.

We note that our gradient descent approach only guarantees a locally optimal solution; the particular solution depends on the convexity of the energy functional as well as the initial condition, as demonstrated in Figure 7. In (a) we show an initial skin, and in (b) the result of our approach. The initial condition in (c) is identical to (a) except the middle ball has a different angle drawing the skin down. In (d), we show the result of our approach starting from (c), resulting in a different solution. In this example and many others in the paper, the initial

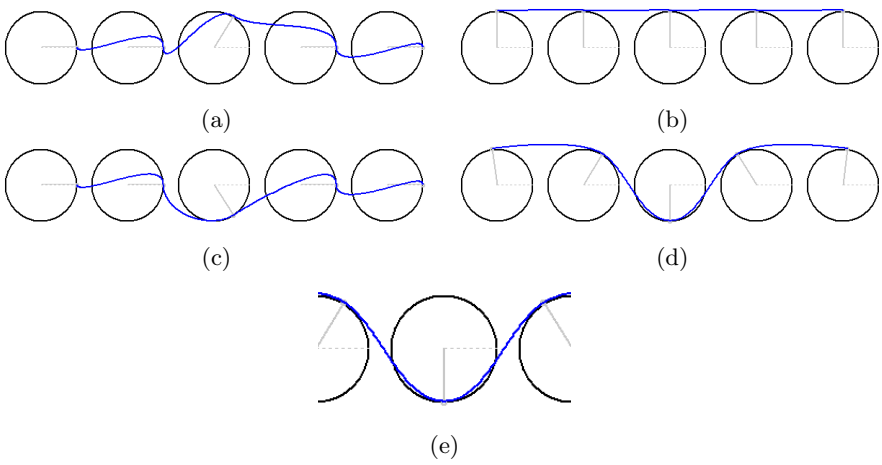


Fig. 7. Solution depends on initial condition. Different solutions (b) and (d) are possible depending on the initial condition (a) and (c), respectively. In (e), we show that the skin may pass through a ball.

skins are chosen to be far from the final solution to demonstrate the effect and robustness of the differential equations. However, in practice, it is typically easy to determine a good initialization based on the choosing an angle for each ball that is along the ray orthogonal to the centerline connecting adjacent ball centroids. Finally, we note that the skin our method generates may pass through a ball (shown in Figure 7 (e)) since it is only constrained to be tangent to the ball at one point of incidence. In our application of modeling blood vessels, this is an acceptable solution since ball itself is a geometric proxy of the local vessel geometry.

5 Conclusion

In this paper, we presented a method for optimally skinning an ordered set of 2D balls. Our formulation of the problem requires that the skin be modeled by a point of contact with each ball and at the point of contact, be tangent to the ball. We have presented novel derivations resulting in differential equations that minimize a convex combination of the arc length and curvature of a third order polynomial spline subject to these constraints. Starting with an initial skin, we evolve the skin's parameters until convergence. Experimental results demonstrate the viability of the method.

For future work, we are interested in extending the approach to interpolate balls in R^3. In this case, the point of contact for a ball will be modeled with two angles, and the derivation will result in a coupled set of partial differential equations for these two angles. The skin will then be a surface that interpolates the balls, constrained by the points of contact.

References

1. K. Singh and E. Kokkevis, "Skinning Characters using Surface Oriented Free-Form Deformations," in *Graphics Interface*, 2000, pp. 35–42.
2. M. Paternell, "Rational Parametrizations for Envelopes of Quadric Families," Ph.D. dissertation, University of Technology, Vienna, Austria, 1997.
3. H. Cheng and X. Shi, "Quality Mesh Generation for Molecular Skin Surfaces Using Restricted Union of Balls," in *IEEE Visualization*, 2005.
4. H. Edelsbrunner, "Deformable smooth surface design," *Discrete and Computational Geometry*, vol. 21, no. 1, pp. 87–115, 1999.
5. N. Kruithov and G. Vegter, "Envelope Surfaces," in *Annual Symposium on Computational Geometry*, 2006, pp. 411 – 420.
6. B. Whited, J. Rossignac, G. Slabaugh, T. Fang, and G. Unal, "Pearling: 3D Interactive Extraction of Tubular Structures from Volumetric Images," in *Interaction in Medical Image Analysis and Visualization, held in conjunction with MICCAI*, 2007.