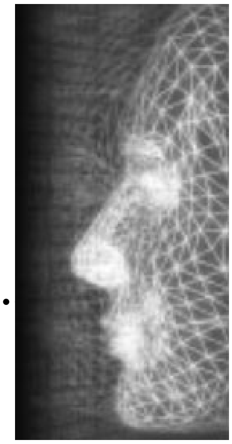
COMPUTER ANIMATION AND VIRTUAL WORLDS
*Comp. Anim. Virtual Worlds* 2005; **16:** 415–427
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/cav.88

**Natural Phenomena and Special Effects**



# Interactive venation-based leaf shape modeling

*By Sung Min Hong*, Bruce Simpson and Gladimir V. G. Baranoski*

***We describe a representation for tree leaves and an interactive modeling system for creating realistic close-up images of leaf clusters. The planar outline of the leaf and the larger members of its venation system are strong factors in the recognition of plant species and as such are essential to realistic imaging. The larger veins also play a major biological role in determining the leaf surface shape and it is this role that we mimic in the shape modeling discussed in this paper. The proposed representation uses a model of a leaf consisting of a three-dimensional skeleton formed by its larger veins and a surface membrane representing the leaf lamina that spans the void between the veins. The veins play two roles. They can be interactively modified to create the 3-D shape of the leaf model. They also provide for realistic light and shadow effects when rendered as generalized cylinders using measured width parameters. The representation consists of two coupled data structures, a tree data structure of veins for the leaf skeleton and an unstructured triangular mesh for the leaf membrane. The skeleton is modified by the user of the modeling system, and the membrane mesh is a surface mesh that follows the skeleton shape computed using harmonic interpolation. Copyright © 2005 John Wiley & Sons, Ltd.***

KEY WORDS: venation; leaf; skeleton; interpolation

## Introduction

The planar outline of a plant leaf and the larger members of its venation system are strong factors in the recognition of tree and plant species and as such are essential to realistic imaging. The larger veins also play a major biological role in determining the leaf surface shape and it is this role that we mimic in the shape modeling discussed in this paper. The 3-D surface shapes of leaves create light and shadow effects that contribute to their visual appeal, as well as providing indications of their age and health as suggested by Figure 1.

In computer graphics applications, veins are usually represented using either texture mapping or bump mapping. These techniques, however, do not account for the physical presence of the veins, which protrude from the leaf surface, adding to the thickness of the foliar tissues at the venation sites. Consequently, neither of these techniques can realistically represent the silhouette view of a plant leaf and, more importantly, the masking and shadowing of light incident on the leaf surface (see Figure 13).

In this paper, we describe a representation for detailed descriptions of plant leaves which account for the physical presence of veins, and an interactive modeling system for creating close up images of leaf clusters. This interactive system provides a practical framework for the simulation of foliar shape changes associated with environmental factors and physiological processes. Furthermore, by incorporating the 3-D modeling of the veins, the proposed representation allows the realistic simulation of the illumination effects mentioned above. It is important to note, however, that the main focus of our research is the geometrical modeling of plant leaves, i.e., rendering issues, such as light transport and absorption within foliar tissues, are beyond the scope of this

*Correspondence to: S. M. Hong, Scientific Computing Laboratory, University of Waterloo, 200 University Avenue West, Waterloo, Ont., Canada N2L 3G1.
E-mail: smhong@cs.uwaterloo.ca

*Figure 1. Photos of maple leaves.*

work. For a comprehensive literature review in this area, the reader is referred to recent texts by Jacquemoud and Ustin[1] and Baranoski and Rokne.[2]

The proposed representation uses a model of a leaf consisting of a 3-D skeleton formed by its larger veins and a surface membrane representing the leaf lamina that spans the void between the veins. The veins are represented by piecewise linear space curves and the skeleton is a hierarchical collection of these curves that follows the biological classification reviewed in Section 'Related Leaf Anatomy'. The interactive modeling system builds two coupled data structures for this representation. One is a tree data structure for the veins of the skeleton; the other is an unstructured triangular surface mesh for the leaf membrane. It starts with scanned images of a leaf, builds and models the skeleton interactively and generates the membrane mesh automatically. An overview of these processes is given below. The system outputs a 3-D leaf model suitable for inclusion in a scene for rendering, such as presented in Figure 10.

## Overview of the Modeling Processes

The interactive modeling system that we describe tries to strike a pragmatic balance between processes that can be automated and those that seem to require interaction to achieve the desired level of realism. It has primarily been designed to support our experimentation with interactive leaf shape modeling. Using Figure 2, we give an overview of how the modeler builds the data structures for a 3-D image of a leaf. These schematics label the successive states of the data structures in lower case Roman numerals, (i), (ii), etc. and label the transition processes in upper case letters, A, B, etc. The states of the data structures during this sequence of processes are shown in Figure 3.

The coloration of real leaves on the side that is normally exposed to the sun is usually different to their shadow side. So the modeling system takes as input a pair of scanned color images of the sunny and shadow sides of a leaf. From these, it extracts, in process A, a polygonal boundary automatically. This polygon is regarded as lying in the $z = 0$ plane of the leaf model coordinate system. The user then, in process B, interactively identifies suitable structural veins for the model from the shadow side scan, as a branching hierarchy of piecewise linear curves. The base width of the selected veins is also recorded in process B to provide parameters for vein rendering.

Process C automatically constructs a quality triangular mesh for the polygonal profile of the leaf that includes the veins as constrained edges. The user can then create a 3-D shape for the leaf skeleton by interactively bending the veins in the model coordinate space, which is process D. This shape is automatically extended to the triangular mesh that represents the leaf membrane by harmonic interpolation in process E. The rendering of an image of the 3-D shaped leaf into a scene is process F. The leaf lamina can be rendered by transforming each triangle to its position in the scene space and using the texture mapping appropriate to the visible side of the triangle. Similarly, the veins can be rendered as generalized cylinders, using prorated values of the vein base width parameter. These structural veins and the curvatures of the membrane allow for realistic light and shadow effects in scenes when global illumination methods, such as ray tracing or path tracing are used.

Process D can be repeated several times starting with the data structures in state (iv) to produces several leaves for a cluster that have the same profile, but different 3-D shapes. We illustrate this in Figures 12 and 10.

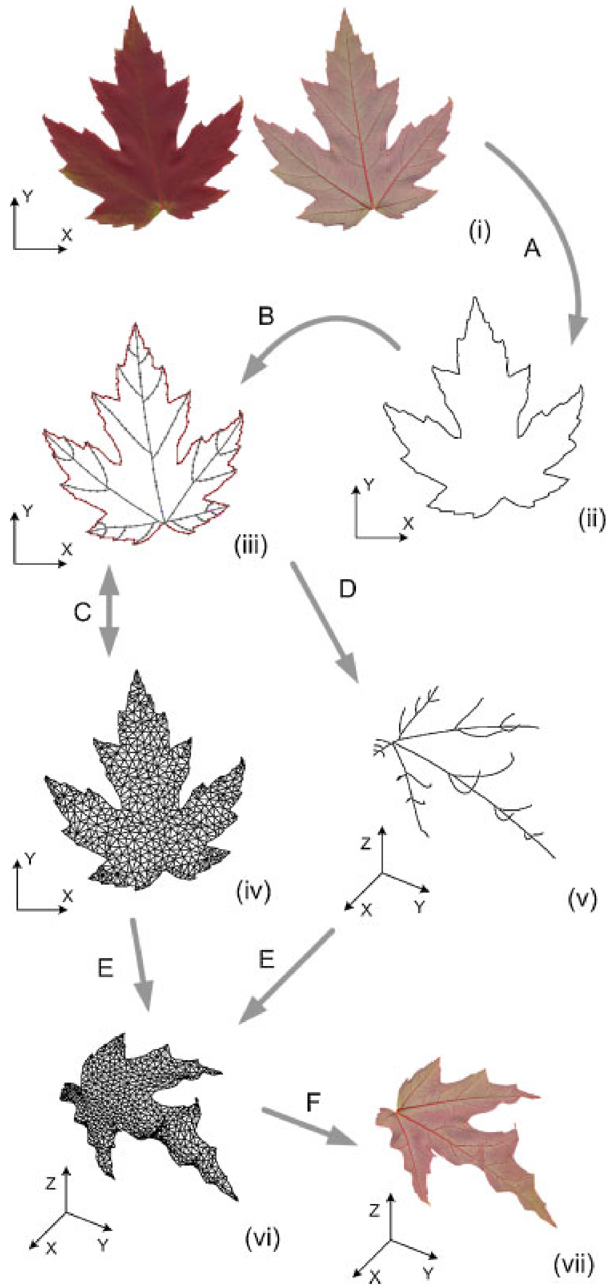



*Figure 2. Modeling processes.*

## Review of Related Research

In a recent paper, Mündermann *et al.*[3] present a discussion of leaf shape modeling based on a series of processes for building a representation of a leaf skeleton and leaf membranes. The choice of skeleton and membrane are significantly different from the ones presented in this paper, and the processes used are correspondingly different. The paper discusses the use of the medial axis of the leaf boundary as the basis for the leaf skeleton. The skeleton for modeling is created as a hierarchy of spline curves with curves added geometrically that extend from the medial axis to the leaf boundary. The resulting skeleton divides the leaf profile into polygonal patches. The skeleton and the leaf boundary are then interactively modeled into a 3-D

| State | Leaf membrane | Leaf skeleton |
|---|---|---|
| (i) | Scanned leaf images In (x, y) plane. | None. |
| (ii) | Leaf profile. | None. |
| (iii) | Veins added as internal edges of polygonal profile. The resulting data represented is a planar straight line graph (PLSG). | Hierarchical list of the same vein line segments in the (x, y) plane. |
| (iv) | A good quality planar constrained Delaunay triangulation that is a refinement of the PSLG of (iii). | The skeleton is updated to include any refinement of vein edges in the mesh. |
| (v) | Unchanged. | Coordinates of ends of line segments are modified. |
| (vi) | All mesh nodes mapped into 3-D shape consistent with leaf skeleton of (v). | Unchanged. |

*Figure 3. States of the data structures for leaf modeling.*

shape and the leaf membrane is created by interpolation of these patches by sweeping using generating curves over the patches.

For polygonal objects generally, the medial axis is a geometric construct that is sometimes referred to as the object's skeleton. In Reference [4], Gold presented a method that computes from a scanned image, both a polygonal boundary and its medial axis. He demonstrated this method on a number of shapes including an artificial maple leaf.

The use of the medial axis in forming a leaf skeleton provides an interesting instance of the trade-off between automation and interaction for leaf modeling. It leads to highly automatable approaches to building a skeleton. Whether such approaches can be developed that provide a high degree of realism is an open question. We are not aware of other research that aims to build models of leaf skeletons that are anatomically based.

The harmonic interpolation technique of process E is different from the process of Reference [3] for generating the leaf surface membrane. Both techniques are automated; however, harmonic interpolation is more general in that it does not require modeling of the leaf boundary, nor does it require that all the veins extend to the boundary.

There is an extensive literature on realistic plant modeling and scene creation involving plants, which are activities to which our leaf models can be directed. Prusinkiewicz *et al.*[5] provide a current, detailed presentation of combining interaction and parameterized algorithms for both of these topics. In particular, it contains a discussion of the use of generalized cylinders which is pertinent to our view of rendering leaf veins. The comprehensive bibliography of this paper references many earlier contributions to plant modeling by the authors, and others. Lintermann and Deussen,[6] have presented a general approach to organizing the wide range of features in plant imaging, including leaves as one component, using parameterized interaction. Bloomenthal,[7] used templates for leaves in his discussion of modeling entire maple trees.

There are many papers that relate to steps in the process described by the schematics in Figure 2. A number of these steps require techniques that deserve reference, but which are not directly connected to the focus of this paper. We reference them in the discussion of the step to which they apply.

## Related Leaf Anatomy

Our primary references for the biology of venation systems are the review article by Roth-Nebelsick *et al.*[8] the comprehensive reference by Hickey,[9] and the article by Bohn *et al.*[10] In this section, we review features that are relevant to the skeleton of the leaf model and decisions about its design.

Two primary functions of a vein of a leaf are to provide mechanical stability to the leaf as a structure and to transport water and nutrients between the leaf and the other organs of the plant. For familiar tree leaves, the veins appear organized into a hierarchical network distributed over the leaf with a distinctive pattern characteristic of the plant. The distinctive venation systems of some common leaves can be seen in the photos of Figures 1, 4, and 13. We detail some obvious features. Veins have a direction indicated by decreasing width; we refer to the wider end as the base. In planar images, the larger veins appear as bands whose center line is a curve with a well-defined tangent that usually changes little in direction. Leaf images show that veins can be classified into distinct groups based on the highly correlated attributes of base width and length.

These groups form the basis for classifying the vein hierarchy. The veins in the group with the largest attributes extend from their base where the leaf boundary joins its stem, usually diametrically across the leaf to its periphery. These are the primary veins at the lowest level of the hierarchy. The oak leaf of Figure 4 has one primary vein; maple leaves typically have five. The veins grouped in the next biggest size category are the secondary veins; they branch from the primaries

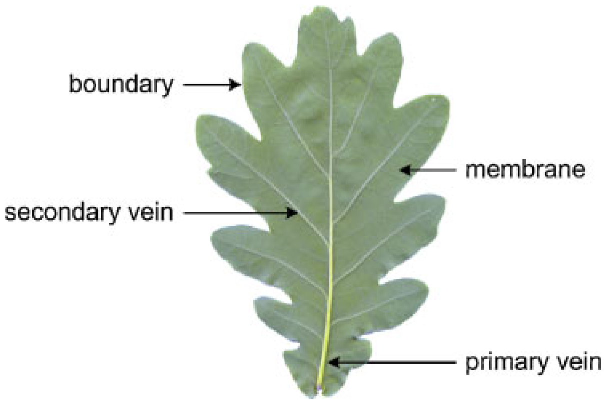



*Figure 4. Photo showing leaf components.*

and extend well out towards the boundary of the leaf. This classification can be extended to six to eight levels for the types of leaves that we are discussing; although the distinctions become increasingly fine.

The network topology of real leaves is influenced by the dual biological functions of mechanical stability and fluid transport of the system. All levels contribute to fluid transport, but it is the role of the lower levels to also provide mechanical stability. Although a leaf may have multiple primary veins, the network virtually never has a cycle of primary veins. Cycles in the subnetwork of primary and secondary veins are unusual but not rare. Cycles involving subnetworks of veins of levels higher than 2 are the norm. It seems clear that the redundancy which cycles bring to the network serves the fluid transport function.

## Process A: Extracting a Leaf Profile

We use the term 'profile' for the 2-D outline of a leaf to distinguish it from the 3-D surface form of a leaf, for which we use the term 'shape.' This profile is regarded as a domain of the plane $z = 0$ of the 3-D space of the leaf model. In process A of Figure 2, a polygonal boundary for the profile is recovered from the scanned image of a leaf by common techniques involving edge detection and boundary simplification. For conciseness, we do not detail these techniques; however, we note that the amount of edge detail in the initial polygonal boundary plays a significant role in the size of the leaf membrane mesh in state (iv). Our boundary recovery techniques are parameterized to provide more or less detail in order to allow some control over the resulting mesh size.

## Processes B and C: Building the Leaf Skeleton and Mesh

Processes B and C have the coupled tasks of building the leaf skeleton and unstructured mesh. The automatically recovered leaf profile (process A) and the appropriateness of the interactively selected initial skeleton (process B) form the biological input to the leaf model. As shown in the middle of Figure 2, the generation of the mesh for the profile (process C) plays two roles. It refines the skeleton and provides the discretization for the harmonic interpolation of process E that creates the leaf membrane.

We discuss the leaf skeleton data structure in detail in the following subsection, and comment on its relation to the biological vein hierarchy. Some details of processes B and C then follow.

### Defining the Skeleton Data Structure

Because the leaf skeleton is the key control for modeling the 3-D shape, we describe its data structure in detail. The leaf skeleton of the modeler is based on a selection of the lower order veins that can be used:

(a) to define the 3-D surface shape of the leaf image,
(b) to provide a 3-D structure for the image of the vein for rendering the leaf.

In principle, different skeletons could be used for these two functions. However, we believe that a biologically appropriate skeleton can serve both adequately.

A vein is represented as a sequence of directed line segments which form a piecewise linear space curve and which we will call a vein sequence. We describe the ends of a line segment as its start and final points and similarly for vein sequences. We note that Process C of Figure 2 refines vein sequences, so a dynamic data structure for the sequence is needed.

We can describe the data structure for the leaf skeleton in graph terms as a shallow tree of vein sequences. It appears to us that a three level hierarchy of vein sequences suffices for purposes (a) and (b). The parent–child relationship of the vein sequences is defined by start points (see Figure 4). A vein sequence that has no ancestor is a primary vein sequence of the skeleton. If the start point of a sequence, *VS*, coincides with the final point of a line segment, *ls*, of a primary vein sequence, *PS*, then *VS* is a secondary vein sequence, *PS* is its parent vein and *ls* is its parent line segment. Tertiary vein sequences have a similar relationship to secondary vein sequences.

This parent–child relationship ensures a unique parent, as long as the two veins of the lower level do not end at the same final point. This would signal a cycle in the veins of the parent level. We impose on the leaf model the condition that no cycles are permitted in the vein hierarchy. In section 'Related Leaf Anatomy', we pointed out that cycles involving primary veins virtually never occur and involving secondary veins are rare. We argue that the algorithmic simplification benefits of this acyclic assumption are significant and the reduction in realism of the resulting image is insignificant. We note that secondary and tertiary vein sequences are permitted to end at a point interior to the leaf boundary, and that tertiary veins can form cycles. These two flexibilities, which are biologically appropriate, may serve purpose (b). We have not experimented with them for purpose (a).

Since the line segments are totally ordered within each vein sequence, the vein tree structure induces a tree structure on the set of line segments. We use this tree structure on the line segments in our description of the vein modeling algorithm in *Process D* below.

## Some Details of the Processes

In Process B, an initial coarse leaf skeleton is created, which is then refined as a byproduct of the mesh generation of Process C. The endpoints for the line segments chosen for the vein sequence for a selected vein are identified by mouse clicks. Then the vein base width is measured and recorded. To create the vein hierarchy, new veins must be identified after their parent line segment has been identified.

The line segments of the skeleton that are created in process B are added as additional vertices and edges to the polygonal boundary of the leaf. The combination is a planar straight line graph (PSLG) of edges and vertices from the boundary plus initial skeleton. Quality mesh generation for a PSLG is a standard computation involving two control parameters, a triangle size tolerance ($\zeta$) and a minimum angle tolerance ($\eta$). A quality mesh generation program computes a constrained Delaunay triangulation that is a refinement of the input PSLG, with no triangle exceeding $\zeta$ in size and no angle less than $\eta$, except possibly for small angles between constrained edges in the input PSLG. See Ruppert[11] or George and Borouchaki[12] for descriptions of this computation. The modeler uses the freely distributed program Triangle by Shewchuk,[13] for this task. The choice of the tolerance parameters influences the size of the mesh generated. The mesh vertices created by the refinement process used in quality mesh generation include some additional points inserted on constrained edges from the leaf skeleton. The vein sequences of the leaf skeleton must be updated to replace line segments that have been refined during the mesh generation phase of Process C. As a result, a discretization that is suitably fine for modeling the veins is obtained.

## Process D: Vein Shape Modeling

The planar veins of Process B are interactively modeled into 3-D space curves in Process D. An example of vein shape modeling for a chestnut leaf is shown in Figure 5. The vein sequences of the skeleton appear as 3-D space curves. In this case, we are modeling an older leaf with a highly curled skeleton.

Each vein is modeled independently by rotation of selected line segments of the vein. The basic concept is that the user selects an active line segment for modification in the leaf skeleton and inputs rotations about its start point. Each rotation is extended to the subtree of line segments rooted at the active line segment. A subtree rotation in a plane is illustrated in Figure 6; it is a common modification of hierarchical graphic structures.

After process C has refined them, the vein sequences are long enough to warrant introducing a technique to facilitate this basic idea. This simplification technique operates at the level of modifying the shape of one vein sequence, *VS*. A temporary control vein sequence, *CVS*, that approximates *VS*, but has fewer line segments, is identified by selecting a subset of the end points of *VS*, including its start and final points. *CVS* is then reshaped

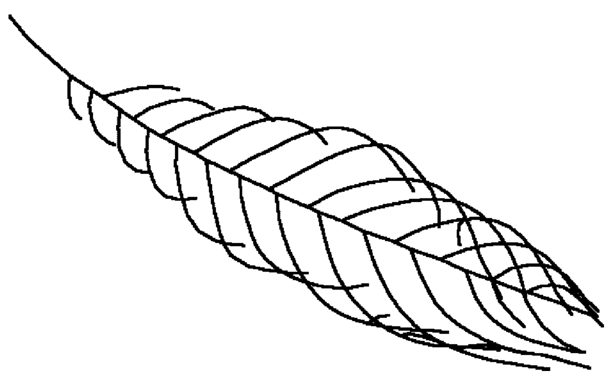


*Figure 5. Shape modeled skeleton: chestnut leaf.*

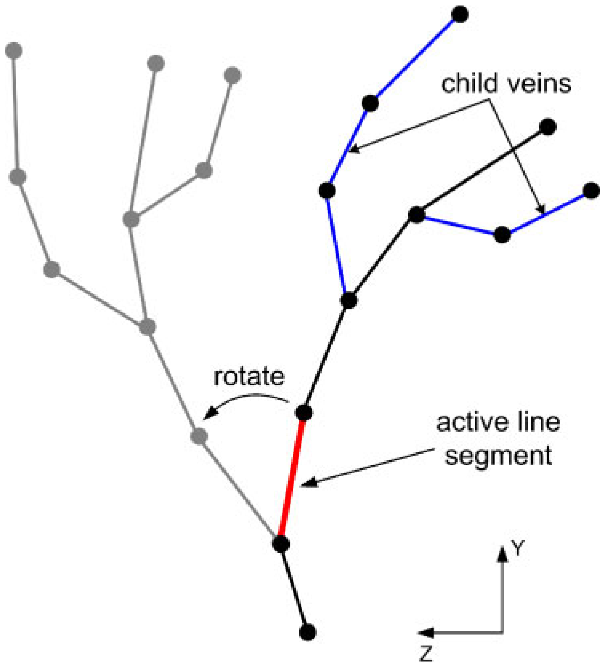



*Figure 6. Rotating subtree of active line segment.*

using a walk through *CVS* applying user selected 3-D rotations to each control vein line segment, and its successors. A cubic spline space curve is fitted to the reshaped *CVS*; this use of a spline curve has been anticipated by several other authors, for example References [3,6]. The basic idea described above is then automated for *VS*. For each line segment of *VS*, rotations about its start point are computed that place its final point on the cubic spline shape curve. These rotations are applied to the leaf skeleton subtree of the line segment. As described in Section 'Modeler Interface' below, the user can see that effect of reshaping *CVS* on the leaf image during the interaction.

## Process E: Building the Leaf Membrane

Harmonic interpolation (Process E) provides a powerful general technique to extend the shape of the modeled skeleton to the leaf membrane using the mesh for the leaf profile generated in Process C. Figure 7 shows the triangular mesh surface of the leaf membrane for the highly curled chestnut leaf skeleton of Figure 5. This mesh is somewhat coarser that we usually use for modeling in order to make the figure less cluttered.

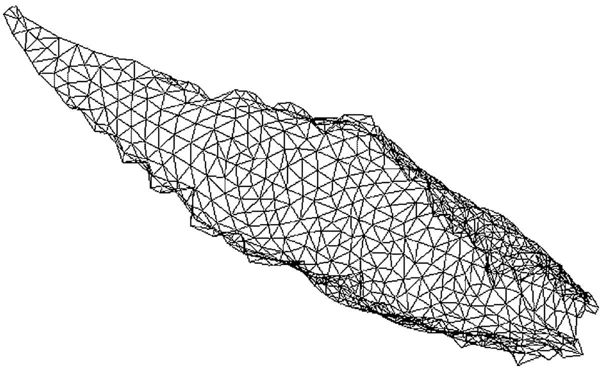


*Figure 7. Harmonic interpolated leaf membrane mesh: chestnut leaf.*

Typically, meshes of the order of about a thousandtriangles are sufficient for shape modeling. Harmonic interpolation is implicit in the sense that a system of equations of about the size of the leaf mesh must be solved for each of the three spatial coordinates of the interpolated surface. A simple description of these equations is given at Equation 2. These computations can be done for meshes of at least several thousand triangles with no noticeable delay in updating the displayed leaf shape during interactive modeling.

Harmonic interpolation is motivated by the following classical specification of a function, $f(r,s)$, in a domain $D$ when $f(r,s)$ is required to take known values, $g(r,s)$ on part of the boundary $\partial D_1$ of $D$. $f$ is specified as the solution of the following partial differential equation boundary value problem

$$\begin{aligned} \Delta f = \frac{\partial^2 f}{\partial r^2} + \frac{\partial^2 f}{\partial s^2} = 0 \ \text{ for } \ (r,s) \in D \\ f(r,s) = g(r,s) \ \text{ for } \ (r,s) \in \partial D_1 \\ \frac{\partial f}{\partial n} = 0 \ \text{ for } \ (r,s) \in \partial D - \partial D_1 \end{aligned} \tag{1}$$

Functions that satisfy Laplace's equation, $\Delta u = 0$, are called harmonic functions. The function $f$ specified by Equation (1) can be viewed as a harmonic function defined in $D$ that interpolates $g(r,s)$ on $\partial D_1$.

The extension of this concept to computing the representation of the leaf membrane as a parametric surface can be described mathematically as follows. By relabeling the $x$ and $y$ variables as $r$ and $s$, we can regard the leaf profile in the $(x,y,0)$ plane as the domain, $D$, of parameter space. We seek to define the parametric surface by three coordinate functions $(x(r,s), y(r,s), z(r,s))$ that are defined for $(r,s)$ in $D$ to form the leaf membrane. The planar leaf skeleton of state (iv) of Figure 3 constitutes an internal 'constraint' on which $(x_{sk}(r,s), y_{sk}(r,s), z_{sk}(r,s))$ are specified by the leaf shape modeling of process D. We specify that $x(r,s)$ be harmonic in the leaf profile, except at the points of the skeleton where $x(r,s) = x_{sk}(r,s)$, and that $\partial x/\partial n = 0$ on the leaf boundary, except at a point of the skeleton that extends to the boundary, and similarly for $y(r,s)$ and $z(r,s)$. In this sense, the model for the leaf membrane $(x(r,s), y(r,s), z(r,s))$, is the harmonic interpolating surface of the modeled leaf skeleton.

Formally, we compute the $x$ coordinates at the vertices of the leaf membrane mesh, now parameterized by $(r,s)$, by solving Laplace's equation, $\Delta x = 0$, using the finite element method (FEM) applied using the mesh and subject to the skeleton and leaf boundary conditions. Repeating this for $y$ and $z$, we get the coordinates $(x_k, y_k, z_k)$ for each mesh vertex, $P_k$. However, we do not need much of the apparatus of the FEM, because in this simple case, the linear equations for the coordinates, $x_k$, $y_k$, and $z_k$, can be formed from a single parameter, $\gamma_{i,j}$, associated with the mesh edge between vertices $P_i$ and $P_j$. Because we are using a Delaunay triangulation, $\gamma_{i,j}$ has a simple geometric description. For an edge internal to the mesh, let $T_A$ and $T_B$ be the mesh triangles that share the common edge between $P_i$ and $P_j$ as shown in Figure 8. Let $Q_A$ and $Q_B$ be the circumcenters of $T_A$ and $T_B$, then $\gamma_{i,j} = \|Q_A - Q_B\|/\|P_i - P_j\|$; see Letniowski.[14]

If edge $P_i$ to $P_j$ is on the boundary of the leave, then $T_A$ has no neighbor $T_B$. In this case, $Q_B$ is the midpoint of edge $P_i$ to $P_j$ and $T_A$ must be acute so that $Q_A$ stays in $T_A$. This restriction on the triangles at the leaf boundary basically requires the mesh to be adequately fine and well shaped near the boundary.

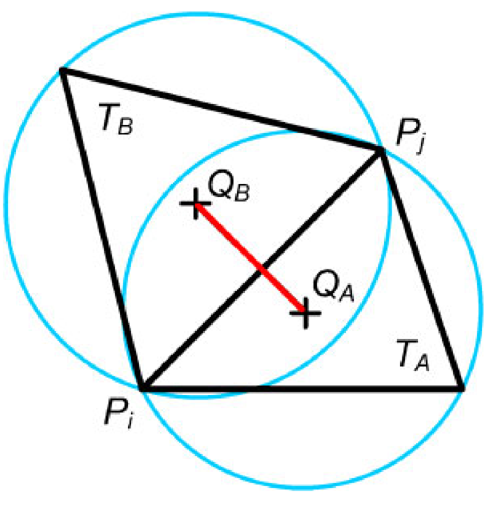



*Figure 8. Neighboring triangles of a Delaunay edge.*

To compute the discretized coordinate function, $x(r,s)$, the FEM method specifies a system of linear equations of the form $Ax = b$ where $x_k$ is the value of the coordinate function at mesh vertex $P_k$. In Equation (2), we give the form of the equation associated with a mesh vertex, $P_i$, not on the leaf skeleton, that is connect to $m$ neighboring mesh vertices, $P_{j_k}$, $k=1$ to $m$ by mesh edges.

$$\left(\sum_{k=1}^{m} \gamma_{i,j_k}\right) x_i - \sum_{k=1}^{m} \gamma_{i,j_k} x_{j_k} = 0 \qquad (2)$$

If $P_i = (r_i, s_i)$ lies on the leaf skeleton, then the equation is simply $x_i = x_{sk}(r_i, s_i)$. This is a system of linear equations with sparse matrix, $A$, which can be solved by most sparse matrix techniques or software. A system of equations with the same matrix, but different right hand side vectors is also specified for the discretized coordinate functions $y(r,s)$ and $z(r,s)$. Efficiencies in computing the harmonic interpolating mesh can be gained by observing that these three systems of equations all have the same matrix of coefficients.

## The Interface of the Leaf Shape Modeler

Figure 9 displays the modeler's GUI; it is created using the GUI toolkit Qt.[15] It comprises a viewing window, on the left, a small control panel for the viewing window, on the upper right, and a panel with menu tabs for controlling the processes of the schematics in Figure 2 on the lower right.

The menu tabs correspond to the processes as follows:

**Profile** Process A—image processing and a boundary edge editor
**Vein** Process B—a vein sequence editor
**Mesh** Process C—mesh generation
**Model** Process D and E—vein modeling and harmonic interpolation
**Render** a simple version of Process F—control of OpenGL rendering of the modeled leaf for display by the modeler

The GUI also provides for importing a scanned image of a leaf, or a previously saved representation, and the export of a representation for future modeling, or scene creation. The viewing window can be used to display a single view for the processes of a menu, or subdivided into four subwindows, as shown in Figure 9. The four subwindow viewing state is used for leaf modeling; it is

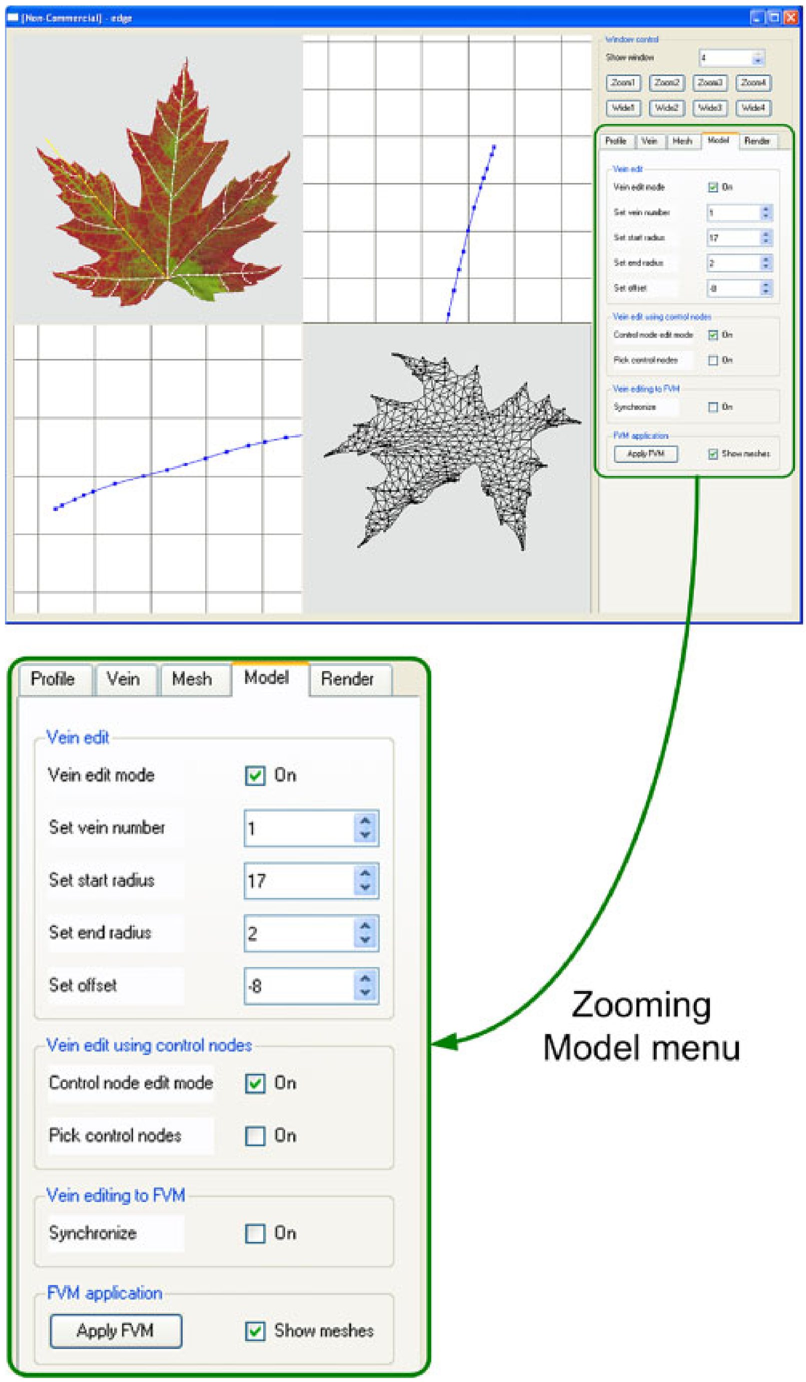



*Figure 9. Modeler interface:* **Model** *menu.*

a common configuration for 3-D modeling. The bottom right subwindow displays the 3-D image of the current leaf representation as either a skeleton, a mesh, or a rendered image, in an orientation controlled by the viewing window control panel. The other three windows support the interactive modeling under the control of the **Model** menu. The upper left window shows a front view of the entire leaf skeleton. The user can use it to select a particular vein sequence to modify, which we refer to as the active vein sequence, *AVS*. The *AVS* alone

*Figure 10. Cluster of maple leaves scene.*

*Figure 11. Three dried leaves.*

is shown in the two other windows, in side and bottom views. Vein modeling is normally done by manipulating the *AVS* in the windows displaying its planar form.

For selecting the *AVS* for modeling, the user may either select a vein sequence of the leaf skeleton directly, or interactively create a temporary control vein sequence, *CVS*, as described in Process D to be used as the *AVS*. In either case, the program logic of rotating a line segment, *ls*, of the *AVS*, and rotating the subtree of line segments rooted at *ls* is the same. In the case of a

*CVS*, the subtree is just the successor line segments in the control vein sequence.

The effects of modifications to the *CVS* are computed for the full leaf skeleton, using projection onto a cubic spline curve as discussed in Process D, and the result is displayed in all the four subwindows.

## Demonstration Scenes

In the two subsections below, we present two demonstrations of specific effects in the leaf modeling that we have described. A demonstration of a cluster of maple leaves is given in Figure 10. The leaves in the cluster were modeled from a single profile. Figure 11 shows three dried leaves, which are an oak leaf, a chestnut leaflet, and a silver maple leaf hanging on a cloth covered board. The profiles of the three leaves were obtained from their scanned image using the proposed method in this paper. The rendering of these images was done using POV-Ray,[16] which is a copyrighted freeware ray-tracer.

### Simulating Leaf Aging

Six models of a leaflet of a composite chestnut leaf are shown in Figure 12. They have been modeled from a single profile and colored to create the appearance of different stages of aging. The intention of Figure 12 is that they look progressively older in clockwise order.

### Light and Shadow Effects

Figure 13 shows a mid-day photograph of a cluster of spring chestnut leaves. Prominent on the lower left is the shadow side of a highly curled leaf clearly showing the secondary veins picking up the direct sunlight and casting shadows on the leaf membrane. In Figure 14, the shadow side of a leaf model of a similarly curled chestnut leaf is shown under three lighting conditions. The top view shows simulated morning light at a low elevation on the left, and the bottom view shows evening light coming from the right. The figure demonstrates the role of the modeled veins in creating effects similar to those of Figure 13.

### Conclusions and Future Work

We have described a leaf model that has the detail necessary for rendering close-up images. In a sense, it represents the fine detail end of multiscale resolution of leaves. For integration with a plant or tree modeling system, it would be necessary, or at least highly

*Figure 12. Leaflets simulating aging sequence (clockwise order).*

*Figure 13. Photo showing self-shadowing of leaf in highlighted box.*

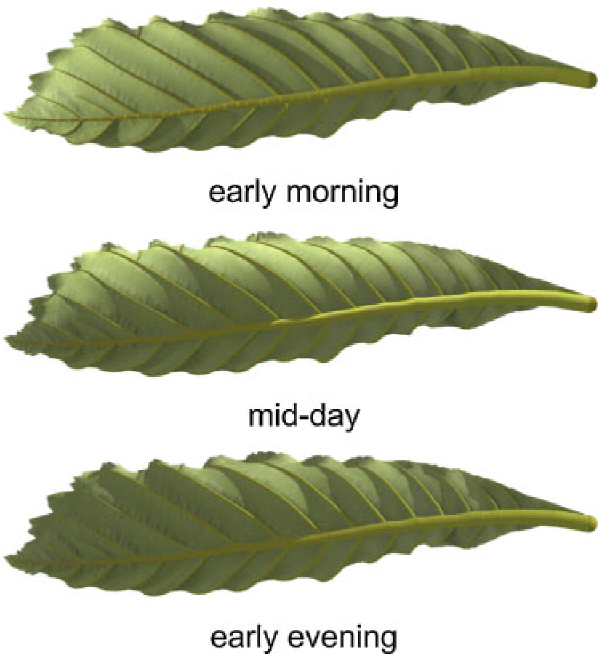



*Figure 14. Modeled light and self-shadowing effects.*

desirable, to develop such a multiscale capability for this modeling approach.

The time consuming activities in our vein modeling processes are the interactive processes of creating the initial skeleton, Process B, and manipulating the skeleton to model the leaf shape, Process D. The automated processes A, C, E, and F cause no noticeable delay to the interaction. Improved image processing techniques, for example, such as presented in Reference [4], may help identify the larger veins of the leaf image and their width parameters and thus partially automate Process B.

A compelling future investigation that this work suggests is research in rendering of plant appearance using biophysically-based models of light interaction with foliar tissue.[2] For example, the shape changes of aging leaves are accompanied by characteristic colors caused by variations in the concentration and distribution of natural pigments (e.g., chlorophylls carotenoids and anthocyanins).[17] The accurate simulation of these biological factors would allow the generation of predictable images of metachromatic (or 'senescent') leaves.

## References

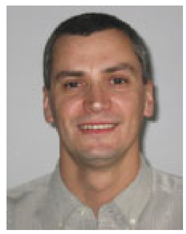
1. Jacquemoud S, Ustin SL. Leaf optical properties: a state of the art. In *8th International Symposium of Physical Measurements and Signatures in Remote Sensing*, Aussois, France, CNES, 2001; pp. 223–332.
2. Baranoski GVG, Rokne JG. *Light Interaction with Plants: A Computer Graphics Perspective*. Horwood Publishing: Chichester, UK, 2004.
3. Mündermann L, MacMurchy P, Pivovarov J, Prusinkiewicz P. Modeling lobed leaves. In *Proceedings of Computer Graphics International*, CGI (ed.), Tokyo, Japan, 2003; pp. 60–65.
4. Gold C. Crust and anti-crust: A one-step boundary and skeleton extraction algorithm. In *Proceedings of 15th Annual Symposium on Computational Geometry*, SIGACT (ed.). ACM Press: Philadelphia, PA, USA, 1999; pp. 189–196.
5. Prusinkiewicz P, Mündermann L, Karwowski R, Lane B. The use of positional information in the modeling of plants. In *Proceedings of SIGGRAPH*, ACM Press: Miami Beach, FL, USA, 2001; pp. 289–300.
6. Lintermann B, Deussen O. Interactive modeling of plants. *IEEE Computer Graphics and Applications* 1999; **19**(1): 56–65.

7. Bloomenthal J. Modeling the mighty maple. In *Proceedings of SIGGRAPH*, ACM (ed.). Philadelphia, PA, 1985, pp. 305–311.
8. Roth-Nebelsick A, Uhl D, Mosbrugger V, Kerp H. Evolution and function of leaf venation architecture: a review. *Annals of Botany* 2001; **87**: 553–566.
9. Hickey LJ. *Anatomy of the Dicotyledons* (2nd edn), Vol. 1. Clarendon Press: Oxford, 1979.
10. Bohn S, Andreotti B, Douady S, Munzinger J, Crowder Y. Constitutive property of the local organization of leaf venation networks. *Physical Review E* 2002; **65**: 061914.
11. Ruppert J. A Delaunay refinement algorithm for quality 2-dimensional mesh generation. *Journal of Algorithms* 1995; **18**: 548–585.
12. George PL, Borouchaki H. *Delaunay Triangulation and Meshing*. Hermes: Paris, France, 1998.
13. Shewchuk JR. Triangle: Engineering a 2D quality mesh generator and Delaunay triangulator. In *First Workshop on Applied Computational Geometry*, ACM (ed.). Philadelphia, PA, 1996; pp. 124–133.
14. Letniowski FW. Three-dimensional Delaunay triangulations for finite element approximations to a second-order diffusion operator. *SIAM Journal of Science and Statistical Computation* 1992; **13**: 765–772.
15. Blanchette J, Summerfield M. *C++ GUI Programming With Qt 3*. (Bruce Perens' Open Source Series). Prentice Hall: Upper Saddle River, NJ, 2004.
16. Persistence of Vision (TM) Raytracer [computer software], 2004. Persistence of Vision Pty. Ltd. ; http://www.povray.org.
17. Ford BJ. A general theory of execration in higher plants. *Journal of Biological Education* 1986; **20**(4): 251–254.

*Authors' biographies:*

**Sung Min Hong** is a research assistant and member of the Scientific Computation Research Group at University of Waterloo. He received the Bachelor's and Master's degrees in Computer Science from University of Waterloo. His research interests are in scientific computation and computer graphics, especially in surface mesh analysis and modeling, and computer animation.

**Richard Bruce Simpson** is professor emeritus of the School of Computer Science, University of Waterloo. He has degrees in Applied Mathematics and Aeronautical Engineering from the University of Toronto and the University of Maryland. He is a member of the Scientific Computation Research Group at University of Waterloo. Much of his research has been directed towards unstructured triangular meshing theory and applications.



**Gladimir Baranoski** received a Ph.D. in Computer Science from the University of Calgary in 1998. Currently, he is a faculty member at the School of Computer Science and the leader of the Natural Phenomena Simulation Group at the University of Waterloo. The results of his research on biophysically-based rendering have been made available to the graphics community through the publication of articles in journals and conference proceedings. He has also organized tutorial presentations for conferences (CGI 2000, EUROGRAPHICS (2001 and 2002) and SIGGRAPH (2002 and 2003)), and he has recently authored a book on topics related to his research.