EUROGRAPHICS 2006 / E. Gröller and L. Szirmay-Kalos
(Guest Editors)

Volume 25 (2006), Number 3

Seamless Integration of Initial Sketching and Subsequent Detail Editing in Flower Modeling

T. Ijiri[†], S. Owada[‡] and T. Igarashi[†§]
[†]The University of Tokyo, [‡]Sony CSL and [§]PRESTO, JST

Abstract

We present an interactive modeling system for flower composition that supports seamless transformation from an initial sketch to a detailed three-dimensional (3D) model. To begin, the user quickly sketches the overall appearance of the desired model as a collection of two-dimensional (2D) strokes on hierarchical billboards. Then the user iteratively replaces the coarse sketch with a detailed 3D model referring to the initial sketch as a guide. Since a flower model consists of many repetitive components, the system helps the user to reuse 3D components to facilitate the modeling process. The global view of the entire model is always shown in a separate window to visualize how local modifications affect the global appearance. Our system helps the user make appropriate design decisions to keep the model consistent with the initial design, which is difficult in traditional bottom-up plant modeling systems in which the global view only emerges after all of the details are specified.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Methodology and Techniques – Interaction Techniques.

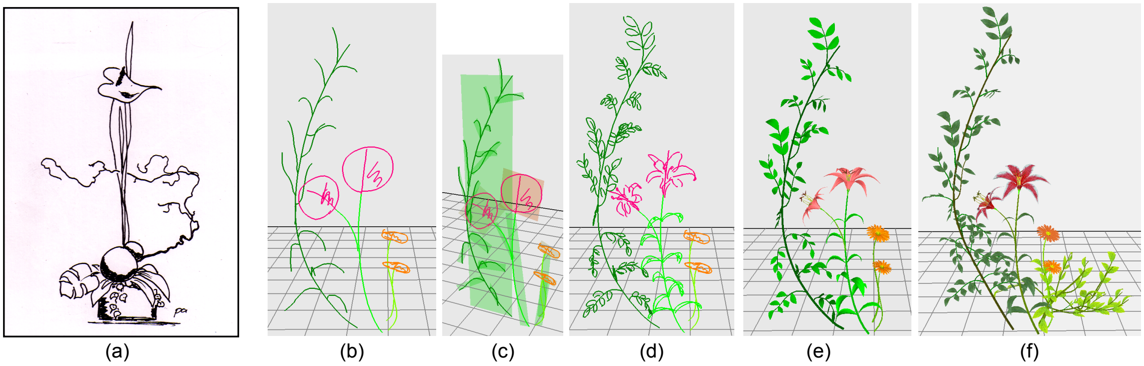

Figure 1: *A concept design sketch used in floral art [Ass96, p. 163] (a) and the modeling process used in our system (b)-(f). The user first designs the overall appearance by drawing sketches (b) on billboards that form a hierarchical structure (c). The user then refines the design by adding and replacing sketches (d). After the initial design is complete, the user models 3D components (e) to replace sketches. Finally, the user refines the model using the copy-and-paste and deformation functions of the 3D components (f).*

1. Introduction

Flowers have been an interesting subject in art because of their beautiful colors and configurations. However, because of their geometrical complexity, it is still difficult to design aesthetically pleasing models of flowers using computer graphics. A single flower consists of both fine detail and hierarchical, multi-scale structures, which equally affect the model's appearance. Furthermore, when designing a scene consisting of multiple flowers, e.g. in flower arrangement, the user must carefully control a number of aspects such as each flower's position, proportion, and pose [Ass96].

The construction of a beautiful three-dimensional (3D) model is not only an engineering activity but also a creative design process. To facilitate this design process, it is important to be able to visualize and examine the overall coarse appearance of the final result during the initial design stage. Such coarse modeling helps the artist to explore the wide range of design alternatives and serves as a reference that guides the subsequent detail modeling. In traditional drawing and painting, artists first sketch the outline of the subject and use it as a reference when adding the details. Sketching is also frequently used in floral art design. An artist first sketches the conceptual design of a flower arrangement and then composes real flowers according to the design sketch (Figure 1(a)).

Unfortunately, existing plant modeling systems are mostly engineering-oriented and do not provide sufficient support for such a creative design process. Most existing systems employ a *bottom-up* process in which the user starts with a detailed specification and the overall appearance only emerges after every detail has been specified. Rule-based systems start with abstract growth rules. Even recent "sketch-based" systems require the user to build each component one-by-one before obtaining the global overview. These bottom-up approaches can sufficiently support the creation of "structurally correct" plant models but they fail to help in making aesthetically driven design decisions. This is especially problematic when designing a scene consisting of multiple flowers, as in flower arrangement models. The user needs to control various details taking overall composition into account, but bottom-up approaches fail to support this process.

Our goal is to provide support for the creative design process in 3D modeling. In contrast to most previous systems, we employ a *top-down (or coarse-to-fine)* design process in which the user first specifies the overall appearance of the final model and adds the details later. The user quickly designs the overall appearance of the model as a simple sketch and gradually transforms the sketch into a detailed 3D model by successive refinement.

This paper introduces a prototype modeling system for a scene consisting of multiple flowers based on our framework. The user first constructs a coarse reference model as a collection of strokes projected onto hierarchical billboards (Figure 1(b) and (c)). Each billboard corresponds to a plant component such as a branch, leaf, or flower in a branch hierarchy. Then the user gradually transforms the initial sketch into a complete 3D model by replacing the billboards with 3D components one-by-one (Figure 1(e)). By reusing individual components, this refinement process can become fairly efficient since the same components appear repeatedly in plants. To allow the user to refer to the global overview at any time to make appropriate decisions in designing details, the system provides a separate overview window that shows the overall appearance of the current model.

The smooth transition from a coarse sketch to a detailed model has been a major issue in sketch-based modeling. Many systems have achieved rapid construction of coarse 3D models, but the construction of detailed 3D models from these coarse models has not been adequately supported even though such a feature would be highly desirable. Our system is one of the first attempts to tackle this problem. The key observation is that plants consist of many discrete components and each component can be efficiently represented by a few strokes. Therefore, it is natural to replace these strokes with the detail components later.

2. Background

Our work is based on existing knowledge from many different fields. This section briefly reviews representative references from the three most relevant fields: floral art, plant modeling, and sketch-based 3D modeling.

Floral art: Plants have been interesting subjects of art, and a great number of articles about floral art have been published. Assmann's *Contemporary Floristry* [Ass96] discusses various aspects of European floral art and basic flower designs. The book efficiently uses sketches to present the composition of each design. We based the flower models used in this paper on Assmann's book.

Plant modeling: The L-System, the most common tool for plant modeling [MP96; PHHM96], was first formulated by Lindenmayer [Lin68] and was introduced to the computer graphics community by Prusinkiewicz and Lindenmayer [PL 90]. In the L-System framework, the user starts the modeling process by writing abstract rules and characterizing components of the plant. Then, the user iteratively modifies the rules and components to obtain the desired shape. Since small local changes can strongly affect the global shape, this bottom-up process makes it difficult to obtain the desired overall appearance. Some systems introduced a top-down approach to rule-based plant modeling. Synthetic topiary [PJM94] simulates plant response to pruning to control the overall shape. Prusinkiewicz et al. [PMKL01] also used positional information to control parameters along a plant's axis. Boudon et al. proposed a bonsai tree modeling system [BPP*03] that provides top-down parameter control using a decomposition graph. However, these systems only specify the outermost boundary of the plants and do not support quick sketching of a more detailed appearance, as seen in Figure 1(d).

Deussen and Lintermann [LD96; DL97; DL99] developed the Xfrog system that combines the power of a rule-based approach and intuitive user interfaces using graph representation. The system provides 11 kinds of graph nodes representing plants components. The user starts with small components and iteratively combines these to create the entire model. This system offers an intuitive interface and the resulting models are highly realistic. However, it is still difficult to design a set of flowers with a specific global appearance by using this bottom-up approach.

Sketch-based 3D modeling: Sketch-based 3D modeling systems allow the user to quickly construct coarse 3D models using gestural interfaces. The user specifies the appearance of the subject model using 2D gestures and the system constructs plausible 3D models from these gestures. Various domain-specific modeling systems have been proposed such as CSG models [ZHH96], free-form surfaces [IMT99], 3D curves [CMZ*99], and clothing [TCH04].

Some recent systems have applied sketching interfaces to plant modeling. Okabe et al. [OOI05] presented a sketch-based tree modeling system. They infer the 3D geometry of a tree based on the assumption that trees tend to maximize the distance between branches. Ijiri et al.

[IOOI05] introduced a flower modeling system using floral diagrams and inflorescences for structure definition. Both systems use sketching to control the geometry but do not use it as a guide for detailed modeling.

The above systems use sketching as a tool to construct traditional 3D models, but some systems explore the possibility of using sketching itself as a new design medium. The projective stroke system [TDM99] projects the user's strokes onto a sphere surrounding the viewpoint and the Harold system [CHZ00] projects strokes to billboards to represent a quasi-3D scene. Bourguignon et al.'s system allows the user to place strokes in the air that indicate local surface contours [BCD01]. We take a similar approach for representing the initial sketch but also provide a way to transform this sketch into a complete 3D model.

3. Overview of the Modeling Process

Our proposed system represents the initial sketch of a plant as a collection of free-form strokes on hierarchical billboards (Figure 2(a)). Each billboard corresponds to a basic plant component such as a branch, leaf, or flower and is embedded in the natural branching structure of a plant. Each billboard contains a main skeleton stroke and a collection of decorative strokes. A skeleton stroke represents the geometry of the component's main axis. The decorative strokes indicate the overall appearance of the component. The user constructs this coarse model interactively with simple sketching and a minimum number of structure control operations. This representation is fairly straightforward to construct and is sufficiently expressive as an initial design medium. Furthermore, it contains fundamental structural information that is used in the following detailed 3D modeling process.

The user can represent the initial design as a very coarse sketch and then refine the design by drawing additional decorative strokes or adding more billboards to the hierarchy. For example, the user might initially represent a flower as a simple circle but can then represent it with many decorative strokes (Figure 2(b)). The user can also represent a branch as a collection of decorative strokes on a billboard first and subsequently decompose it into multiple billboards (Figure 2(d)).

After completing the sketch, the user transforms it to a complete 3D model by replacing billboards one by one with 3D components (Figure 2(b)-(d)). The user can either create a new 3D component or reuse existing components from a library. In this replacement process, the skeleton of each 3D component is adjusted according to the skeleton stroke on the billboard.

Three types of floral components are available in our system: stems, leaves (or petals), and flowers (Figure 3). A skeleton is defined for each component and used to reconfigure its geometry in the 3D modeling process. A stem is represented as a generalized cylinder; its skeleton is its axis. A leaf is represented as a B-spline surface; its skeleton is the central row of its control points. A flower is a composite representation consisting of a base receptacle with floral elements arranged on it; the receptacle is represented as a revolving surface and its skeleton is its axis. The hierarchical structure of plants is represented by stem components having arbitrary components as branching children, e.g. lateral branches and leaves. The configuration of a child branch is specified by the rotate angle, the down angle, and the relative position along the parent stem in the system (Figure 3(d) and (e)).

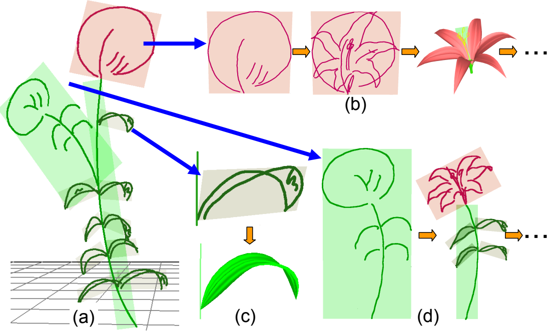

Figure 2: *Initial sketch as hierarchical billboards. The user starts with sketches on a billboard (a), and then replaces each sketch either with detailed sketches (b) and (d) or 3D objects (c).*

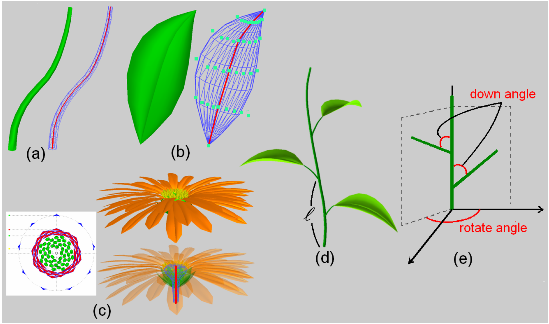

Figure 3: *Component representation; stem (a), leaf (b), flower and its floral diagram (c), and stem with branches (d). The skeleton of each object is highlighted in red.*

4. User Interface of the Prototype System

Our system consists of a main edit window and an overview window (Figure 4). The main edit window is further divided into a 3D workspace pane, a component library pane, and a floral diagram pane. The 3D workspace is where the actual modeling operations occur. The user constructs a coarse sketch and gradually transforms the sketch into a complete 3D model in this workspace. The component library pane shows the collection of existing 3D components and allows the user to add any of them to the model in the workspace. The floral diagram pane shows the floral diagram editor [IOOI05] and serves as an interface for defining an arrangement of flower components on a single flower.

The main workspace supports fast view navigation to help the user work with the hierarchical structure. The main workspace is always “focused” on a specific billboard or component in the plant hierarchy. Initially, the workspace is focused on the base scene but the user can change the focus by simply clicking on the target component. Then the camera automatically “flies” to the component so that the component appears at the center of the workspace pane with maximum size. Non-focused objects are shown transparently. This view navigation is indispensable for working on the details, but the global view of the scene is lost in this process.

The system provides an overview window to show the global appearance of the current model. This global view is important during the design task because it allows the user to always see the final picture and helps him or her make appropriate design decisions during the modeling process. For example, it can prevent the user from spending too much time in designing the details of a branch that is too small to be seen or covered by other branches in the global view. The global window can also show auxiliary elements such as a vase so that they are taken into account in making design decisions. It is also possible to use advanced rendering methods such as ray tracing to preview the final rendering result because this window does not require real-time feedback.

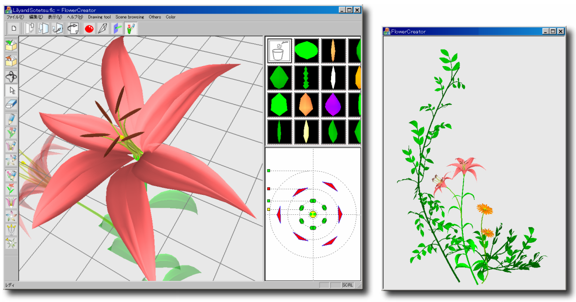

Figure 4: *Snapshot of the system. Our system consists of a main edit window (left) and an overview window (right).*

5. Construction of an Initial Sketch

The user creates an initial sketch as hierarchical billboards. To add a new billboard to the hierarchy, the user presses the “create a new billboard” button and draws a skeleton stroke. The starting point of the stroke determines the parent. If the stroke starts from the ground, it becomes a root branch (Figure 5(a)). If the stroke starts from a skeleton stroke on an existing billboard, the new stroke is attached to the existing skeleton stroke (Figure 5(b)). The new stroke drawn on the screen is projected onto a plane that is parallel to the screen and passes through the 3D position where the new stroke is attached. A new billboard then appears around the skeleton stroke.

The “create a new billboard” button actually consists of three separate buttons, each of which corresponds to a component type. One is for creating a billboard representing a branch, another is for a leaf, and the third is for a flower. The type of the new billboard is determined by which button is pressed before drawing the skeleton stroke. The newly created billboard is marked as active, and the user can add an arbitrary number of decorative strokes to the active billboard. The user can deactivate the current billboard by clicking on the background and activate another billboard by clicking on the stroke associated with the subject billboard. The user can also erase any stroke whenever he or she desires by using the eraser tool.

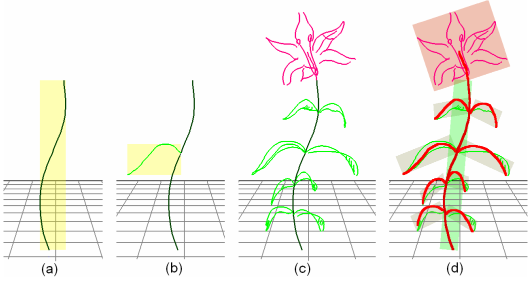

Figure 5: *The construction of hierarchical billboards. The user creates billboards by drawing strokes from the ground (a) and from existing sketches (b). The user then draws more sketches (c) and (d) presents all billboards; the skeleton is shown in red.*

6. Transformation of the Sketch into a 3D Model

Given the initial sketches, the next task is to transform the hierarchical billboard to an actual 3D model by replacing billboards with 3D components, one by one. Our current implementation supports three methods to do this. One is to create a new 3D component on the billboard. The other two methods reuse existing 3D components from the component library. Note that during and after the transformation process, old sketches are overlaid translucently to guide the creation of 3D components.

6.1 Creating a New Components

The first method is to create a new 3D component on a billboard. The actual procedure is as follows. First, the user focuses on the target billboard and then presses the “create 3D component” button to start a modeling process. The actual 3D modeling session is different depending upon the type of the billboard: branch, leaf, or flower. These modeling interfaces are designed based on the system proposed by Ijiri et al. [IOOI05]. In contrast to their system in which the user creates detailed components from scratch, our system gradually transforms a design sketch represented as hierarchical billboards into a complete 3D model. The skeleton stroke of the billboard provides the basic geometric information. We use the skeleton as the central axis of a newly created component, and then the user provides additional geometric information interactively.

If the billboard type is a branch, when the user presses “create 3D component” button, the system generates a generalized cylinder along the skeleton of the billboard (Figure 6(a)). The user then specifies its radius, color, and texture using standard GUIs. An interface is also provided for modifying the shape of the skeleton, which will be described in Section 4.5. If the billboard contains decorative strokes that represent child components, the user can add a detailed sketch by adding a new billboard around the 3D stem (Figure 6(b)).

If the billboard type is a leaf, the system shows a curved surface that contains the skeleton stroke and is perpendicular to the billboard (Figure 7(b)). Then the user draws two strokes representing side contours on the screen and they are projected onto the curved surface. The system uses the skeleton of the billboard and the two strokes as an input of leaf object creation [IOOI05] (Figure 7(c)). The skeletons of the billboards and of the created leaf coincide. The user can also modify the shape of the leaf by drawing modifying strokes that represent cross sections of the leaf object (Figure 7(d)).

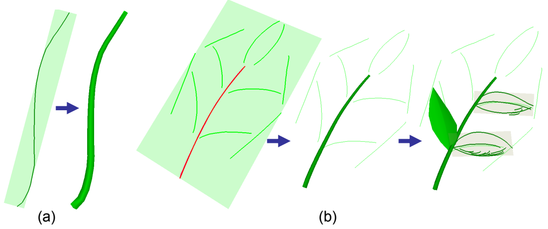

Figure 6: *Construction of a stem. The system generates a cylinder along the skeleton on the billboard (a). The user can later attach billboards or components to the stem (b).*

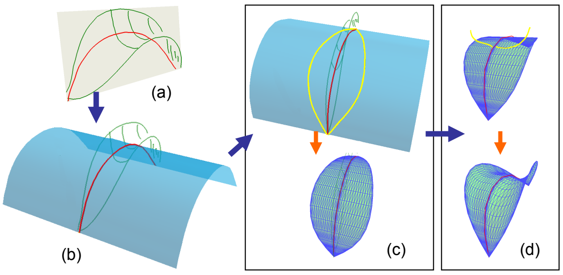

Figure 7: *Construction of a leaf. A billboard (a) is transformed into a leaf. The system presents a curved blue canvas (b), on which the user creates a leaf by drawing two outlines (c). The user can also modify the shape by drawing modifying strokes (d).*

If the billboard type is a flower, the system places a base receptacle object at the location of the skeleton (Figure 8(b)). We use only the start and end points of the skeleton, since the axis of a receptacle is a straight line segment. The system also produces a floral diagram [IOOI05] that represents an arrangement of floral components on the receptacle in 2D form (Figure 8(c)). The user first modifies the size and the shape of the receptacle by drawing its profile, and then arranges icons on the diagram that represent floral parts such as the pistils, stamens, petals, and sepals. Icons are then mapped onto the receptacle in 3D. For each icon, the user generates a billboard and roughly sketches its shape as described in Section 4. Finally the user transforms the billboard into a 3D model. Pistils and stamens are modeled in the same way as stems. Modeling of petals and sepals follows the same process used for leaves.

The flower structure can be too complicated for the user to design floral components on the small receptacle. If so, the user can construct a component in a separate workspace and place it on the receptacle later. Decorative strokes are also useful for guiding this placement process.

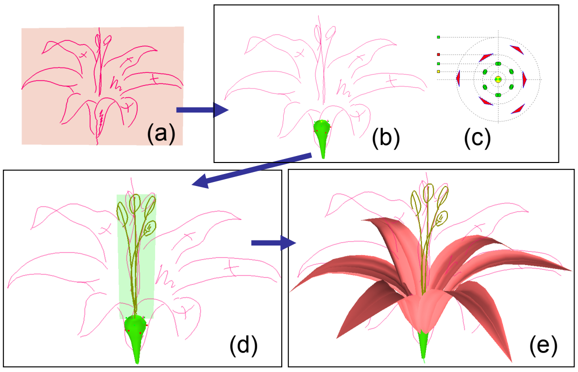

Figure 8: *Construction of a flower. A billboard (a) is transformed into a flower. The system generates a floral receptacle (b) and presents its diagram (c). The user then creates billboards or 3D components and arranges them on the receptacle (d), (e).*

6.2 Reusing a 3D Component

Since a plant model has many repetitive patterns, it is important to be able to reuse 3D components. The 3D component library serves this purpose. The user can add the focused 3D component in the main workspace to the component library pane by pushing the “register” button. A thumbnail of the component then appears in the slot. Our system supports two ways to insert a component from the library into the workspace.

The first method is to replace an existing billboard with the component by using the skeleton information of the billboard (Figure 9). The user starts a session by pressing the “replace billboard” button. The user selects one or more subject billboards and specifies a source component from the library. The system then reconfigures and places the source components so that the main axis of each 3D component is aligned with the corresponding skeleton stroke (Figure 9(c)). The same component can often appear several times at the same level in a plant’s hierarchy. For example, many lateral leaves are attached to a parent branch. To help the user replace multiple billboards at the same level, the system provides an interface for selecting all of them at the same time. If the user double clicks on a billboard, the system automatically searches for and selects all sibling billboards at the same hierarchical level as the target (Figure 9(d)).

The second method is to place a copy of the component at an arbitrary location on the plant without referring to any specific billboard (Figure 10). The user starts a session by pressing the “placing component” button. The user then selects a source component from the library and draws a free-form stroke in the main workspace to specify where to put the component. The system converts the stroke to a 3D skeleton stroke in the same manner that is used for billboard creation. Then the system adds the selected component to the model so that the main axis of the component is aligned with the skeleton stroke. This operation can be seen as a shortcut for billboard creation and its subsequent replacement with a 3D component. This is particularly useful when the user sketches many elements in a single billboard (as shown in Figure 6(b)) or adds details that were omitted in the initial sketch.

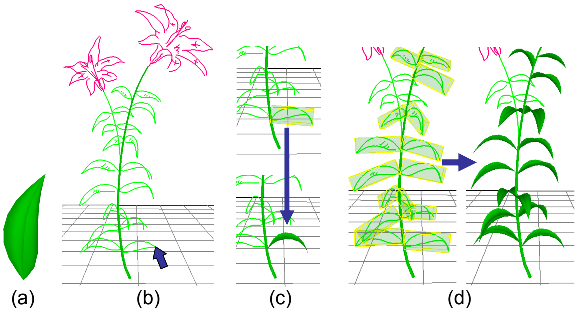

Figure 9: *Billboard replacement. The user clicks to select a component (a) and target billboards (b). The system then replaces billboards with the specified component (c). Users may also double-click to select all sibling billboards (d).*

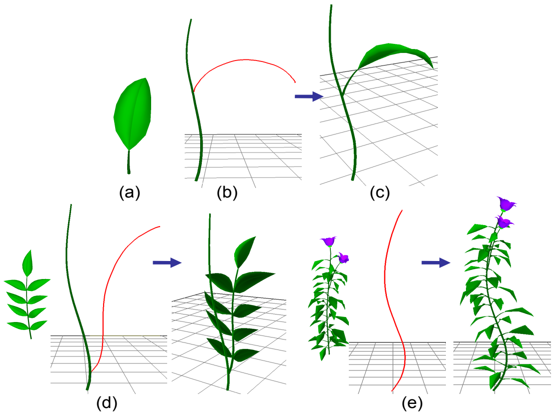

Figure 10: *Placing 3D components. The user selects a source (a) from the library and draws a skeleton stroke (b). The system then places the object (c). Examples of more complicated objects are shown in (d) and (e).*

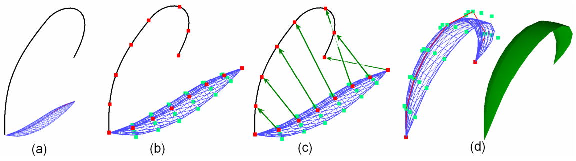

Figure 11: *Fitting a leaf object to the target skeleton.*

In both replacement and copy placement, the system transforms the source component to make its skeleton match the target skeleton. If the source component is a stem, the system reshapes the generalized cylinder using the target skeleton as the new axis and scales its radius using a ratio of the original skeleton length to the target skeleton length. When the stem has child components, the system resizes each child using the length ratio and attaches them to the stem keeping its relative position, rotate angle, and down angle (Figure 10(d) and (e)). If the component is a leaf, the system first adjusts the scale of the leaf (Figure 11(a) and (b)) and resamples the target skeleton according to the number of central control points of the leaf (Figure 11(b)). It also calculates a vector from each central control point to the corresponding point on the target skeleton (Figure 11(c)). The system then moves each central point and its side points using the corresponding vector (Figure 11(d)). If the source component is a flower, the start and end points of the target skeleton are connected to construct a line segment. The system then scales and places the base receptacle to make it match the line segment. At the same time, all components on the receptacle are scaled using the same ratio.

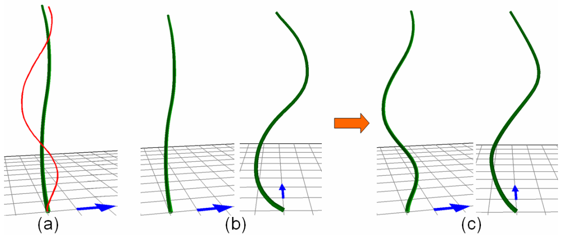

Figure 12: *Skeleton-based deformation. The user draws the stroke shown in red (a) and then the system deforms the target stem (b) so that the stem's skeleton fits the stroke on the screen (c).*

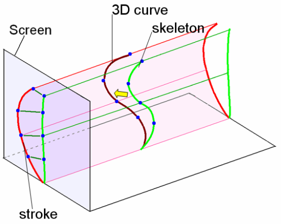

Figure 13: *Fitting the skeleton to the user-drawn stroke.*

7. Skeleton-based Deformation of 3D Components

Our system defines the geometry of a 3D component using a curved 3D skeleton stroke. All skeleton strokes are created from 2D free-form strokes drawn on the screen and subsequently projected onto a flat billboard. As result, all skeletons in this system are planar, i.e. they cannot represent spirals. To remedy this problem and support the creation of more expressive 3D geometry, our system provides an interface for deforming the main axis of the 3D components (Figure 12). If the deformed component is associated with an original sketch, the association is cleared at this moment and the overlaid sketch disappears.

The interface is very simple. The user presses the "deform 3D component" button, selects a target component in the workspace, and draws a free-form stroke specifying the shape of the main axis. The system moves each vertex of the skeleton in parallel to the screen so that the skeleton matches the user drawn stroke on the screen (Figure 13). The user can repeat this process from different viewpoints until the desired 3D shape is obtained. This operation can be seen as a variant of the gestural deformation operation in [IMT99; KG05; CMZ*99].

8. Results and Discussion

Figures 1 and 14 present flower arrangement models and their initial design using sketches. It took less than 3 hours from the design of the coarse prototype to the completion of the fine 3D model. Since our sketch-based prototyping session provides an easy-to-use interface and minimizes the cost of trial-and-error efforts, the user can carefully place multiple flowers and their components considering many aesthetic aspects. The resulting models show that sketches are carefully designed to reflect intricate design decisions and the final fine models strongly preserve the initial intention. In our experience, the copy creation function was particularly useful during the fine modeling process. The copy function does not reproduce an instance in the same way as the original model; rather, it creates a more natural variation using a stroke drawn by the user [SFS05].

Our current implementation reuses the central skeleton only for subsequent 3D modeling and decorative strokes exist only for visual reference. An alternative approach would involve reusing these decorative strokes as well during 3D modeling. We did not choose this option because it inevitably constrains the drawing process and thus interferes with creative design. For example, our system allows the user to draw not only meaningful floral components but also arbitrary annotations as decorative strokes on billboards. It would be difficult to retain such flexibility if the system reused these strokes during 3D model construction.

Our current system is limited in its ability to model huge plants. Although we introduced automatic replacement of multiple billboards or copy creation, our current system basically requires the user to control all details. Therefore, it is almost impossible to create models of huge plants such as trees, which may consist of thousands of branches and even more leaves. In the future, we could manage this problem by combining our method with rule-based approaches or some structure editor to control a highly detailed structure.

9. Conclusions and Future Work

We presented a sketch-based flower modeling system that supports seamless transformation from the initial design to the final detail edit; the user first performs rapid prototyping by sketching onto billboards and then iteratively replaces the billboards with finer sketches or 3D components. The sketch interface facilitates fine-tuning of a user's initial intention without involving any of the difficulties related to 3D modeling. Our replacement interface supports seamless transformation from an initial sketch to a detailed 3D model. The system's automatic replacement of multiple billboards, creation of copies, transparent overlay of initial sketches, and overview window all help to facilitate the 3D modeling process.

In future work, we plan to apply the top-down modeling process into more general 3D models. This is not so straightforward because it is difficult to define arbitrary surfaces with sketching and to gradually refine the model in a unified manner. One interesting direction is to combine our approach with the modeling-by-example method proposed by Funkhauser [FKS*04]. It may be quite helpful to quickly define the overall appearance of the target model initially before starting the actual cut-and-paste modeling session.

Acknowledgments

We would like to thank Prof. Yoichi Takebayashi for his comments and advice based on his experience and thoughtful insights. We also thank the *Eurographics* reviewers for their insightful comments. This work was funded in part by the IPA and JSPS Research Fellowship.

References

[Ass96] ASSMANN P., *Contemporary Floristry*, Appel-Druck Donau-Verlag GmbH, 1996.

[BPP*03] BOUDON F., PRUSINKIEWICZ P., FEDERL P., GODIN C., KARWOWSKI R.: Interactive Design of Bonsai Tree Models. *Computer Graphics Forum*, 22, 3 (2003), 591-599. (Proc. Eurographics '03)

[BCD01] BOURGUIGNON D., CANI M. P., DRETTAKIS G.: Drawing for Illustration and Annotation in 3D, *Computer Graphics Forum,* 20, 3 (2001), 114-122. (Proc. Eurographics '01)

[CMZ*99] COHEN J., MARKOSIAN L., ZELEZNIK R., HUGHES J.F, BARZEL R.: An Interface for Sketching 3D Curves. In *Proceedings of ACM I3D '99* (1999). pp.17-21.

[CHZ00] COHEN J.M., HUGHES J.F., ZELEZNIK R.C.: Harold: A World Made of Drawings, In *Proceedings of NPAR,* 2000, pp. 83-90.

[DL97] DEUSSEN O., LINTERMANN B.: A Modeling Method and User Interface for Creating Plants. *In Proceedings of Graphics Interface '97* (1997), pp. 189-197.

[DL99] DEUSSEN O., LINTERMANN B.: Interactive Modeling of Plants. I*EEE Computer Graphics and Applications*, 19, 1 (1999), 56-65.

[FKS*04] FUNKHOUSER T., KAZHDAN M., SHILANE P., MIN P., KIEFER W., TAL A., RUSINKIEWICZ S., DOBKIN D.: Modeling by Example, *In Proceedings of ACM SIGGRAPH '04* (2004), ACM, pp. 652-663.

[IMT99] IGARASHI T., MATSUOKA S., TANAKA, H.: Teddy: A Sketching Interface for 3D Freeform Design. *In Proceedings of ACM SIGGRAPH '99* (1999), ACM, pp. 409-416.

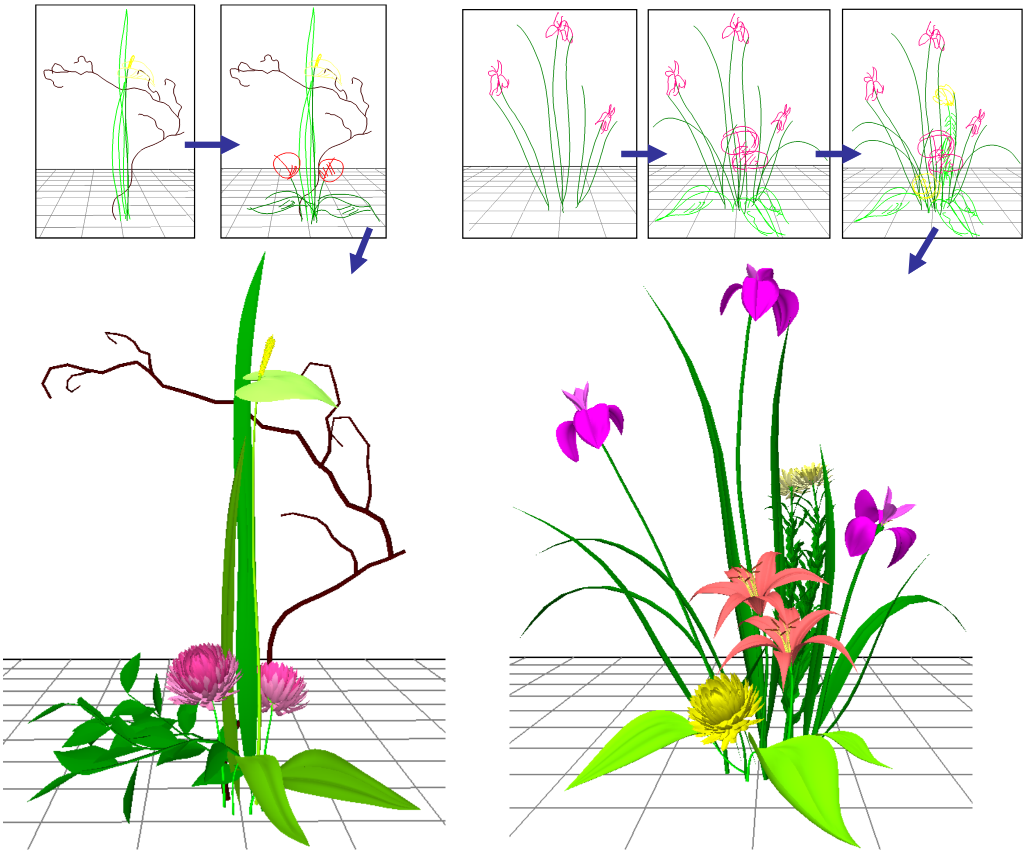

Figure 14: *Flower arrangement models designed using our system.*

[IOOI05] Ijiri T., Owada S., Okabe M., Igarashi T.: Floral Diagrams and Inflorescences: Interactive Flower Modeling Using Botanical Structural Constraints. *In Proceedings of ACM SIGGRAPH '05* (2005), ACM, pp. 720-726.

[KG05] Kho Y., Garland, M.: Sketching Mesh Deformations. In *Proceedings of the 2005 Symposium on Interactive 3D Graphics and Games,* pp. 147-154.

[Lin68] Lindenmayer A.: Mathematical Models for Cellular Interactions in Development, I & II. *Journal of Theoretical Biology* (1968), 280-315.

[LD96] Lintermann B., Deussen O.: Interactive Modelling and Animation of Branching Botanical Structures. *In Proceedings of Eurographics Workshop on Computer Animation and Simulation '96* (1996), 139-151.

[MP96] Měch R., Prusinkiewicz P.: Visual Models of Plants Interacting with Their Environment. *In Proceedings of ACM SIGGRAPH '96* (1996), ACM, pp. 397-410.

[OOI05] Okabe M., Owada S., Igarashi T.: Interactive Design of Botanical Trees Using Freehand Sketches and Example-based Editing. *Computer Graphics Forum,* 24, 3 (2005), 487-496. (Proc. Eurographics '05)

[PL90] Prusinkiewicz P., Lindenmayer A.: *The Algorithmic Beauty of Plants.* Springer-Verlag, New York, 1990. With J. S. Hanan, F. D. Fracchia, D. R. Fowler, M. J. M. de Boer, and L. Mercer.

[PHHM96] Prusinkiewicz P., Hammel M., Hanan J., Měch R.: L-systems: From the Theory to Visual Models of Plants. In *Proceedings of the 2nd CSIRO Symposium on Computational Challenges in Life Sciences* (1996).

[PJM94] Prusinkiewicz P., James M., Měch R.: Synthetic Topiary. In *Proceedings of ACM SIGGRAPH '94* (1994), ACM, pp. 351-358.

[SFS05] Streit L., Federl P., Sousa M. C.: Modelling Plant Variation Through Growth. *Computer Graphics Forum, 24, 3* (2005), 487-596. (Proc. Eurographics '05)

[PMKL01] Prusinkiewicz P., Mündermann L., Karwowski R., Lane B.: The Use of Positional Information in the Modeling of Plants. In *Proceedings of ACM SIGGRAPH '01* (2001), ACM, pp. 289-300.

[TDM99] Tolba O., Dorsey J., McMillan L.: Sketching with Projective 2D Strokes. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, Asheville, NC. 1999.

[TCH04] Turquin E, Cani M. P., Hughes J. F.: Sketching Garments for Virtual Characters, *Eurographics Workshop on Sketch-Based Interfaces and Modeling*, 2004.

[ZHH96] Zeleznik R. C., Herndon K. P., Hughes, J. F.: SKETCH: An Interface for Sketching 3D Scenes. In *Proceedings of ACM SIGGRAPH '96* (1996), ACM, pp. 163-170.