

# ShadowDraw パラダイムに基づく 例示ベース樹木モデリング

萬屋 宇人<sup>1,a)</sup> 藤代 一成<sup>1,b)</sup>

**概要:** 樹木モデリングは、複雑な枝葉構造をもつ樹木の生成規則やモデリングの簡易化を追求するなど、従来からさまざまな研究が行われている。本研究では、モデリングの簡易化を重視し、ShadowDraw パラダイムを利用したスケッチベース樹木モデリングを提案する。ユーザはスケッチを描くスキルを必要としないだけでなく、現存する樹木の生成規則に則したモデルの影をガイドとして利用できるため、現実味のある 3次元樹木モデルを簡単に生成することができる。

**キーワード:** 樹木モデリング, 生成規則, スケッチ入力, 例示ベースモデリング

## An Example-Based Approach to Tree Modeling Based on ShadowDraw Paradigm

TAKATO YOROZUYA<sup>1,a)</sup> ISSEI FUJISHIRO<sup>1,b)</sup>

**Abstract:** Trees have complicated structures and a wide variety of appearance. For that reason, tree modeling has been widely studied to pursue generative rules of tree shape as well as simplified modeling process. In this study, we focus particularly on the simplification of modeling process, and propose a sketch-based approach to tree modeling based on ShadowDraw paradigm. The paradigm does not only require the users to have any specific drawing skills, but allows them to sketch tree structures using as a guide, underlying shadow images of trees produced by obeying generative rules and then to obtain realistically-shaped 3D tree models easily.

**Keywords:** Tree modeling, generative rules, sketch input, example-based modeling.

### 1. 序論

樹木モデリングは、コンピュータを用いた 3次元樹木モデルの生成を目的として、今日も研究され続けている、長い歴史をもつ分野である。樹木は、複雑な枝葉構造や多種多様な外観をもつため、モデリングのなかでも最も難しい対象の一つとされている。そのため、樹木の植物学的な規則やモデリングの簡易化を追求するなど、従来からさまざまな研究が行われてきた [1]。また、樹木モデルは応用範囲がひじょうに広く、多くの分野/用途で利用されている。例えば、

建築物や都市開発の完成予想図等の景観設計、映画やゲーム等のエンタテインメントに利用されるだけでなく、植物学の研究やドライブシミュレータをはじめとした学問/教育などにも利用され、重要な役割を担っている。

樹木モデリングは、基本的には見た目に写実的な 3次元樹木モデルの生成を目的とするが、その実現方法の違いから、ルールベース/イメージベース/スケッチベースの 3つの手法に分類できる。ルールベース手法は、植物の生成規則に基づく 3次元樹木モデルの生成を目的とする手法であり、最も現実味を帯びた枝葉構造を実現できる可能性が高い。しかし、効果的なパラメタの設定には生物学の専門知識や広範囲にわたる手作業が必要となり、初心者が扱うにはひじょうに難しい手法である。イメージベース手法は、

<sup>1</sup> 慶應義塾大学  
Keio University

a) yorozuya@fj.ics.keio.ac.jp

b) fuji@ics.keio.ac.jp

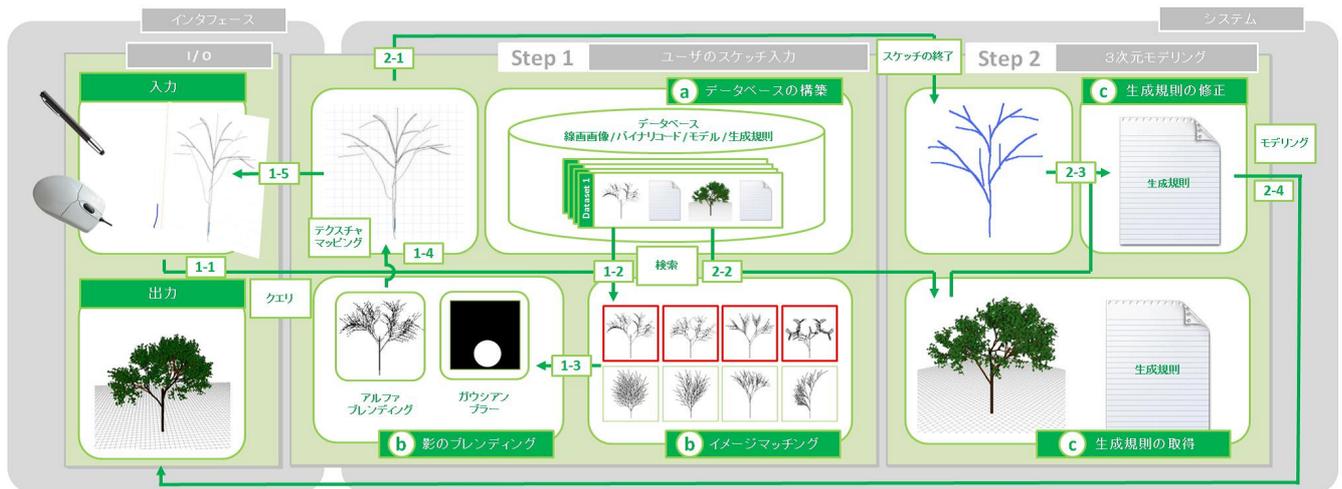


図 1 提案システムにおける処理の概要

樹木画像を用いた 3 次元樹木モデルの生成を目的とする手法であり、実際の樹木画像を用いているため、見た目には最も写実的な樹木モデルを生成することができる可能性がある。しかし、現存する樹木にない全く新しい形状をもつ樹木モデルを生成することは難しい。スケッチベース手法は、スケッチ入力による 3 次元樹木モデルの生成を目的とする手法であり、その直感的な入力方法から、初心者でも簡単に素早くモデルを生成することができる。しかし、ユーザの描く樹木スケッチは架空の枝葉構造であるため、植物学的な枝葉構造の実現が難しい手法である。

本研究では、樹木モデリングの簡易化を重視し、ShadowDraw パラダイム [2] を利用したスケッチベース樹木モデリングを提案する。ShadowDraw パラダイムを導入することで、緩いスケッチナビゲーションのもとで、ユーザはスケッチを描くことができ、スケッチのインタラクティブ性を強化することができる。また、実際の樹木に則したモデルをガイドとして利用できるため、既存のスケッチベース手法に比べて、より生成規則に準拠した樹木スケッチを描くことができると考えられる。

本研究で提案するシステムは、ShadowDraw パラダイムに基づく対話的なスケッチ描画と 3 次元化の 2 つのステップにより、ユーザの意思を反映しながらも、生成規則に準拠した現実味のある 3 次元樹木モデルを生成できる (図 1)。本システムは、あらかじめ現存する樹木に則したデータセットをデータベース上に格納している。ユーザスケッチに最も類似するデータセットの再構築を自動的に行うことで、ユーザのパラメタの設定や広範囲な手作業を必要とすることなく、樹木モデルを生成することができる。

## 2. 関連研究

### 2.1 スケッチベース手法

スケッチベース手法は、樹木モデリングの 3 つの手法の

中でもモデリングの簡易化に特化した手法であるといえる。スケッチ入力による直感的な操作により、初心者でも簡単に素早く樹木モデルを生成することができる。

Okabe et al.(2005)[3] は、樹木スケッチを枝間の距離が最大になるように 3 次元の枝パターンを構築することで、3 次元樹木モデルを生成する手法を提案している。このシステムでは、ジェスチャで枝を追加/削除/切断する先進的な編集に加え、あらかじめ作成しておいた樹木モデルを用いて枝葉を付加する編集も可能である。Ijiri et al.(2006)[4] は、ルールベースの代表的な手法である L-System[15][16] を利用することが大きな特徴であり、樹木の成長を制御する 1 本のストロークを描くだけで、3 次元樹木モデルを生成する手法を提案している。このシステムでは、ユーザの描くストロークの形状の変化により、L-System のパラメタを修正するため、初心者には扱いづらい手法である L-System を簡易化したシステムともいえる。Chen et al.(2008)[5] では、マルコフ確率場を利用し、データベース上の樹木モデルから得られたパラメタに基づいて最適化を行うことで、ユーザスケッチの 3 次元化を可能にしている。Anastacio らは、スケッチによる L-System のパラメタ制御の研究 [6][7] を行っている。近年では、Wither et al.(2009) の葉のシルエットのスケッチを利用する研究 [8] や、Liu et al.(2010) の樹木構造/深度値の情報をそれぞれ保持した 2 つのスケッチを利用する研究 [9][10][11] が行われ、従来のスケッチの描画対象/描画方式にとらわれない研究に注目が集まっている。また、Jerry et al.(2011)[12] は、手続き的にモデルを生成する手法を提案している。

本研究では、ShadowDraw[2] を導入することで、スケッチナビゲーションに基づいてインタラクティブ性をより強化することができるだけでなく、枝葉構造の描画を主とするスケッチベース手法において、生成規則に準拠した枝葉構造を実現することができる。

## 2.2 スケッチアシスタンス

スケッチアシスタンスは、ユーザの描くスケッチに対するインタラクティブな研究であり、本研究においても重要なテーマの一つである。本研究では、Lee et al.(2011)[2]、および Limpaecher et al.(2013)[13] の考え方を利用し、スケッチベース手法の新たな可能性を提示する。

Lee et al.(2011) の ShadowDraw では、描きかけのスケッチからユーザが描こうとしている対象を実行時に推測し、それに類似する形状をもつ複数の関連画像を検索/合成して、それらをキャンバスの背後に影として動的に表示する。これにより、単一画像のトレースとは異なり、第一人称性を失わずに、ユーザの入力をナビゲートすることが可能なスケッチパラダイムとして一般化されている。本研究では、従来のスケッチベース手法と同様に、枝情報を樹木構造の主特徴ととらえ、それを影として表示することで、樹木スケッチのナビゲーションを図る。

Limpaecher et al.(2013) の DrawAFriend では、ユーザの描きかけの顔スケッチに対し、クラウドソーシングで得られた多くのスケッチを平均化した線画画像に基づいて描線を修正する。小さなデバイスにおいても、ユーザスケッチの味を残したまま、ユーザの意図したスケッチを得ることができる。なお、得られたユーザスケッチを新たにデータベースに格納することで、自己改善するデータベースを実現している。本研究では、ユーザスケッチの味を残したまま、架空のあいまいな枝葉構造を生成規則に準拠した枝葉構造に補正することで、樹木スケッチの修正を図る。

## 3. データベースの構築

データベースの構築では、現存する樹木に対する、生成規則、樹木モデル、バイナリコード、線画画像の4つの要素から構成されるデータセットの作成とそのデータベースへの格納を行うことで、後続のイメージマッチングの高速化、3次元樹木モデルの再構築に資する(図1(a))。

図2に、実際に本システムのL-Systemにより生成したデータセットの例を示す。

### 3.1 L-Systemの開発

本研究では、本システムに特化したL-Systemを開発/利用している。元々のL-Systemは、1968年にLindenmayerらにより提案され[15][16]、Prusinkiewiczらにより改良された[17][18]、ルールベース手法を代表する手法であり、生成規則を数学的な記号に置換し、定性的な記述により樹木モデルを生成する。一方、本研究におけるL-Systemは、生成規則を数字に置換し、定量的な記述により樹木モデルを生成する。このL-Systemは、元々のL-Systemと同等の性能を発揮することができ、生成規則を定量的に記述することから、樹木モデルの完成図と必要な操作方法をイメージしやすく、比較的簡単に多くの異なる形状をもつ樹木を生

成することができる特長をもつ。

生成規則は、制御点数、枝分岐数、再帰回数、年輪半径、根ノードの制御点の座標、子ノードの自己相似比を順に格納した、CSV形式のテキストデータである。なお、子ノードの自己相似比は、それぞれの子ノードの親ノードに対する相対位置比、奥行きに関する相対回転角度、高さに関する相対回転角度、相対縮小比を順に格納したものであり、枝分岐数が大きいほど、そのデータ量は比例して大きくなる(図2(a))。

### 3.2 データセットの作成

樹木モデルは、生成規則を再帰回数だけ再帰的に繰り返すことで生成することができる。樹木モデルの枝構造については、内部ノードの制御点を結ぶ線分を芯線とするスイープで表現し、葉構造については、葉に見立てた六角形のポリゴンで表現する。本システムでは、ユーザの3次元化された樹木モデルとの比較/評価のため、512 × 512画素のJPEG画像としてデータベース上に格納する(図2(b))。

バイナリコードは、子ノードの自己相似比における奥行きに関する回転を除く情報を2値化し、順に格納することで得ることができる。なお、相対位置比、相対縮小比に関しては、 $m_1$ 段階で表現し、高さに関する相対回転角度については、あらかじめ定めた範囲に基づいて $m_2$ 段階に分類することで値を割り当てる。値を範囲として分類することで、ユーザスケッチとの比較におけるあいまい検索の実現を図る。相対位置比、相対縮小比、高さに関する相対回転角度の順にコードに加えることで、単一のバイナリコードを生成する(図2(c))。生成規則と同様に、枝分岐数が大きいほど、そのデータ量は比例して大きくなる。

線画画像は、生成規則において、奥行きに関する回転を考慮せず、再帰的に繰り返すことで生成することができる。これは、ユーザがスケッチを描くとき、主に横方向に広がる枝を描き、奥行きを考慮せずに樹木の全体形状を描く傾向があるとする仮定に基づいている。なお、そのモデルについては、内部ノードの制御点を結ぶアンチエイリアシングされた黒色の実線で表現する。また、モデルの生成において、再帰回数を4回に限定することで、ユーザスケッチの描画における影の複雑化の防止を図る。本システムでは、影の表示のため、512 × 512画素のJPEG画像としてデータベース上に格納する(図2(d))。

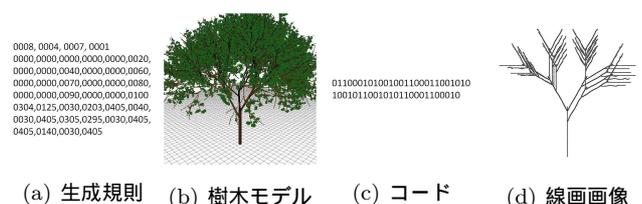


図2 データセットの例 ( $m_1 = 16, m_2 = 8$ )

## 4. イメージマッチング

イメージマッチングでは、ユーザの描線が加わるたびに、その描きかけのスケッチに類似する複数のデータセットをデータベース上から動的に検索し、得られたデータセットの線画画像を影として表示する(図 1(b)).

ShadowDraw[2] では、ユーザスケッチおよびデータベース上の線画画像を小さなパッチに分割し、実行時にそれぞれのパッチ内のエッジ情報を比較することで、ユーザスケッチに類似するデータセットをデータベース上から検索する。一方、本システムでは、樹木構造の特徴を位相的に分析し、実行時にそれぞれの位相を比較することで、類似するデータセットを検索する。本システムでは、樹木構造を位相的にとらえることで、スケッチのあいまいさを許容し、スケッチの描画位置/描画スケールに依存することなく、一連の処理を行うことができる。スケッチの描画対象を樹木に限定することで、多くのオブジェクトをスケッチの描画対象とする元々の ShadowDraw に比べ、より自由で本質的なスケッチを行うことができると考えられる。

イメージマッチングで得られた複数のデータセットを、ユーザが常に視認できるように、線画画像を動的に合成し、ユーザの描きかけのスケッチの背後に影として表示する。

### 4.1 スケッチの補正

樹木の枝葉構造はフラクタルな形状であるといわれている。ユーザが樹木スケッチを描く際、奥行きを考慮せずに樹木の全体形状を描く傾向があると仮定しても、描かれたスケッチはフラクタルな形状をもつべきであり、仮にそうでない場合には、スケッチのあいまいさととらえることができる。ユーザスケッチをフラクタルな形状に補正し、そのバイナリコードを生成する。

ユーザのスケッチについて、構造の中心となる根ノードとそれを親とする子ノード、その子ノードを親とする孫ノード、の順に内部ノードを走査する。そして、親子関係にあるエッジの対について、親に対する子の相対位置比、相対縮小比に関する情報の整数値をコードで表す。また、あらかじめ定めた範囲に基づいて、親に対する子の相対回転角度に関する情報もコードに加える。以上のコード生成を内部ノードに対して再帰的に実行し、世代間で加重平均することで、ユーザスケッチのあいまいさを考慮した、樹木全体を表す単一のコードに集約する。また、各エッジの制御点についても、世代間で加重平均する。これらの加重平均は、階層が根ノードに近いほど、そのノードの情報が重要であるという仮定に基づき、重みづけ係数を  $n_1(0 \leq n_1 \leq 1)$  し、親ノードと子ノードを  $n_1 : (1 - n_1)$  で重みづけする。

### 4.2 スケッチの解析

本システムは、ユーザの意図しない位置/スケールによ

って描画されたスケッチにも対応することができる。本研究では、 $L$ -関数法を利用することによって、ユーザの描きかけのスケッチを点群として、その分布を常に解析している。

$L$ -関数法は、 $K$ -関数法における描かれた点の個数/密度の影響を取り除くために基準化されたものであり、ある点からの距離  $h$  の値により、どのくらいの空間的範囲で点が集中/分散しているかを判断することができる[14]。 $L$ -関数法は式(1)で表され、 $K$ -関数法は式(2),(3)で表される。なお、 $\theta(h)$  は入力が正ならば1を、入力が負ならば0を出力する階段関数である。また、対象とするある平面の面積を  $S$ 、平面上に存在する点を  $\vec{p}$ 、点の総数を  $n$  とする。本システムでは、閾値  $L_{threshold}$  を設け、式(1)により得られた  $L(h)$  に対して、点の分散値に基づく分類を行う。 $L(h)$  が  $L_{threshold}$  より大きいならば、ユーザスケッチは縮小されたサイズで描かれていると推測することができ、 $L(h)$  が  $L_{threshold}$  より小さいならば、ユーザスケッチは意図したサイズで描かれていると推測することができる。

図3に、影の生成の様子を示す。ユーザの意図しない位置/スケールでのスケッチに対しても、インタラクティブに影が対応している様子がわかる。

$$L(h) = \sqrt{\frac{K(h)}{\pi}} - h \quad (1)$$

$$K(h) = \frac{1}{n\lambda} \sum_{i=1}^n \sum_{j=1, j \neq i}^n \theta(h - |\vec{p}_i - \vec{p}_j|) \quad (2)$$

$$\lambda = n/S \quad (3)$$

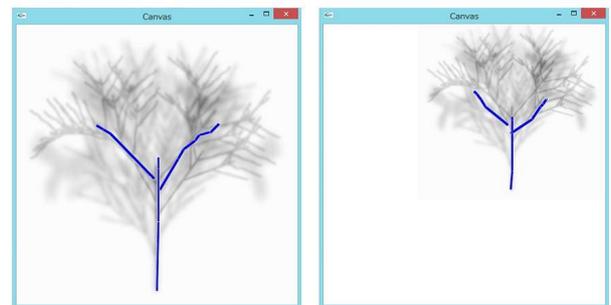


図3 インタラクティブな影の生成

### 4.3 イメージマッチング

描きかけのユーザスケッチとデータベース上のデータセットとのバイナリコードを比較することで、スケッチに類似する複数のデータセットを動的に検索する。

それぞれのデータセットにおいて、ユーザスケッチとの各子ノードの対応関係を求める。対応すると考えられる各子ノードの位相差の線形和が最も小さくするとき、各子ノードがそれぞれ対応していると見なすことができ、その最小値が最も小さいデータセットを、ユーザスケッチに最も近い位相をもつ枝葉構造として選択する。

#### 4.4 影のブレンディング

イメージマッチングで得られたユーザスケッチに類似するデータセットの線画画像を影として表示する。

まず、得られた線画画像にマッチスコアに応じて強弱をつけたガウシアンブラーをかけ、それらをアルファブレンディングによって合成する。次に、ユーザの操作するマウス/タッチペンの位置を中心とする円についての2値のマスク画像を生成し、その円の内外でガウシアンブラーの強弱に変化を加える。このように、マッチスコアやマウス位置に応じてガウシアンブラーに強弱をつけることで、影として表示されているデータベース上の複数の類似画像を気にかけることなく、ユーザはスケッチを描くこともでき、スケッチの自由/本質を保持することができると考えられる。最後に、合成された画像をキャンパスの背後にテクスチャマッピングすることにより、動的に影を生成する。

### 5. 3次元モデリング

L-System[15][16]を用いて、ユーザスケッチを3次元化する(図1(c))。イメージマッチングで得られた最も類似するデータセットの生成規則を取得し、制御点の座標、子ノードの自己相似比を書き換える。書き換えにおいて、重みづけ係数を $n_2(0 \leq n_2 \leq 1)$ とし、ユーザスケッチとデータセットを $n_2 : (1 - n_2)$ で重みづけする。

ユーザはいつでもスケッチを3次元化することができる。完成した樹木モデルに対して、再びスケッチモードに切り替えてスケッチを描き足すことができ、また、再帰回数を増減させることで樹木の自動生長を行うこともできる。

### 6. 結果

#### 6.1 実行環境と開発環境

本システムの実装には、Samsung SERIES 7 SLATE PC (700T1A-A01) (CPU: Intel(R) Core(TM) i5-2467M, クロック周波数: 1.60GHz, メインメモリ: 4.00GB, OS: Windows(R)8 Enterprise 64bit)を用いた。スケッチ描画時の入力デバイスとしてマウス/タッチペンに対応している。なお、開発環境として、Microsoft Visual Studio Professional 2012[19]を用い、OpenGL[20]、およびOpenCV[21]のライブラリの一部を利用している。

#### 6.2 結果と評価

これまで試験的に16の樹木情報をデータベース上に格納し、ユーザスケッチに類似する構造をもつ4の樹木の線画画像を影として表示することで、予備評価を行った。

図4に実行過程の様子、および生成された樹木モデルの様子を示す。常にユーザの描くストロークに近い位相をもつ樹木画像がキャンパス上に重畳されている様子(図4(a))、得られたユーザスケッチが3次元化されている様子(図4(b))がわかる。また、図5に本システムを利用して生成

した樹木モデルの例を示す。なお、いずれの実行結果でも、 $n_1 = 0.9$ ,  $n_2 = 0.5$ ,  $h = 10$ ,  $L_{threshold} = 15$ とした。

ユーザテストでは、背後の影により、初めからから樹木のスケッチの描き方を知る必要性がなく、スムーズに素早く描くことができる点が評価された。しかし、生成された樹木モデルに対して、葉の形状や全体形状が思ったものではないとの意見もあり、改善すべき点であると考えられる。

### 7. 結論

本研究は、従来のスケッチベース手法に、インタラクティブ性の強化と生成規則に準拠した枝葉構造を実現し、スケッチベース手法の新たな可能性を提示するものである。さらに、提案手法は、現存する樹木に則したモデルを集合知として再利用し、ユーザスケッチに基づいてそれらの全体形状を再構築する点で例示ベース手法とも位置づけられ、既存のカテゴリに当てはまらない新たな樹木モデリングのポテンシャルも有すると考えられる。

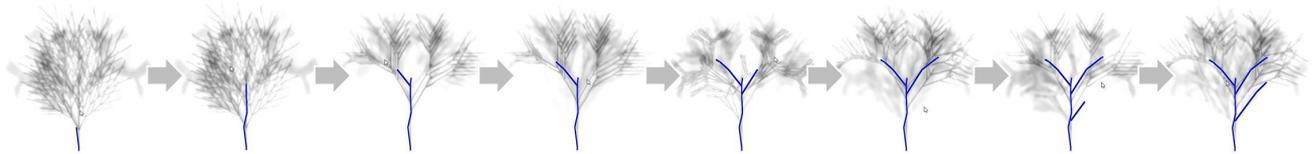
今後の課題として、従来のスケッチベース手法と同様に、樹木の全体形状を制御する樹冠情報をスケッチの描画対象として加えることが挙げられる。さらに、データベースの高速検索の手法/索引アルゴリズムを検討する必要があると考えられる。本研究はデータベース上の樹木にユーザのインタラクションを加えた例示ベース手法であるため、格納しているデータセットが小さければ、そのパフォーマンスを最大限に発揮することができないといえる。最後に、モデリングシステムの改良が挙げられる。現時点において、生成モデルは、簡単な樹木の枝葉構造のみを構築するため、写実性を追求する場面での利用を連想しづらい。葉/樹皮のレベルでモデリングを行うことで、その場面での利用における評価も行うことができるだけでなく、本研究の目的を視覚的に明確化できると考えられる。

### 謝辞

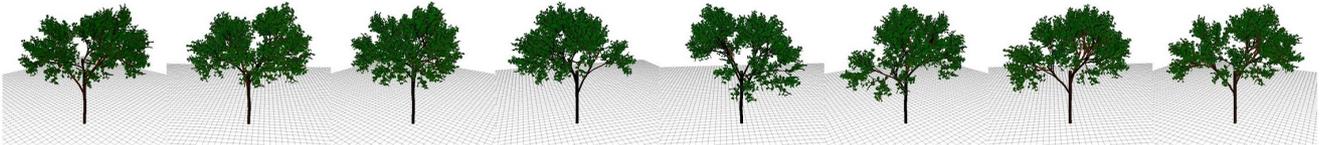
本研究の一部は、科研費基盤研究(B)25280037の支援により実施された。

### 参考文献

- [1] Sing Bing Kang and Long Quan: *Image-Based Modeling of Plants and Trees*, Morgan and Claypool Publishers, November 2010.
- [2] Yong Jae Lee, C. Lawrence Zitnick, and Michael F. Cohen: "ShadowDraw: Real-Time User Guidance for Free-hand Drawing," *ACM Transactions on Graphics*, Vol. 30, No. 4, Article 27, July 2011.
- [3] Makoto Okabe, Shigeru Owada, and Takeo Igarashi: "Interactive Design of Botanical Trees Using Free-hand Sketches and Example-based Editing," *Computer Graphics Forum*, Vol. 24, Issue 3, pp. 487-496, September 2005.
- [4] Takashi Ijiri, Shigeru Owada, and Takeo Igarashi: "The Sketch L-System: Global Control of Tree Modeling Using Free-Form Strokes," in *Proceedings of the 6th International Symposium of SmartGraphics*, pp. 138-146, July 2006.



(a) ShadowDraw パラダイムを利用してスケッチを描く様子



(b) 描き終わったユーザスケッチ (a) から生成した 3次元樹木モデルをさまざまな角度で描画した様子

図 4 予備評価の実行結果：スケッチと 3次元化の様子

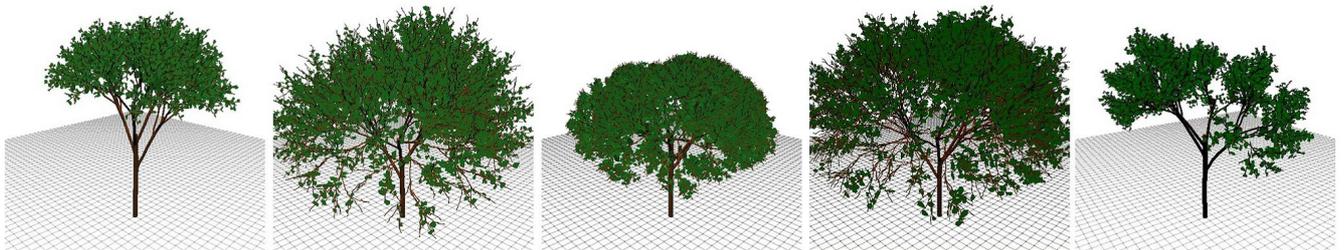


図 5 予備評価の実行結果：提案システムで生成したさまざまな 3次元樹木モデル

- [5] Xuejin Chen, Boris Neubert, Ying-Qing Xu, Oliver Deussen, and Sing Bing Kang: "Sketch-Based Tree Modeling Using Markov Random Field," *ACM Transactions on Graphics*, Vol. 27, No. 5, Article 109, December 2008.
- [6] Fabricio Anastacio, Przemyslaw Prusinkiewicz, and Mario Costa Sousa: "Sketch-Based Parameterization of L-Systems Using Illustration-Inspired Construction Lines," in *Proceedings of the Fifth Eurographics Conference on Sketch-Based Interfaces and Modeling*, pp. 119-126, November 2008.
- [7] Fabricio Anastacio, Przemyslaw Prusinkiewicz, and Mario Costa Sousa: "Sketch-Based Parameterization of L-Systems Using Illustration-Inspired Construction Lines and Depth Modulation," *Computers and Graphics*, Vol. 33, Issue 4, pp. 440-451, August 2009.
- [8] J. Wither, F. Boudon, M. -P. Cani, and C. Godin: "Structure From Silhouettes: A New Paradigm For Fast Sketch-Based Design of Trees," *Computer Graphic Forum*, Vol. 28, Issue 2, pp. 541-550, April 2009.
- [9] Jia Liu, Xiaopeng Zhang, and Hongjun Li: "Sketch-Based Tree Modeling by Distribution Control on Planes," in *Proceedings of 9th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry*, pp.185-190, December 2010.
- [10] Jia Liu, Xiaopeng Zhang, Hongjun Li, and Mingrui Dai: "Creation of Tree Models From Freehand Sketches by Building 3D Skeleton Point Cloud," in *Proceedings of the Entertainment for Education, and 5th International Conference on E-learning and Games*, pp. 621-632, August 2010.
- [11] Jia Liu, Xiaopeng Zhang, Hongjun Li, Weiming Dong, and Jean-Claude Paul: "Sketch-Based Tree-Modeling Using Depth Retrieval," in *Proceedings of Computer Graphics International*, May 2010.
- [12] Jerry O. Talton, Yu Lou, Steve Lesser, Jared Duke, Radomir Mech, and Vladlen Koltun: "Metropolis Procedural Modeling," *ACM Transactions on Graphics*, Vol. 30, No. 2, Article 11, April 2011.
- [13] Alex Limpaecher, Nicolas Feltman, Adrien Treuille, and Michael Cohen: "Real-Time Drawing Assistance Through Crowdsourcing," *ACM Transactions on Graphics*, Vol. 32, No. 4, Article 54, July 2013.
- [14] Philip M. Dixon: "Ripley's K Function," *Encyclopedia of Environmetrics*, Vol. 3, pp. 1796-1803, January 2002.
- [15] A. Lindenmayer: "Mathematical Models for Cellular Interactions in Development I. Filaments with One-Sided Inputs," *Journal of Theoretical Biology*, Vol. 18, Issue 3, pp. 280-299, March 1968.
- [16] A. Lindenmayer: "Mathematical Models for Cellular Interactions in Development II. Simple and Branching Filaments with Two-Sided Inputs," *Journal of Theoretical Biology*, Vol. 18, Issue 3, pp. 300-315, March 1968.
- [17] P. Prusinkiewicz and A. Lindenmayer: *The Algorithmic Beauty of Plants*, Springer-Verlag, New York, August 1990.
- [18] P. Prusinkiewicz, M. Hammel, J. Hanan, and R. Mech: "L-Systems: From the Theory to Visual Models of Plants," in *Proceedings of the 2nd CSIRO Symposium on Computational Challenges in Life Sciences*, 1996.
- [19] Visual Studio: [www.microsoft.com/visualstudio/jpn/](http://www.microsoft.com/visualstudio/jpn/)
- [20] OpenGL: [www.opengl.org/](http://www.opengl.org/)
- [21] OpenCV: [sourceforge.net/projects/opencvlibrary/files/opencv-win/2.3/](http://sourceforge.net/projects/opencvlibrary/files/opencv-win/2.3/)