Automatic rule generation for procedural modeling of a sketched tree using genetic programming and particle swarm optimization

Kaveh Hassani
kaveh.hassani@uottawa.ca
University of Ottawa

Won-Sook Lee
wslee@uottawa.ca
University of Ottawa

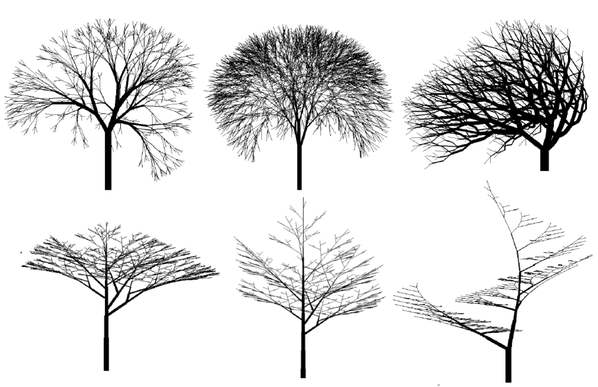
Tree modeling is the process of generating realistic 3D models of trees within virtual environments. These models are widely employed in computer animation, game design, and botanology. To address this demand, several approaches such as image-based modeling and procedural modeling have been widely explored. L-system is the most exploited procedural modeling paradigm in this area. It employs a few rules to model the growth of a tree. It has been shown that using high order L-systems leads to realistic 3D tree models. In many cases, only a sketch of a tree is drawn and the rules are not available. Defining a set of rules to reproduce the drawn tree is a tedious task. In this research, a combination of genetic programming and particle swarm optimization is proposed to automatically generate a set of rules from a sketch to model the corresponding tree.

Procedural Tree Modeling

- A set of rules to determine the mathematical model of the tree growth.
- Advantages:
  - Memory efficiency.
  - Dynamism (i.e. growth process).
  - Scalability (i.e. growth level).
  - Portability.
  - Generating identical trees while preserving the general aspects.

L-Systems

- Mathematical theory of plant development.
- The central concept is rewriting.
- Defining complex objects by successively replacing parts of a simple initial object using a set of rewriting rules.
- Similar to Chomsky's grammars, L-systems operate on strings.



Variations of L-Systems

- D0L-System
  - Deterministic and context-free.
  - Ordered triplet $G=<V, \omega, P>$.
  - V is the alphabet.
  - $\omega \in V^+$ is the axiom.
  - $P \subset V \times V^*$, is a production $(a, X) \in P : a \rightarrow X$.
- Push Down 0L-System
  - Stack data structure as memory.
- Stochastic 0L-System
  - Probabilistic model for identical trees.
  - Ordered quadruplet $G_\pi=<V, \omega, P, \pi>$.
  - Probability distribution $\pi : P \rightarrow (0,1]$.
- D2L-System
  - Context-sensitive.
  - $P: a_l<a>a_r \rightarrow x$.
  - if a is preceded by a_l and followed by a_r, then produce x.
- Parametric L-System
  - Operates on parametric strings.
  - Ordered quadruple $G=<V, \Sigma, \omega, P>$.
  - V is the alphabet.
  - Σ is the set of formal parameters.
  - $P \subset (V \times \Sigma^*) \times C(\Sigma) \times [(V \times E(\Sigma))]^*$.
  - $C(\Sigma)$: set of logical expressions.
  - $E(\Sigma)$: set of logical expressions.
- Turtle interpretation
  - Geometric interpretation of strings.
  - Move forward a step of length d.
  - Draw a line.
  - Turn to left or right in direction a.

Particle Swarm Optimization (PSO)

- Swarm Intelligence paradigm.
- Imitates the social behavior of flock of birds.
- Reaches the global optimum within a few iterations.
- Outperforms evolutionary algorithms.

Genetic Programming (GP)

- Evolutionary computational paradigm.
- Results in a parametric model that describes the dataset.
- The model is expressed by combination of operators and operands.

Feature Extraction

- Similarity between the target tree and the tree generated by algorithm.
- Line detection algorithm.

```
Extract Features (Input: Sketch)
{
  Sketch ← Pre-Process (Sketch)
  for ( degree = 0;degree < 360; degree + = 30 )
        Features ← Detect Lines (sketch, degree)
  Return Feature
}
```

Proposed Approach

- Extracting rule-base from sketched tree.
- GP algorithm extracts an optimal set of rules.
- PSO algorithm optimizes the parameters of the extracted rules.
- Fitness is the similarity between the features of the target and generated trees.
- Results in a set of optimal rules that model the sketched tree.

```
GenerateRule (Input: Sketch)
{
 Target Features ← Extract (Sketch)
  Initialize GP population ( )
  while (Proper rule-base is not generated)
   {
    foreach (individual in GP)
     {
      Optimize parameters by PSO ( )
       Tree ← Draw (individual)
       Features← Extract (Tree)
       Fitness ← Similarity (Features, Target Features)
     }
    Perform evolutionary operations ( )
   }
  return best individual as rule-base
}
```

References

- P. Prusinkiewicz, The algorithmic beauty of plants, Springer-Verlag, 1996.
- P.Tan, G. Zeng, J. Wang, S. Kang, and L. Quan, Image-based tree modeling, ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2007, 26(3), Article No. 87, 2007.

Grand 2013, 14-16 May, Toronto, Canada