Real-Time Rendering of Seasonal Influenced Trees

Michael Braitmaier, Joachim Diepstraten, Thomas Ertl
Visualization and Interactive Systems Group
University of Stuttgart[*]

Abstract

We present an algorithm for the visualization of trees during the passing of the year, taking into account the changing and vanishing of the tree's leaves as well as their color changes. In combination with the Phong lighting algorithm the color changes are directly integrated into the lighting model. In conjunction with the use of common off the shelf hardware shader programs we aim for a performance increase allowing for fast realtime rendering of trees. In contrast to earlier works we are able to rely on a more theoretical foundation of biological and chemical insights, taking into account botanic research results about biochemical reactions influencing the leaves' pigments during the seasons. This permits us to make our algorithm as realistic as possible regarding the available information and current research results in both biology and chemistry.

1. Introduction

Visualization of organic objects and natural phenomenona is still very challenging. Especially if their complexity raise problems concerning their visualization and animation. This is amongst other things greatly obvious with trees. On the other hand a large group of applications require the proper presentation of trees of different genus. Additionally they should be animated and influenced by their environment.

Early methods of displaying trees such as fractal methods and billboards could not satisfy the need for good quality visualization including environmental influences. These methods especially lacked seasonal changes. This of course was due to the fact that the complexity of trees used up the existing performance on the hardware available at these times. With oncoming and available high performance graphics hardware the limitation that applied to the first works in this area can be overcome and even allows more realistic looking trees, while still keeping the rendering of the scenes at interactive frame rates.

Furthermore not much was known on seasonal changes of trees and how they are influenced by their environment. With ongoing progress in biology and chemistry the foundations are given to realize a realistic looking seasonal change of a tree, without the need to only rely on experimental and observation data.

This paper first provides an overview of related work and addresses how this work differs from similar work already published. This is followed by a short overview of the tree model as proposed by Weber and Penn and the visualization algorithm realized in our application. Section 4 goes into detail on how the theory on shape-changing leaves as well as the color changing of leaves can be explained. The determination of the season dependent starting point of changes taking place within leaves concludes this section. Having presented the basic theory of our work Section 5 outlines the aspects of realizing the theoretical facets in our application and is finally followed by the presentation of our results in Section 6. The paper is summarized by the conclusion and possible future work to be done in Section 7.

2. Related Work

Much research has already been done in the field of visualizing trees and plants. First realistic-looking approaches introduced by Lindenmayer [7, 8] and furthermore in Lindenmayer and Prusinkiewicz [19] build the foundation of complex tree visualization. Recent papers also deal with ecosystem simulation [2], environmental sensitive structures [5, 17, 20] and many algorithms dedicated to the rendering of forests [11, 12, 13, 14, 15]. In contrast of using this common model introduced by Lindenmayer, we decided to use a different one that was first presented by Weber and Penn [23]. Based upon this approach we realized our seasoning algorithm which differs in crucial aspects from earlier work done by Mochizuki et al. [16]. We take the temperature as triggering factor for seasonal changes in contrast to Mochizuki et al.'s [16] approach utilizing sunlight. How-

---

[*] Universität Stuttgart, VIS, Universitätsstraße 38, 70569 Stuttgart, Germany; E-mail: `Mbraitmaier@gmx.de, {Joachim.Diepstraten | Thomas.Ertl}@informatik.uni-stuttgart.de`.

ever the mutual dependency between sunlight and temperature can not be denied.

The foundations of our work relies on biological facts, by simulating concentration changes of leaf pigments based on biochemical reactions in contrast to the observational and experimental approach done by Mochizuki et al. Thereby we rely on the work of Klapötke [6] with respect to the pigments that are important for the coloring of leaves.

In combination with the Arrhenius equation – proposed by the Swedish physical chemist Svandte Arrhenius in 1887 – we want to create realistic pigment concentration changes over time. Together with the Phong lighting model as described in Foley et al. [4], we achieve a realistic lighting of leaves. As pigments concentration occur only in a few layers within leaves (see Figure 2) only consideration has to be taken to light rays traveling through the first subepidermal layer. We therefore refer to the work of Shirley [22] about Fresnel coefficients and the work of Baranoski and Rokne [1] for calculating the lighting calculation inside the leaf. However Baranoski and Rokne [1] use a raytracing algorithm to trace rays through the different layers of a leaf by utilizing a BDF (Bidirectional Distribution Function). This is way too complex to currently simulate with common graphics hardware and therefore out of the scope of this paper.

3. Tree visualization

The basic idea of our tree visualization includes displaying a detailed tree consisting of a trunk with several stem levels concluded by the final leaves. During the initial research two approaches shaped out to be best suited. On the one hand the approach introduced by Lindenmayer and Prusinkiewicz [19] using a grammar-based model, the so called Lindenmayer systems and on the other hand the model introduced and presented by Weber and Penn [23]. In either model it should be possible to integrate dynamic behavior as caused by environmental influences such as wind or gravity. Furthermore it should be intuitive, to ease the creation of new tree descriptions. Finally the amount of steps for transforming a tree description into its final visualization and the extension of the tree description are additional key requirements. After balancing the two models against each other, we finally came to the conclusion that the model of Weber and Penn best fits our requirements.

Even though their model does not represent complete botanical correctness, it still allows for a realistic appearance of a tree. It provides a tree description in terms of its geometry. Therefore the tree is divided into levels based on the branching of the tree. The stems are subdivided into several frustums. By rotating them relative to each other a natural-looking curving of stems can be achieved. The amount of frustums in turn is given through the tree's description.

For each level the model specifies characteristic values such as branching angle, length of a twig or tapering. Through the combination of these characteristic values in conjunction with further global and basic parameters a tree structure is created through applying an recursive algorithm on the input parameter data. The structure of such a tree definition is shown in Figure 1.

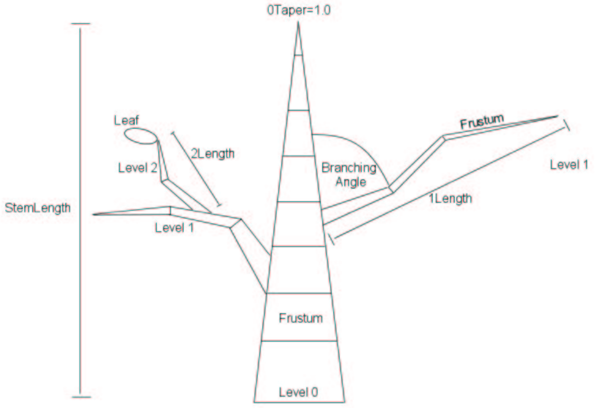

Figure 1. Simplified schematics of the structure of a tree as defined by the tree specification

The tree starts out with the trunk level or also denoted as parameter level zero. It is followed by one to three twig levels, each defined by its relative length and its branching angle towards its parent stem. Furthermore tapering and curving factors describe the shape of the twigs of each individual stem level. The curving of a twig is guaranteed by dividing a twig into several frustums. By rotating each frustum, the twig appears to be curved. Its curvature is defined through a set of curving parameters that allow a normal arc-like or even s-like twig shapes.

4. Seasoning

The seasoning algorithm should perform a continuous change of leaves during the seasons of the year, that includes color and appearance as well as the size of a leaf. To achieve these goals the algorithm is divided into two parts, namely the shape-changing and the color-changing part. Basically one only needs to consider changes of the tree's leaves in the two seasons of spring and autumn. In summer their appearance does not change considering their seasonal influences. In winter a broad-leafed tree normally shows no leaves at all.

4.1. Shape-changing algorithm

To create the effect of growing leaves it is obvious to utilize a scaling operation. Therefore the leaves are scaled from an initial value near zero to its original size given in its definition file. Currently we grow all leaves at the same rate, but the architecture of the dynamic algorithm provides starting points for different celerity of growth resembling a more natural behavior.

The simulation of falling leaves in autumn is not actually animated, but instead leaves are just relocated from their tree position to a position on the ground. The reason for keeping these modification so simple is to keep complexity of the calculations low, so the frame rate is not harmed too much, as the introduction of dynamic changes on the tree can be quite time consuming.

4.2. Algorithm for color determination

To create the effect of changing color of the tree's leaves a decision was made to come up with a more biochemical approach. In detail we first identified the significant elements responsible for the leaf color. The color of a leaf is mainly caused by the pigments it contains, more precisely the light absorption behavior of these pigments. The influence of the chlorophyll and carotene pigments is well known today [10].

But there are also other pigments that affect the leaves' color especially in autumn. To restrict the complexity of the calculations for the color of a leaf, we decided to concentrate on the most important pigments, these are the previously mentioned chlorophyll, carotene, anthocyanin, and tannin. Chlorophyll is responsible for the greenish color, while carotene creates a yellow-orange color. The anthocyanin pigments mainly cause a red appearance while finally the tannin is the base pigment existent all over the year and it is responsible for the brown base color (Klapötke [6]).

The basic idea of our algorithm is to determine the amount of light reaching the palisade cell layer of a leaf, determine the reflected light by considering the absorption behavior of the pigments and thereby determine the color of the leaf.

The determination of the resulting color for a leaf would be done by subtracting the absorption spectra of the pigments from the spectra of the incident light, thereby determining a new spectra describing the resulting color. Concerning the required data for each pigment a function is needed that describes the behavior of that pigment regarding the wavelength segment from ultra-violet (approx. 390nm) to infra-red (approx.750nm). By integrating over these functions and subtracting them from the integral of the function of the sunlight spectra, a new integral function is generated. This new function denoting the integral of the resulting color's function only needs to be derived to obtain the function describing the leaf's color spectra.

Unfortunately this approach failed due to lack of data regarding pigment absorption spectra. Though there are plenty of diagrams especially for chlorophyll and carotene, these diagrams are insufficient to generate a function describing the spectra of the pigments. Besides there were little to no absorption diagrams or absorption value tables at the different wavelengths available regarding anthocyanins and tannin. Extraction of absorption values out of the diagrams to create the required functions was not an option, because the accuracy of these diagrams is unknown and the extraction possibilities for absorption data could not be guaranteed to be exact.

Due to the lack of data, the approach to use had to be simplified. It now uses information that is accessible and still generates an autumn coloring, similar or equal to the one that could be generated with the mentioned accurate method. The wavelength at which the pigments have their maximum light absorption, is the only data that could be gathered for all pigments, respectively chlorophyll [10], carotene [21], anthocyanin [3] and tannin [18]. Therefore it is used as the base of our calculations. Additionally the association of the wavelength of maximum absorption to each pigment turned out to be an advantage, as the wavelength itself could be converted into a RGB-color value. Based on this RGB-value we try to determine the remaining light color after the sunlight passed such a pigment. This is done by computing the complementary color of the previous calculated light color derived from the absorption wavelength. By doing this reflection calculation for each pigment and combining these reflection values we thereby get the final leaf reflection color. This color is then used as the diffuse component for the Phong lighting of the leaves.

Due to the fact that leaves pigments are part of the palisade cell layer of a leaf, the aspect of incident light entering this sublayer and being reflected and therefore leaving the leaf at its upper epidermal layer has to be considered (see Figure 2). The approach used for this reflection is derived from the work of Baranoski and Rokne [1]. They use a BDF for reflection calculation of leaves taking into account the leaf's sublayer and furthermore uses basic optical physics described by Shirley [22]. The Fresnel reflection coefficient is utilized at the layer border between the palisade cell and spongy cell layer to calculate the amount of light being reflected back to the upper side of the leaf. We do not consider Fresnel reflection for the light at the epidermal layer to the palisade cell layer and backwards as the observable difference in the quality of the visualization is ne-

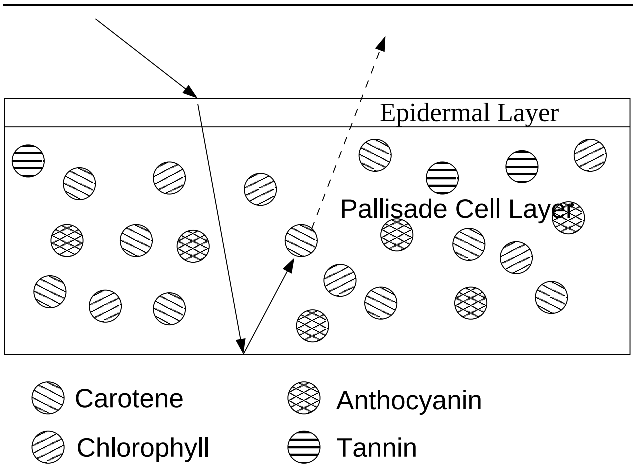

Figure 2. Concept of color determination of a leaf considering effect of pigments' absorption

glectable and stands in no relation to the required algorithmic effort in the pixelshader program.

With the decrease of chlorophyll the leaf's color is mainly determined by the color produced by the carotene pigment which leads to a change from green to a more yellow/orange appearance. As the decomposition of chlorophyll and its accompanying decrease of chlorophyll concentration also increases sugar concentration in the leaf. As described in Klapötke [6] the production of anthocyanins is increased as well which changes the yellow-orange color to a more reddish one. As time moves on, the anthocyanins are also decomposed, leaving only the tannins as the last pigments in the leaf.

The saturation of the different autumn colors is dependent on the tree genus which correlates with the maximum concentrations of chlorophyll, carotene and tannin that can occur during the seasons. Based on this concentration the amount of anthocyanins that can be produced in the leaves' cells is limited. To consider this behavior in the model, the tree definition contains four parameters specifying their maximum concentration.

4.3. Determining the start of seasonal changes

Besides the color calculation itself, its starting point when color and shape changes occur still has to be determined. Basically there are three possible approaches. The first one simply relies on a beginning and an end for the seasons, while the second one utilizes bio-chemical reactions within the leaf. The third approach considers experimental tree genus specific threshold temperatures compared to a given weather database, containing the average temperature for each month. As mentioned in Klapötke [6] color changes of a leave are based on a decrease of the chlorophyll concentration in the leaf. Though chlorophyll is decomposed during all seasons of the year, the chlorophyll synthesis produces more chlorophyll than decomposed by oxygen radicals. Due to this overproduction of chlorophyll the chlorophyll pigments overlap with other pigment – carotene and tannin.

When the production of chlorophyll through the chlorophyll synthesis begins to decrease the rate of decomposition of chlorophyll is higher than the rate of chlorophyll synthesis and therefore the concentration of chlorophyll is decreasing. To simulate the intensity of the photosynthesis and chlorophyll synthesis reaction the idea is to utilize the Arrhenius equation which determines the reaction rate of a chemical reaction based on its environmental temperature, its activation energy and a reaction typical constant value.

Along with the falling of the environment temperature, the reaction rates are affected and this way the seasonal color changes are triggered. Due to the lack of activation energy values for the Arrhenius equation this approach was not feasible. Therefore we decided to keep the concept of the temperature being the initiating factor in contrast to other work [16] which selected the incident sunlight as the main factor. We believe that there is a strong coupling between incident sunlight and temperature so the use of temperatures as initiating event seems justified. A threshold temperature can be defined in the tree specification, allowing every tree to have its own genus specific temperature thresholds for each season. A combination of the temperature threshold and a weather definition file, denoting monthly average temperatures allow also the simulation of trees in different climes.

However this concept does not account for the behavior of a decrease in the pigments that would have been considered by the use of the Arrhenius equation. Therefore we had to use a natural looking decreasing behavior this can be found in a exponential function.

To simulate the decay of chlorophyll pigments a simple inverse exponential function was used while the growth of the anthocyanin pigment concentration utilizes an exponential function of the form:

$$anthocyanin(x) = (e^x - 1) \cdot 2 \tag{1}$$

By subtracting the value one from the exponential value we ensure an initial concentration value of anthocyanin of zero. The multiplication is an experimental determined value for increasing the steepness of the function to achieve the change from green leaves to yellow-orange followed by the red color as it is described by [6].

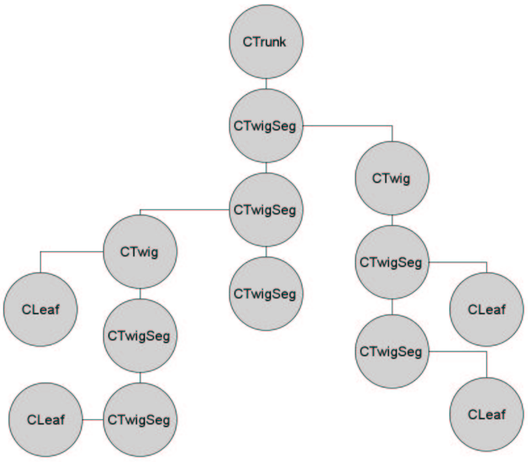

Figure 3. Example of a tree's structure within the scenegraph-tree of the intermediate representation.

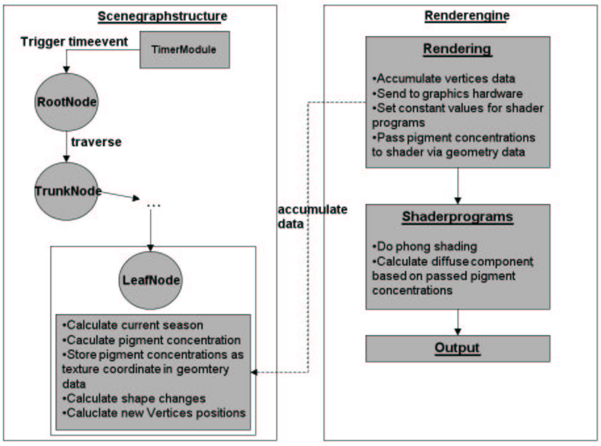

Figure 4. Steps of the realization of the seasoning algorithm containing exchanged data between the steps.

5. Realization

The application was separated in implementation packages, namely the FileLoader, the intermediate representation package InterMed and the main package TreeVis, containing the main application loop and the 3D-API-specific wrapping class. The FileLoader's main task is to load the specification file and store the values extracted from the file in its own data containers. These data containers are then used to build a scenegraph-like structure representing the tree. The nodes of the graph represent the trunk, twigs, leaves and the required frustums (see Section 3).This scenegraph structure is inspired by previous work of Lintermann and Deussen [9]. They used a similar approach to implement a interactive rendering of plants. To implement our intermediate structure we were inspired by this node-based approach. This intermediate structure is also the basis for applying dynamic changes based upon physics or environmental influences. Figure 3 shows an example of such a scenegraph-tree.

Each CTwigSeg-node contains a geometry-segment while the other nodes just contain matrices to allow for easy, relative positioning of the tree's segments. Therefore transformation of the tree's geometry into world coordinates requires full accumulation of all matrices within the tree proceeding. This results in a full scenegraph traversal for every frame. A separate renderlist has been implemented to allow for precomputation of the vertex position in world coordinates. Unfortunately this method only results in a speedup for non-dynamic behavior of the tree. To realize the seasoning algorithm we had to combine the dynamic shape-changing of the tree's leaves and the coloring model as proposed in section 4.2. The shape-changing of the leaves is basically done by applying new transformation matrices and then recompute the world coordinates of the vertices by traversing the scenegraph structure.

As basic lighting model a standard Phong lighting [4] was implemented using a combination of vertex and pixel shaders. We incorporate the seasoning coloring effect into the diffuse part of the Phong lighting model. The standard determination of the leaf color is replaced by our new seasoning calculation to get a season dependent leaf color.

To allow each leaf to start with its color-changing cycle at an independent point of time, we have to compute pigment concentration for each leaf every frame. Accordingly we have to pass these calculated pigment concentrations to our shader programs. This is done by attaching the concentration values to the geometry data of the leaf which requires a recomputation of the geometry data of each leaf at each frame at least during the phase of color changes. Figure 4 shows the basic steps of the algorithm with the data flow between the steps.

6. Results

We want to present some of our results we achieved with our implementation. So far the program is designed to support four different operation modes.

- static rendering with standard T&L
- static rendering of a tree with per-pixel Phong lighting

- dynamic rendering of a tree with per-pixel Phong lighting and season-dependent coloring
- dynamic rendering of a tree with per-pixel season-dependent coloring and leaf-shape changes

We tested our tree visualization program on different computer configurations. The test procedure includes rendering with the static tree visualization approach as well as rendering with the dynamic visualization approach. This should give a good overview on how performance is influenced by the seasonal changes. Therefore our implementation was tested with a ATI Radeon 9700 Pro, Radeon 9800 Pro and the GeForce FX 5950 graphics card. Furthermore computers with different processors were selected to make differences in CPU performance obvious. The following table shows the results we achieved showing a Quaking Aspen tree with approximately 56.000 polygons. Admittedly the application in combination with the Direct3D runtime was tested in the debug environment. Once compiled and executed in a release environment, a performance increase can be expected.

Computer	FPS
AMD Athlon 2200 XP+ ATI Radeon 9700Pro	250
Intel Pentium 4 2.8GHz ATI Radeon 9700Pro	250
Intel Pentium 4 2.8GHz ATI Radeon 9800XT	330
Intel Pentium 4 2.8GHz GeForce FX 5950	59

Table 1. Results showing our tree with static rendering and fixed coloring of the leaves.

Computer	FPS
AMD Athlon 2200 XP+ ATI Radeon 9700Pro	11
Intel Pentium 4 2.8GHz ATI Radeon 9700Pro	18
Intel Pentium 4 2.8GHz ATI Radeon 9800XT	22
Intel Pentium 4 2.8GHz GeForce FX 5950	14

Table 2. Results showing our tree with the use of dynamic features and seasonal coloring.

The results show that the current algorithm is still strongly dependent on the CPU performance. The results of the AMD 2.2GHz and the Pentium4 2.8GHz in contrast show this clearly, especially in the configuration with the same graphics hardware. Therefore the difference of seven frames (Table 2) between the AMD Athlon 2200 XP+ and the Intel Pentium P4 2.8GHz both equipped with a ATI Radeon 9700Pro is only caused by the differences of the CPUs. This is furthermore obvious taking into account the equal frame rates in the case of rendering a static tree (see Table 1) because rendering in this case consists of simple geometry rendering in combination with pixel shaders and no transformations at all. The problem can be addressed by reducing the complexity of the internal scenegraph structure, which at the moment comprises approx. 12.000 nodes for the tested tree with 56.000 polygons.

Furthermore it is recognizable that the shader unit performance on ATI graphic cards is better for our implemented algorithm.

7. Conclusion and Future Work

Based on the modified tree visualization model of Weber and Penn we are able to display a very realistic looking tree. By parsing the tree specification and converting the gathered data into an application-internal intermediate representation we create a hierarchical structure for fast access of the tree's properties. Furthermore the intermediate representation allows us to freely modify the tree's structure which practically leads to the opportunity to simulate wind or growing effects by applying simple transformation matrices which are derived from the appropriate physical laws. Through the use of vertex- and pixel shader programs we apply a Phong lighting model to the leaves and additionally a DOT3-bumpmapping algorithm to the tree's bark to account for an improved, spatial appearance. On this lighting model we base our algorithm for realistic seasonal coloring, taking into account the basic scattering effect of the light in the palisade cell layer of the leaf by the use of the Fresnel coefficient. Therefore we are able to create realistic looking seasonal coloring based upon the pigment concentration presented as exponential decay functions and the tree genus specific composition of pigments, defined through the tree description. In combination with our weather database we are also able to simulate different climes and environments, based on temperature, rainfall and sunlight. This coloring in combination with our shape-changing algorithm provides a realistic looking year-cycle of a tree. Yet we are not able to compete with advanced static tree visualization algorithms as our implementation still lacks several performance optimizing algorithms such as point-based rendering or level of detail. Also our intermediate representation still offers room for performance optimizations as the scenegraph structure is still too complex especially if dynamic features are needed. These topics will likely be addressed by future work.

Acknowledgement

We would like to thank the anonymous reviewers for their useful comments for improving this paper. This project was partially founded by the Deutsche Forschungsgemeinschaft within the Center of excellence 627 “World Models for Mobile Context-Based Systems”

References

[1] G. Baranoski and J. Rokne. An algorithmic bdf model for plant tissue. In *Computer Graphics*, InGraphics Forum Proceedings, Eurographics ’97, March 1997. ACM Siggraph ’95 Conference Proceedings.

[2] O. Deussen, P. Hanrahan, B. Lintermann, R. Měch, M. Pharr, and P. Prusinkiewicz. Realistic modeling and rendering of plant ecosystems. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 275–286. ACM Press, 1998.

[3] D. O. Fishman. Photoinduced electron transfer and spectral analysis of anthocyanin based organometallic dyes. Internet Online Resource - California State Science Fair 2003 Project Summary, April 2003. http://www.usc.edu/CSSF/History/2003/Panels/S05.html.

[4] J. Foley, A. V. Dam, S. Feiner, and J. Hughes. *Computer Graphics: Principles and Practice in C (2nd Edition)*. Addison-Wesley Pub Co, 1995.

[5] H. Jones and M. Aitken. Modelling the effect of wind blown trees. In *Proceedings of the Eurographics UK Chapter Annual Conference*, pages 245–254. EG, 1997.

[6] T. M. Klapötke. Why do (maple) leaves change color in autumn? Internet Online Resource, 1996. http://skuld.cup.uni-muenchen.de/ac/klapoetke/htm/maple1.pdf.

[7] A. Lindenmayer. Mathematical model for cellular interaction in development. *Journal of Theoretical Biology*, 18:280–315, 1968.

[8] A. Lindenmayer. Development systems without cellular interaction, their languages and grammar. *Journal of Theoretical Biology*, 30:455–484, 1971.

[9] B. Lintermann and O. Deussen. Interactive modelling of plants. *IEEE Computer Graphics and Applications*, 19, 1999.

[10] U. Lüttge, M. Kluge, and G. Bauer. *Botanik*. VCH, 1994.

[11] N. Max. Unified sun and sky illumination for shadows under trees. *CVGIP: Graphical Models and Image Processing*, 53:223–230, 1991.

[12] N. Max. Hierachical rendering of trees from precomputed multi-layer z-buffers, 1996.

[13] N. Max and K. Ohsaki. Rendering trees from precomputed z-buffer views, 1995.

[14] A. Meyer and F. Neyret. Multiscale shaders for the efficient realistic rendering of pine-trees, 2000.

[15] A. Meyer, F. Neyret, and P. Poulin. Interactive rendering of trees with shading and shadows, 2001.

[16] S. Mochizuki, D. Cai, T. Komori, H. Kimura, and R. Hori. Virtual autumn coloring system based on biological and fractal model. In *Graphics and Applications*, Proceedings of the 9th Pacific Conference on Computer Graphics and Applications PG’01, 2001.

[17] R. Měch and P. Prusinkiewicz. Visual models of plants interacting with their environment. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 397–410. ACM Press, 1996.

[18] W. Phutdhawong, S. Chowwanapoonpohn, and D.Buudhasukh. Electrocoagulation and subsequent recovery of phenolic compounds. *Analytical Sciences*, 16:1083–1084, 2000.

[19] P. Prusinkiewciz and A. Lindenmayer. *The Algorithmic Beauty of Plants*. Springer-Verlag, 1990.

[20] P. Prusinkiewicz, M. James, and R. Měch. Synthetic topiary. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 351–358. ACM Press, 1994.

[21] G. Qingping, J. Zhaoxiang, X. Xiaoyang, and D. Haiyan. Assaying of -carotene in bed-sore tissue regenration soft capsule. Proceedings of the International Workshop on Seabuckthorn, IWS ’99, 1999. IWS ’99 Conference Proceedings.

[22] P. Shirley. *Physically based lighting caluclations for computer graphics*. PhD thesis, University of Illinois, 1991.

[23] J. Weber and J. Penn. Creation and rendering of realistic trees. In *Computer Graphics*, Computer Graphics Proceedings, Annual Conference Series, pages 119–128, 1995. ACM Siggraph 95 Conference Proceedings.

Figure 5. The first image shows the tree in summer with full green leaves. The second image shows first changes in the leaves's color due to changes in the leaves's pigment concentration and finally in the third image the red coloring of the leaves becomes dominant due to increasing anthocyanin.

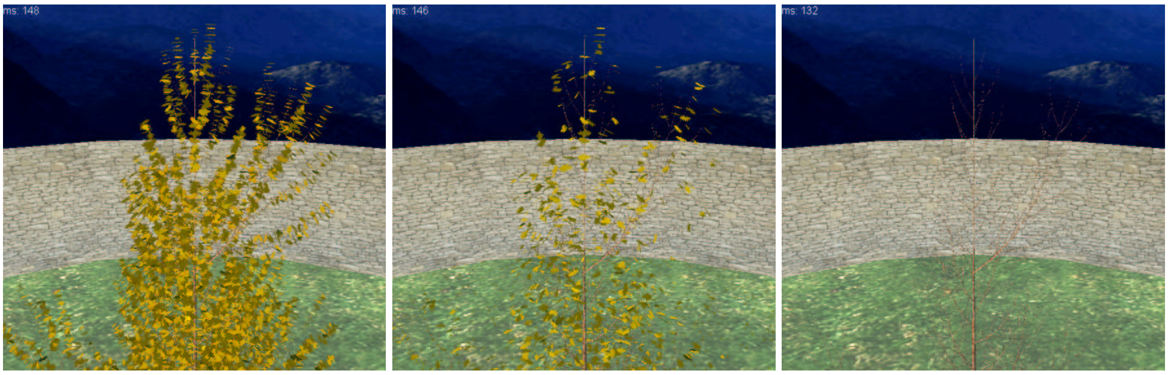

Figure 6. In the first image the leaves turn brownish due to the remaining tannin pigments. The second and third image shows the tree in the stadium of falling leaves.

Figure 7. With starting of spring the tree's leaves grow again. Starting from small to their final size, showing the tree in its full glory.