Regular Paper

A Display Method of Trees by Using Photo Images

KATSUMI TADAMURA*, KAZUFUMI KANEDA**, EIHACHIRO NAKAMAE*,
FUJIWA KATO*** and TAKAO NOGUCHI***

This paper describes a method of displaying trees that is suitable for use in visual environmental assessment for large construction projects.

It is highly desirable to be able to display various trees quickly and as realistically as possible. In our implementation, two photographed textures of a tree are mapped onto several transparent planes, and views of trees from arbitrary viewpoints are generated. The resulting images appear quite natural. For shading and shadowing, a tree contour model is employed.

The proposed method has the following merits, all very important for visual environmental assessment: (1) the tree database can be easily constructed, (2) the calculation cost of displaying groups of trees is fairly low, (3) still images with arbitrary viewpoints as well as animations can be created.

The advantages of our method are demonstrated by examples of its application to a city renewal plan and landscape.

1. Introduction

In the planning stages of large construction projects involving, for example, urban renewal or the extension of transmission lines through rural areas, the results of accurate environmental assessment are highly desirable. Various images of the complete project are helpful not only for garnering detailed opinions from residents but also for helping the designer to make decisions. Recently, several methods of displaying visual images for environmental assessment by using computer graphics techniques have been reported [1, 2, 3]. Compared with traditional methods such as hand drawing and 3D scale modeling, these methods have the following advantages: (1) various images with different shapes and colors of buildings are easily obtained; and (2) still images from any viewpoint, as well as animations, can be generated. However, improved methods of displaying natural objects, such as trees and vegetation, are desirable, especially for showing groups of trees alongside a road or on a golf course in order to select the types of trees and their locations.

To create an image with trees for environmental assessment, the following conditions should be satisfied:

1. It must be easy to construct a tree database, because a variety of species of trees are often used for such assessment.
2. It must be possible to display a group of trees quickly, because a variety of images are required.
3. It must be possible to create still images from arbitrary viewpoints, as well as animations.

Existing methods for displaying trees can be classified into three types: stochastic modeling, rule-based modeling, and impressionist modeling. For stochastic modeling, Fournier. et al. [4] employed fractal methods to display natural objects. Reeves [5] proposed particle systems for displaying grass. Demko. et al. [6] developed a calculation method for fractals, called IFS, in order to display complicated tree shapes with limited input data. Bloomenthal [7] generated very realistic tree images by using a bump mapping technique. Reeves. et al. [8] proposed a method for displaying forests by using particle systems. Oppenheimer [9] generated trees whose trunks were stochastically modeled. Viennot. et al. [10] improved upon References [7], [8], and [9] to control the final form.

Rule-based modeling, which was developed by Aono. et al. [11] and Smith [12], generates geometric models based on botanical descriptions of trees. Prusinkiewicz. et al. [13] and Reffye [14] presented models of the growth of trees. Agui. et al. [15] proposed a method that generates various tree shapes automatically.

Impressionist modeling was proposed by Gardner

*Department of Management and Information Sciences, Hiroshima Prefectural University.
**Faculty of Engineering Hiroshima University.
***Tokyo Electric Power CO., Inc.

[16]. In this approach, a tree is represented by mapping onto a quadric surface texture patterns generated by mathematical functions.

The main aim of stochastic modeling and rule-based modeling is to represent the detailed structure of each tree; that is to say, a tree plays the central role in an image. If these methods are used to generate images for environmental assessment, construction of a rich database of trees is too time-consuming. In stochastic modeling, trial and error are used to obtain the desired types and shapes of trees, and in rule-based modeling we must begin with investigations into the morphology of trees. Another problem is the time consumed in displaying many groups of trees, if these trees are represented as polygon models by using stochastic modeling or rule-based modeling.

Compared with these modeling approaches, impressionist modeling can more easily represent trees and is more efficient. However, it creates less highly detailed trees, making it somewhat difficult to recognize the tree types.

In this paper we present a new approach to modeling trees that play a relatively important, but supporting, role in an image. In our method, two textures of as tree are mapped onto several transparent planes. A tree database including various trees is relatively easy to construct, because texture images of trees can be made by digitizing tree photographs. The calculation cost of displaying a group of trees is lower than in particle systems [8] because of the use of texture mapping.

However, if the viewpoint changes, images are animated, and trees may look unnatural because each photo image contains only two-dimensional information. In order to overcome this problem, two photographs of a tree taken from directly above and from the side are mapped onto a set of transparent planes, a vertical plane, and several horizontal planes.

These planes, hereafter called image planes, are described in the next section. Shading and shadowing are the topics of Sections 3 and 4, respectively. In Section 5, the advantages of our method are demonstrated examples of its application to urban renewal planning and rural landscapes.

2. Representing a Tree by Using Two Photo Images

2.1 Setting Image Planes

It should be possible to generate images for environmental assessment from arbitrary viewpoints, such as from a seat in an automobile, from the eye level of a person walking along a road, and from the windows of a tall building. The appearance of trees and the aspect of their leaves change as the viewpoint is moved. To accommodate the viewpoint being located at any position or moving along any route, and in order to display the trees as naturally as possible, the image

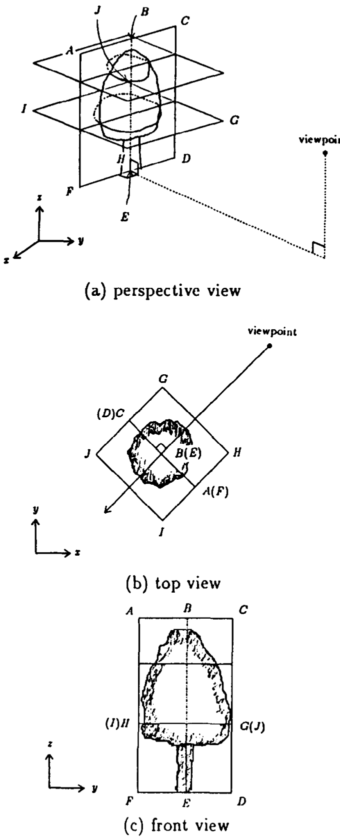


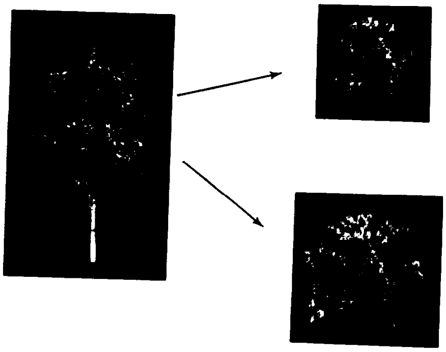
Fig. 1 Setting imaginary planes for mapping tree images.

planes are set as follows (see Fig. 1):

1. The normal of a vertical image plane *ACDF* is always set parallel to the horizontal component of the view direction; the image plane is rotated around the line *BE*, the central axis of the trunk.
2. One of the horizontal image planes (plane *GHIJ*) is set at the height of the maximum width of the tree on the vertical plane, and the others are set at appropriate heights. The number of horizontal planes depends on the branching structure and the density of leaves. Horizontal planes are rotated synchronously with the vertical plane.
3. If a reflecting or a refracting object exists, a set of image planes is set in the same manner as mentioned above; the normal of each vertical image plane is set parallel to the horizontal component of a reflected or refracted viewing line.

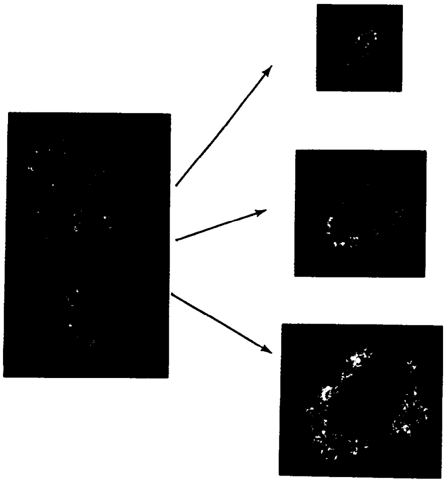
2.2 Mapping Method

The vertical texture of a tree is made by eliminating the background area from a vertical photograph; in this step, image processing techniques are employed. The texture is then mapped onto a vertical plane. The tex-



(Vertical texture) (Horizontal texture)

(a) Lush tree



(Vertical texture) (Horizontal texture)

(b) Tree with sparse leaves

Fig. 2 Textures for mapping onto image planes.

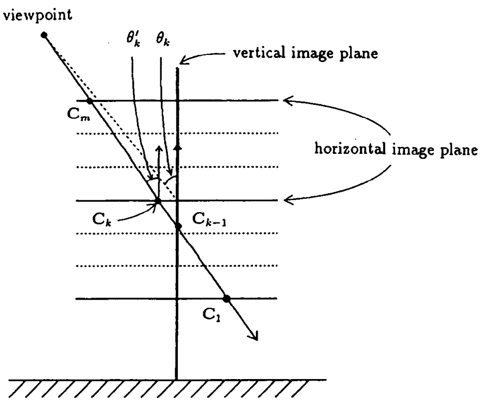
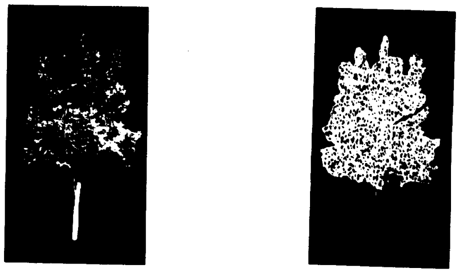


Fig. 3 Synthesizing texture color on each transparent plane.

(a) pencil type (b) ellipsoid type (c) inverse cone type

(combination of a cylinder and a cone)



(a) texture (b) small domain

Fig. 5 Small domains in texture.

ture image obtained from a horizontal photograph is mapped onto some horizontal planes, using the following modification. The diameter of the texture is adjusted to match the width of the vertical texture, and if the density of leaves is low, a group of leaves that exist near the height of each horizontal plane is suitably extracted; two examples with one vertical texture and several horizontal textures of trees are shown in Fig. 2.

When the viewing line moves in some direction, the tree should continue to appear natural and change smoothly. The color at the intersection of each viewing line with each image plane is computed in order of priority along the ray from the viewpoint, using the following equations (see Fig. 3):

$$\mathbf{B}_{k+1}=(1-t_{k+1})\mathbf{C}_{k+1}+t_{l+1}\mathbf{B}_{k},$$

$$\mathbf{B}_1=\mathbf{C}_1,$$

where $\mathbf{B}_k$ is a color (consisting of R, G, B components) synthesized at the intersection between the k-th most distant transparent plane and the viewing line, $\mathbf{C}_k$ is a shaded texture color of the k-th intersection, described in Section 3.1, and t_k is the transparency coefficient of the texture on the k-th transparent plane.

The transparency coefficient satisfies the following conditions: (1) if the viewpoint is set directly above the tree, only the textures mapped onto the horizontal planes are displayed, (2) if the viewpoint is set to the right of the tree, only the texture mapped onto the vertical plane is displayed, and (3) the coefficient value continuously changes between these two viewpoints. That is, t_k should be a parameter of the angle, θ_k', between the normal of the imaginary plane and the viewing line. However, in this paper the function $t_k = 1 - \cos\theta_k$ is used in order to save computation time, where θ_k is the angle between the normal of the image plane and the line connecting the viewpoint and the center of the plane.

When the number of intersections is m, the final synthesized tree color $\mathbf{C}$ is $\mathbf{B}_m$. When the viewpoint moves horizontally, the outer appearance of each tree does not change in this method. At first sight, this might seem likely to result in unnatural animation. However, the animations [17-20] produced by the authors demonstrate that the relations between the locations of individual trees have the greatest influence on the view; changes in the outer appearance of individual trees are practically irrelevant unless the viewer is closely watching an individual tree.

3. Shading

3.1 Shading of Trees

The shape of a tree for shading is approximated by a polyhedron surrounding it, a so-called contour model. It is well known that shapes of trees can be roughly classified into the three types shown in Fig. 4. Using these models, the shading effect of each texture element is calculated as follows.

1. The texture of a tree is divided into small domains as shown in Fig. 5, and a unit normal vector $\mathbf{N}_{k,i}$ ($i = 1, 2, \ldots, n$) is then assigned to each of them, where n is the number of small domains and k is an image plane number. The size of the small domains depends on the size of the pattern of leaves on the texture image. The direction of the assigned normal represents that of the leaves approximated as a small plane; the method of generating the normal is described in Section 3.2 below.
2. The point P for shading is obtained by calculating the point at which the viewing line meets the contour model. As the direction of the x-axis is outward with respect to the direction of the assigned normal $\mathbf{N}_{k,i}$ (see Section 3.2), the direction of $\mathbf{N}_{k,i}$ has to be moved; the rotating angle ψ between the x-z plane and normal $\mathbf{N}$ of the contour model at P is calculated and used to adjust $\mathbf{N}_{k,i}$ to $\mathbf{N}_{k,i}'$ (see Fig. 6).
3. We assume that $\mathbf{N}_{k,i}$ is the normal on a small domain of i, which includes the intersection of the viewing line and the image plane k, and that the $\mathbf{N}_{k,i}'$ is a vector at a point rotated at an angle of ψ degrees from $\mathbf{N}_{k,i}$ about the center of the tree. $\mathbf{N}_{k,i}'$ is assumed to be the normal of the small domain including point P'. The shaded color, $\mathbf{C}_k$, at point P is calculated by the following equation according to Lambert's law:

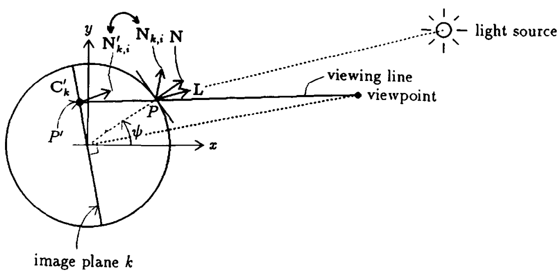


Fig. 6 Shading model.

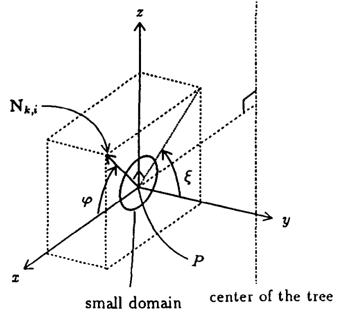


Fig. 7 Unit normal assigned to a small domain.

$$\mathbf{C}_k = \begin{cases} (1-\alpha)(\mathbf{L}\cdot\mathbf{N}_{k,i}')\mathbf{C}_k' + \alpha\mathbf{C}_k' & (\text{if } \mathbf{L}\cdot\mathbf{N}_{k,i}' > 0) \\ \alpha\mathbf{C}_k' & (\text{otherwise,}) \end{cases}$$

where $\mathbf{L}$ is a unit vector at point P toward the light source, $\alpha (0 \le \alpha \le 1)$ is the ambient light factor, and $\mathbf{C}_k'$ is the texture color at P' on plane k.

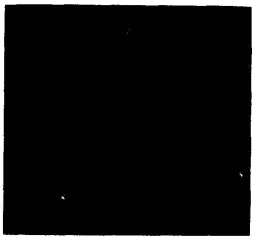
3.2 Assigning Unit Normals to Small Domains

When a horizontal axis x including point P and the center of the tree is chosen, $\mathbf{N}_{k,i}$ represents the normal of the leaf (see Fig. 7). Although leaves face in various directions, usually most face outward (in the direction of the x-axis). Let us assume the following normal distribution for the normals of leaves:

$$\begin{cases} (N_{k,i})_x = \cos\varphi \\ (N_{k,i})_y = \sin\varphi\cos\xi \\ (N_{k,i})_z = \sin\varphi\sin\xi, \end{cases}$$

$$\varphi = \pi a\,|\,randn\,| \left(0 \le \varphi \le \frac{\pi}{2} \text{ is used,}\right)$$

(a) $a = \frac{1}{6}$



(b) $a = \frac{5}{3}$

Fig. 8 Distribution of unit normal assigned to small domains.

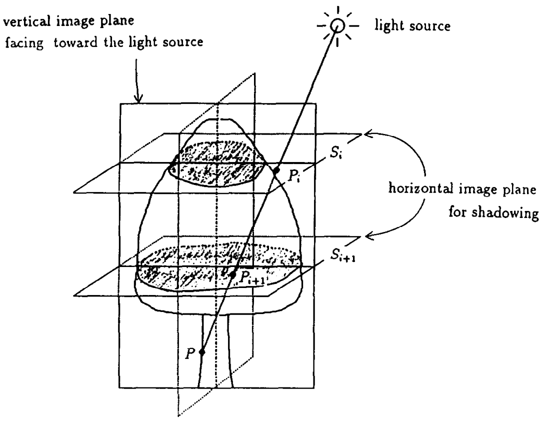
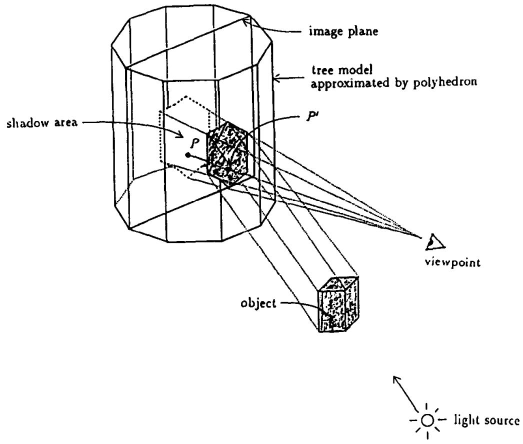
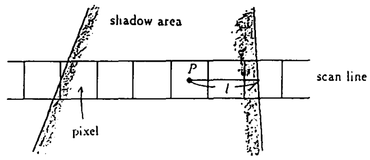


Fig. 9 Shadow cast by a tree onto its own trunk.



(a) determination of shadow area



(b) gradation of shadow boundary

Fig. 10 Shadow cast by an object onto a lush tree.

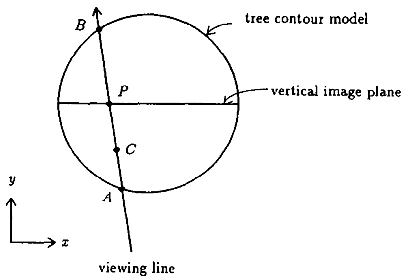


Fig. 11 Determination of the position of leaves on a tree with sparse leaves.

$$\xi = 2\pi\,|\,rand\,|,$$

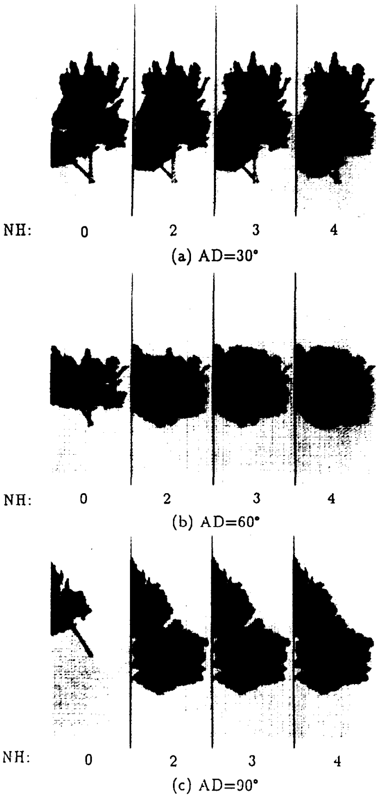
where *randn* is anormal random number (the average is 0 and the variance is 1), *rand* ($0 \leq rand < 1$) is a uniform random number, and $a (a > 0)$ is a parameter that varies the distribution of the directions of small domains.

$\mathbf{N}_{k,i}$ is distributed over a hemisphere with a radius of unit length. The smaller the parameter a, the more closely that unit normals are distributed around the x axis (see Fig. 8). That is, the parameter is set to a small value for trees whose leaves regularly look outward.

4. Shadowing

For shadowing, a set of image planes is generated in the same way as for the viewpoint except that the normal of each vertical image plane is set parallel to the horizontal component of the direction to the light source. The same texture images used for the viewpoint are used for shadowing, even though no texture image is displayed.

Shadowing is executed for the following four cases: (1) shadows cast by trees onto their own trunks, (2) shadows cast by trees onto objects, (3) shadows cast by objects onto trees, and (4) shadows cast by trees onto



NH: number of horizontal transparent planes
AD: angle of declination

Fig. 12 Pencil-type tree.

other trees.

4.1 Shadows Cast by Trees onto Their Own Trunks

In most tree images, leaves cast shadows onto their tree's trunk (see Fig. 9). Every point P on a trunk is checked to determine whether or not any texture color on the horizontal image planes used for shadowing (such as S_i and S_{i+1}) exists at the intersections (such as P_i and P_{i+1}) of the line connecting point P and a light source with the planes (such as S_i and S_{i+1}). If at least one texture color exists, the point lies in shadow.

4.2 Shadows Cast by Trees onto Objects

First, a check is made to determine whether the calculation point is inside the shadow volume generated by the image planes used for shadowing. Only if the point lies in the shadow volume is the test executed. the method is the same as that described in Section 4.1, but in this case both vertical and horizontal image planes are tested.

4.3 Shadows Cast by Objects onto Trees

4.3.1 Lush Trees

Shadows cast on a lush tree are calculated by using the tree model approximated to a polyhedron for shading. That is, shadows are determined by testing whether or not the intersection P' of the tree model and the viewing line lies in the shadow volumes of other objects (see Fig. 10(a)).

The shadow boundary may be too sharp and look unnatural. In order to address this problem, the boundary of a shadow is softened as follows. First, the distance l from the center point P of a pixel to the nearest shadow boundary is calculated (see Fig. 10(b)). Next, a random number between 0 and $max(0, 1-l/l_{max})$ is generated, where l_{max} is a parameter defining the length of a penumbra; that is, the region beyond l_{max} is a shaded area. Then the color at the calculation point is obtained by multiplying the shaded color by the random number. Random numbers are assigned beforehand to the small domains in the same manner as the normals for shading, to prevent animated images from flickering.

4.3.2 Trees with Sparse Leaves

In order to determine whether the leaves of a tree with sparse leaves lie in shadow, the three-dimensional position of the leaves must be set. In our method, the positions of leaves are determined by using the following probabilistic method, since the tree textures have only two-dimensional information.

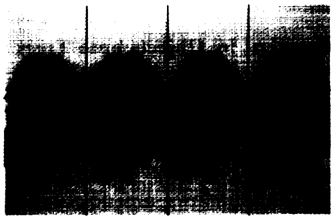
Let P be a point on the texture of an image plane to be processed (see Fig. 11). The leaf at point P lies on the line AB, where points A and B are the intersections of the viewing line and the tree model. The position of the leaf, C, on the line AB is determined by using a random number that is assigned beforehand to the small domains in the same manner as that described in Section 4.3.1, in order to prevent animated images from flickering. If point C lies in any shadow volume, the texture at point P is processed in the same way as if it were in a shadow area.

4.4 Shadows Cast by Trees onto Other Trees

Shadows cast by other trees are calculated as follows. First, the leaf positions are determined by the method described in Section 4.3. Then, the shadow test of Section 4.2 is employed.

5. Examples

First, the number of horizontal image planes appropriate for displaying trees and shading and shadow effect is examined by using the three tree types illustrated in Fig. 4. Then, some examples applied to a parking lot, street trees, and rural landscapes demonstrate the advantages of the proposed method.

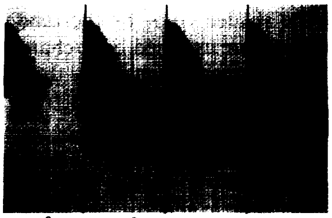


NH: 0 2 3 4

(a) AD=30°

NH: 0 2 3 4

(b) AD=60°



NH: 0 2 3 4

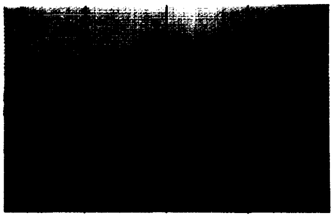
(c) AD=90°

NH: number of horizontal transparent planes
AD: angle of declination

Fig. 13 Ellipsoid-type tree.

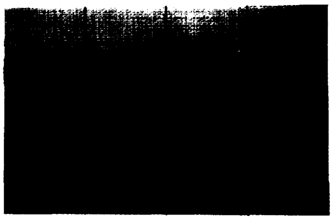
NH: 0 3 4 5

(a) AD=30°



NH: 0 3 4 5

(c) AD=90°



NH: 0 3 4 5

(b) AD=60°

NH: number of horizontal transparent planes
AD: angle of declination

Fig. 14 Inverse-cone-type tree.

The appropriate number of horizontal image planes depends on the shape of the tree and the lushness of the leaves. In Fig. 12, a lush, pencil-type tree is displayed, and the number of horizontal image planes is varied. The angles of declination in Fig. 12(a), (b), and (c), are set to 30, 60, and 90 degrees, respectively. In Fig. 13 and 14, a lush, ellipsoid-type tree and an inverse cone type tree with sparse leaves are examined. Obviously, trees that have no horizontal plane disappear when the angle of declination is set to 90 degrees. These figures show that the greater the number of horizontal planes, the more natural the tree appears, but that three or four horizontal planes are generally adequate.

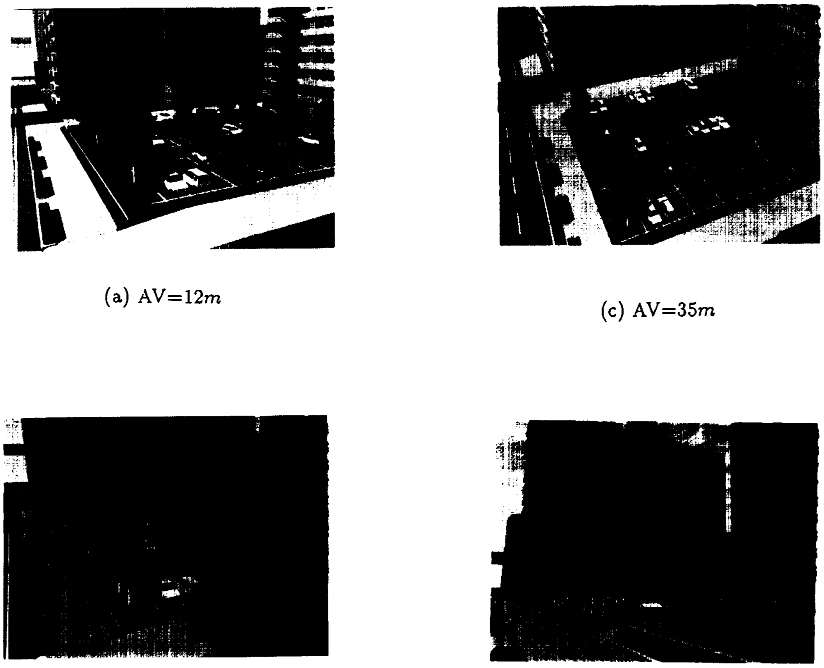
Figure 15 shows examples in which an object, a wall in this case, casts a shadow onto a lush tree (Fig. 15(a)) and a tree with sparse leaves (Fig. 15(b)). On the left side of each figure, no special processing for shading and shadow effect is done. The method discussed in Section 4.3 is employed on the right side of each figure, and parameter a, which varies the distribution of small domains, is set to 1.0. Parameter l_{max}, which controls the width of the shadow boundary, is set to 10 pixels in the

(a) AD=0°

(b) AD=30°

AD: angle of declination

Fig. 15 Shadows cast by an object onto a tree.



(a) AV=12m

(c) AV=35m

(b) AV=12m

(d) AV=2m

AV: altitude of viewpoint from ground

Fig. 16 Application to the design of a parking lot.

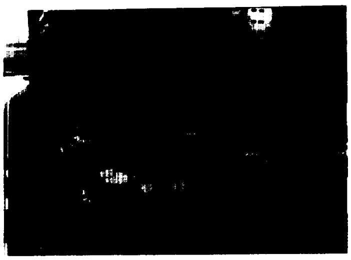


Fig. 17 Parking lot in the evening.

Fig. 18 Trees along a street.

Fig. 19 A golf course.

lush tree image, and it is assumed that the front faces in the vertical texture of a tree with sparse leaves have a higher probability of being visible than the back faces.

Figure 16 shows an application to the landscaping and architecture of a parking lot. In Fig. 16(a) and (b), different types of trees are used, and in both cases the altitude of the viewpoint is fixed at 12 meters, while in Fig. 16(c) and (d), it descends from 35 meters to ground level (2 meters). Figure 17 shows a parking lot in the evening.

Figure 18 shows a landscape design of trees along a street with a skylight effect and sunlight (the sun has a finite diameter). In this figure tree shadows are cast onto the road. The CPU time for this example was 38 minutes and 43 seconds (on 3.2 MIPS machine). The ratio of CPU time for figures with and without trees is 1.472 (for ten trees).

Figure 19 shows an application to the design of a golf course.

The proposed method was applied in animations [17–19] for visual environmental assessment, and to animation [20] of a rural landscape.

6. Summary

We have described a method of displaying trees for visual environmental assessment.

In the method, two textures digitized from two photographs taken from the right side and from above the tree are mapped onto a set of transparent planes, a vertical plane whose normal always faces toward the horizontal component of the direction to the viewpoint, and several horizontal planes rotated synchronously with the vertical plane. Various images can be generated at a fairly low calculation cost, because they are based on texture mapping, and still images with arbitrary viewpoints as well as animations can be created.

For shading and shadowing, the shape of a tree is approximated by a polyhedron surrounding it. Shading is executed by using normals assigned to small domains into which the texture of the tree is divided. Shadows cast onto trees, as well as tree shadows cast onto objects, look natural whether the tree is lush or has sparse leaves.

Acknowledgment

The authors wish to thank the students of the electrical machinery laboratory in Hiroshima University for their cooperation, especially Toshio Akinobu, Haruhiko Iriyama, and also to thank Professor Nishita for stimulating discussions. The data are courtesy of Osaka Municipal Government.

References

1. Maver, T. W., Purdie, C. and Stearn, D. Visual Impact Analysis—Modeling and Viewing the Natural and Built Environment, *Computer & Graphics*, **9**, 2 (1985), 117-124.
2. Nakamae, E., Harada, K., Ishizaki, T. and Nishita, T. A Montage method: The Overlaying of Computer-Generated Images onto a Background Photograph, *Computer Graphics*, **20**, 4 (1986), 207-214.
3. Kaneda, K., Kato, F., Nakamae, E., Nishita, T., Tanaka, H. and Noguchi, T. Three-Dimensional Terrain Modeling and Display for Environmental Assessment, *Computer Graphics*, **23**, 4 (1989), 207-214.
4. Fournier, A., Fussell, D. and Carpenter, L. Compuer Rendering of Stochastic Models, *Comm. ACM*, **25**, 6 (1982), 371-384.
5. Reeves, W. T. Particle Systems—A Technique for Modeling a Class of Fuzzy Objects, *Computer Graphics*, **17**, 3 (1983), 359-376.
6. Demko, S., Hodges, L. and Naylor, B. Construction of Fractal Objects with Iterated Function Systems, *Computer Graphics*, **19**, 3 (1985), 271-278.
7. Bloomenthal, J. Modeling the Mighty Maple, *Computer Graphics*, **19**, 3 (1985), 305-311.
8. Reeves, W. T. and Blau, R. Approximate and Probabilistic Algorithms for Shading and Rendering Structured Particle Systems, *Computer Graphics*, **19**, 3 (1985), 313-322.
9. Oppenheimer, P. E. Real Time Design and Animation of Fractal Plants and Trees, *Computer Graphics*, **20**, 4 (1986), 55-64.
10. Viennot, G. X., Janey, N. and Arquès, D. Combinatorial Analysis of Ramified Patterns and Computer Imagery of Trees, *Computer Graphics*, **23**, 3 (1989), 31-40.
11. Aono, M. and Kunii, T. L. Botanical Tree Image Generation, *IEEE CG & A*, **4**, 5 (1984), 10-34.
12. Smith, A. R. Plants, Fractals, and Formal Languages, *Computer Graphics*, **18**, 3 (1984), 1-10.
13. Prusinkiewicz, P., Lindenmayer, A. and Hanan, J. Developmental Models of Herbaceous Plants for Computer Imagery Purposes, *Computer Graphics*, **22**, 4 (1988), 141-150.
14. Reffye, P., Edelin, C., Françon, J., Jaeger, M. and Puech, C. Plant Models Faithful to Botanical Structure and Development, *Computer Graphics*, **22**, 4 (1988), 151-158.
15. Agui, T., Fukuda, S. and Nakajima, M. Botanical Tree Generation Method for Scenery Rendering, *Trans. IPS Japan*, **32**, 5 (1991), 618-628.
16. Gardner, G. Y. Simulation of Natural Scenes Using Textured Quadric Surfaces, *Computer Graphics*, **18**, 3 (1984), 11-20.
17. Hiroshima University. The Sky (video), SIGGRAPH 88 Animation Screening Room (1988).
18. Hiroshima University. Scenes at a Street Corner (video), SIGGRAPH 89 Animation Screening Room (1989).
19. Hiroshima University. A Passing Shower (video), SIGGRAPH 90 Film and Video Theatre (1990).
20. Hiroshima University. The Works of A Landscape Painter (video), SIGGRAPH 91 Electric theatre (1991).

(Received October 14, 1991; revised January 27, 1992)