Radiosity for Large Vegetation Scenes

Helmut Mastal and Robert F. Tobler and Werner Purgathofer

Institute of Computer Graphics, Vienna University of Technology
Karlsplatz 13/186/2, A-1040 Vienna, AUSTRIA
mastal@zid.tuwien.ac.at, {rft,wp}@cg.tuwien.ac.at

Abstract

Calculating radiosity solutions for large scenes containing multiple plants is all but impossible using the radiosity method in its original form. With the introduction of sophisticated hierarchical and clustering algorithms radiosity for vegetation scenes becomes a solvable challenge. The precomputation of the diffuse light distribution in leaf canopies of forests and other plants can be used to calculate realistic images, but also for agricultural planning purposes. This state of the art report gives an overview of the methods that can, and have been, used to calculate global illumination in vegetation scenes, including hierarchical methods, statistical methods based on simplifications, and specialized methods that have been optimized to handle scenes with a dense, non-isotropic distribution of objects such as canopies.

1. Introduction

Radiosity belongs to the global illumination methods for generating photorealistic images by computer graphics. The physical foundation of radiosity is the diffuse reflection and diffuse refraction of light on surfaces. In radiosity a perceived photon cannot be traced back to its emitter, it does not know from where it originated. Therefore every object of a scene can have an impact to the brightness and colour of a viewed surface, even those objects that are not directly visible from the viewed object, and even the viewed object itself can give a contribution to the irradiance of a viewed object. In radiosity no straight–forward solution of the rendering equations is possible to determine the radiance values of some surfaces. Rather a holistic approach is necessary that calculates all radiosities of all participating objects or surfaces. An equilibrium of light energy has to be found for the scene. This is also true for vegetation scenes that are rather complex radiosity scenes.

2. Original Radiosity

In nature most reflection phenomena are diffuse, glossy effects are also very common, but exact specular reflection is very rare, except for water surfaces not disturbed by waves. The surfaces of plants, of trees, of earth, of rocks are mainly rough and therefore diffuse reflection of light is very important in nature. Even the incoming light of the sun or the moon is only partly direct, and a non-negligible contribution of light is diffuse because of refraction in the atmosphere.

Outdoor scenes were not so often presented in the past. The number of light interactions of a scene grows with the square of the number of objects involved. For normal outdoor scenes the number of interactions exceeds computability on today's processors with naive algorithms.

Radiosity methods originated from the theory of radiative heat transfer in the 1980s [Gora 84]. Original radiosity algorithms assume the scene to be discretized into small planar elements which have constant brightness. Enforcing an energy balance at every element results in a system of linear equations of the form:

$$B_i = E_i + \rho_i \sum_j^n F_{ij} B_j \qquad (1)$$

where B_i is the radiosity, E_i is the emissivity, ρ_i is the diffuse reflectance, F_{ij} is the so called form factor (the percentage of light leaving element j that arrives at i), and n the number of elements in the scene, yielding a system of equations:

Published by the Eurographics Association, ISSN 1017-4656.

$$\begin{pmatrix} 1 & -\rho_1 F_{12} & \dots & -\rho_1 F_{1n} \\ -\rho_2 F_{21} & 1 & \dots & -\rho_2 F_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ -\rho_n F_{n1} & -\rho_n F_{n2} & \dots & 1 \end{pmatrix} \begin{pmatrix} B_1 \\ B_2 \\ \vdots \\ B_n \end{pmatrix} = \begin{pmatrix} E_1 \\ E_2 \\ \vdots \\ E_n \end{pmatrix}$$

This system can be solved using iterative algorithms such as the Gauss–Seidel method. Physically, the Gauss–Seidel method is equivalent to successively *gathering* incoming light. An alternative iteration scheme is to reverse the process by successively *shooting* light [Cohe 88]. The most expensive part of calculation is computing the form factors. The differential form factor is given by:

$$dF_{ij} = \frac{\cos\theta_i \cos\theta_j}{\pi r_{ij}^2} dA_j \tag{2}$$

The angle θ_i (or θ_j) relates the normal vector of element i (or j) to the vector joining the two elements. The form factor between two finite areas is the double integral

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos\theta_i \cos\theta_j}{\pi r_{ij}^2} dA_j dA_i \tag{3}$$

These formulae do not take into account occlusion. The differential form factors have to be accumulated only if the infinitesimal elements are mutually visible. This was achieved by the hemi–cube algorithm [Cohe 85]. Also algorithms based on ray tracing were proposed for form factor calculation [Wall 89]. The greatest drawback of these early radiosity algorithms was the $O(n^2)$ growth of the form factor matrix and in computation time.

The first method to reduce the computational costs was motivated by the method of *substructuring* used in finite element calculations. The polygons comprising the scene are discretized at two levels, the level of patches and the level of elements into which each patch is broken [Cohe 86]. In the 1990s stochasic radiosity methods (Monte Carlo radiosity) were introduced by Shirley [Shir 91]. Stochastic methods could reach a reasonable accuracy in a reasonable time for larger scenes without creating the $n * n$ matrix. They are closer to the physical nature of light (photons) and so can include all types of reflection within one algorithm.

3. Hierarchical Radiosity Algorithms

The first successful radiosity approach to render average complex scenes with arbitrarily high accuracy or very large scenes with an average accuracy in a reasonable time, was presented by Hanrahan, Salzmann and Aupperle [Hanr 91], called Rapid Hierarchical Radiosity. The hierarchical subdivision algorithm was inspired by methods developed for solving the N–body problem in the eighties. The idea behind both problem solutions was to reduce the $O(n^2)$ growth of interactions and time to $O(n)$ or almost $O(n)$ by approximating within a given precision a number of interactions with a single interaction.

Hanrahan et al. observed, that usually many form factors are computed at a much higher precision than is necessary. Therefore in the hierarchical algorithm a polygon is decomposed into a hierarchy of patches and elements and the form factor matrix is represented in a hierarchy of interactions at different levels of detail.

The procedure *Refine* represents the algorithm. It first estimates the form factors F_{pq} and F_{qp} between two elements p and q, and then either subdivides one of them and refines further recursively, or it links them and terminates the recursion:

```
Refine:
  Fpq = FormFactorEstimate (p, q)
  Fqp = FormFactorEstimate (q, p)
  if (Fpq < Feps && Fqp < Feps)
    Link (p, q)
  else
    if (Fpq > Fqp)
      if (Subdiv (q, Aeps) ) Refine (q) else Link (p, q)
    else
      if (Subdiv (p, Aeps) ) Refine (p) else Link (p, q)
```

If the form factor estimate is less than F_{eps}, then the true form factor can be approximated by the estimate, and the patches are linked in *Link* for interaction. Otherwise *Subdiv* is called to divide a patch into four subpatches building a subdivision hierarchy with a quadtree structure. *Subdiv* returns *false* if the patch area is smaller than A_{eps} to prevent infinite recursion.

Fig. 1 shows the quadtree subdivision of two perpendicular surfaces resulting from this algorithm, and the interactions at each level in the hierarchy. Distant elements interact at a high level in the hierarchy – reducing the total number of interactions –, while elements close to one another are subdivided and interact at a lower level in the hierarchy.

Fig. 2 shows the decomposition of the form factor matrix into blocks. Only two perpendicular lines as the two–dimensional equivalent of surfaces are shown. The subdivision hierarchy is represented by a binary tree in this two–dimensional scene. Each interaction shown as a labeled arc in the left part of the figure corresponds to the block with the same label in the matrix.

Upto that point occlusions between the patches were not considered in the algorithm. Intervening occluding surfaces can only decrease light transport between two patches. Thus the true form factor in the presence of occlusion is never greater than the form factor estimate described above. The effect of occlusion can be modeled as follows:

$$F = V_e * F_e \tag{4}$$

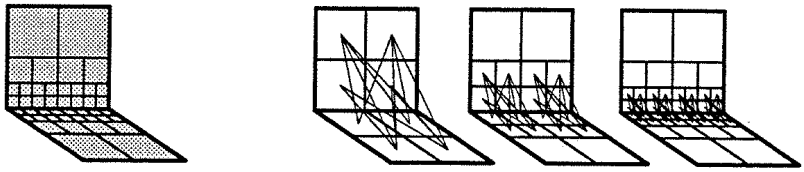

Figure 1: *Interactions between a pair of perpendicular surfaces.*

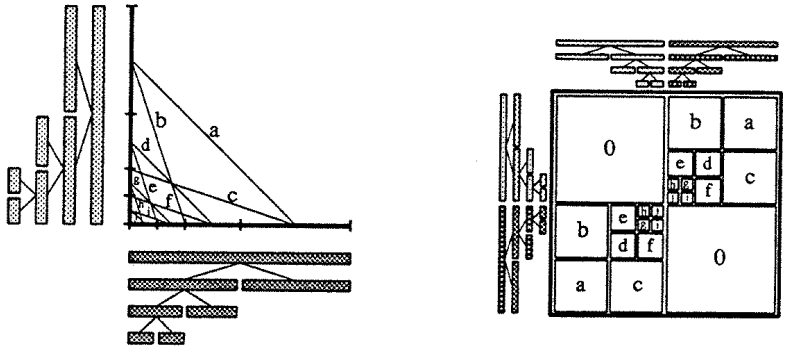

Figure 2: *Decomposition of the form factor matrix into blocks.*

where F_e is the estimated form factor without considering occlusion, and V_e is the estimated visibility. If $V_e = 1$ then the two patches are totally visible; if $V_e = 0$ then they are completely occluded; otherwise they are partially visible. Since the form factor estimation is not precise, the calculation of V_e need only be done at the same precision.

The recursive refinement procedure can exploit visibility coherence in a natural way to avoid unnecessary refinement and visibility calculations. If two subpatches become totally invisible relative to each other, then the refinement can be terminated immediately. If two subpatches become totally visible, visibility tests may be stopped, but further refinement may still occur.

After the form factors have been determined the matrix can be inverted iteratively to get a solution for the radiosities. Because of the block structure of the form factor matrix, each iteration step can be done in linear time. Gathering is performed using the classic Jacobi iteration using the following recursive procedure:

```
Gather: ForAllElements (q, p->interactions)
        Fpq = FormFactor (p, q)
        p->Bg += Fpq * p->Cd * q->B
    GatherChildren (p)
```

The average brightness of each patch is stored in *B* and its diffuse color reflectance is stored in *Cd*. The brightness is gathered in *Bg* and is computed by receiving energy from all patches *q* stored in the list of interactions of *p*. The total amount of energy received by an element is the sum of the energy received by it directly, plus the sum of all the energy received by its parent subpatches. To update the radiosities all the energy gathered is pushed down to the leaf nodes, and then pulled upwards to the root polygon. During this upward pass, the radiosity of interior subpatches is set equal to the area weighted average of its children's radiosities.

The radiosity equation can be solved by shooting instead of gathering. Also a multigridding variation of the shooting algorithm is possible, resulting in fewer expensive iterations before convergence. A final improvement to the algorithm bases the refinement of two patches on BF, the total energy transported between the patches.

4. Clustering in Hierarchical Radiosity Algorithms

In [Sill 94] Sillion introduced a new approach to hierarchical radiosity calculations for very complex scenes. By grouping together nearby surfaces to clusters for calculating their energy exchanges with other objects, these clusters can be seen as volumes, and the cost for linking can be reduced.

Energy exchange between surfaces and/or volumes can be represented in the same way by the equation:

$$B_i = E_i + \rho_i \sum_j F_{ij} B_j \qquad (5)$$

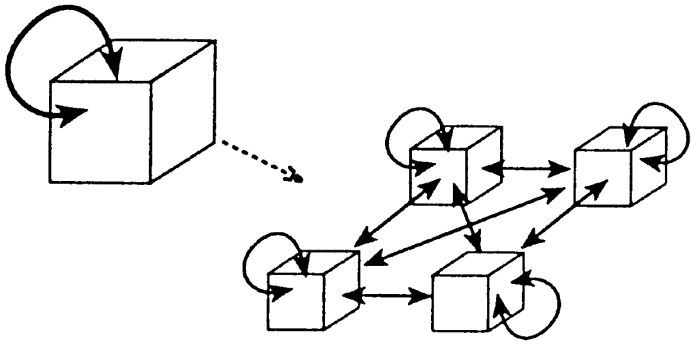

Figure 3: *Subdivision of a self–link.*

if ρ_i either means reflectance for surfaces or albedo for volumes.

The form factor F_{ij} depends on the four cases:

Surface – Surface:

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\tau \cos\theta_i \cos\theta_j}{\pi r^2} dA_j dA_i \tag{6}$$

Surface – Volume:

$$F_{jk} = \frac{1}{A_j} \int_{A_j} \int_{V_k} \frac{\tau \kappa_k \cos\theta_j}{\pi r^2} dV_k dA_j \tag{7}$$

Volume – Surface:

$$F_{ki} = \frac{1}{V_k} \int_{V_k} \int_{A_i} \frac{\tau \cos\theta_i}{4\pi r^2} dA_i dV_k \tag{8}$$

Volume – Volume:

$$F_{mk} = \frac{1}{V_m} \int_{V_m} \int_{V_k} \frac{\tau \kappa_k}{4\pi r^2} dV_k dV_m \tag{9}$$

where τ means transmittance between elements, A_i area of patch i, V_k volume of volume k, and κ_k extinction coefficient of volume k.

4.1. Algorithm for Isotropic Volumes

Using the formulae for form factors the hierarchical radiosity algorithm can easily be adapted for volumes. The algorithm for volumes decides – recursively – whether the transfer of energy is correctly represented at the current level, otherwise it subdivides one of the volumes. It first computes a bound on the error incurred by a link at the current level. This can be done either by bounding the error on the form factor alone, or by bounding the radiosity transfer (BF refinement). If a link is established, the form factor has to be computed exactly with the true transmittance factor.

For volumes a link has to be established to the volume itself during the initial linking phase. This is a fundamental difference to planar surfaces. So the entire initial linking phase can be replaced by the creation of a single link, from the root of the hierarchy – the whole scene – to itself. If a self-link is subdivided, links have to be created for all possible pairs of hierarchical children, including new self-links (Fig. 3).

After all links have been established, energy transfers are computed in a gathering step. To get the correct radiosities a push–pull procedure is necessary as in the surface case. While for surfaces often a constant reflection is assumed, the albedo value of a volume may generally vary within the volume.

4.2. Clustering Algorithm

The hierarchical algorithm for volumes is now applied to clusters. The radiative properties of surface clusters are simulated by isotropic scattering volumes. An isotropic scattering volume element has an equivalent area of $A = 4\kappa V$. Therefore an equivalent extinction coefficient can be computed as

$$\kappa = \frac{A}{4V} \tag{10}$$

The cost of visibility computation is dramatically lowered by the use of semi–transparent volumes. Links and energy exchanges are computed between clusters in a top–down manner, thus no initial linking of the surfaces is needed.

Finally Sillion suggests a general clustering strategy for a mixture of surfaces and volumes. The approximation of clusters by isotropic media is only valid for extremely small surfaces. For finite–sized surfaces the radiance distribution leaving a cluster is generally not isotropic. To overcome these problems in a general hierarchical scene, surfaces are allowed to be attached at the lowest level of the volume hierarchy.

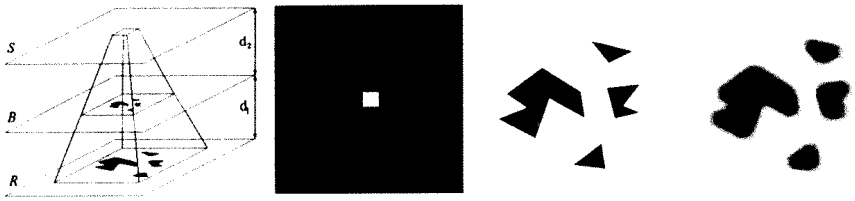

Figure 4: *A simple case of parallel light source (S), occluder (B), and receiver (R).*

Visibility relationships within a cluster are ignored, but the relative orientation of each individual element with respect to the direction of energy transfer is calculated correctly. So the additional cost for the surface calculations remains reasonable.

This method's merits are to have combined hierarchical radiosity methods for surfaces and volumes. But his clustering algorithm is not very suitable for large vegetation scenes because of the original assumption of isotropy of light scattering. Light is definitely not scattered isotropically in leaf canopies as later studies have shown. The isotropy assumption rather holds for scenes with clouds of smoke or vapour.

4.3. Soft Shadows Textures

Soler and Sillion developed a fast and accurate method for rendering soft shadows [Sole 98]. Those shadows occur if extended light sources illuminate planar surfaces through occluding clusters. Soft shadows especially are important with plants in indoor scenes. This method is based on convolution.

The convolution is derived from the special case, where the light source, the receiver and the occluder are all planar, and lie in parallel planes (Fig. 4). The irradiance at a point y on the receiver S is:

$$H(y) = \frac{E}{\pi} \int_S \frac{\cos\theta(x,y)\cos\theta'(x,y)}{d(x,y)^2} v(x,y)\,dx \qquad (11)$$

where E is the exitance of the source, $d(x,y)$ is the distance between x and y, θ and θ' the incident angles of the ray $x \to y$ on the source and the receiver, and $v(x,y)$ is a binary *visibility function*. The common approximation in radiosity algorithms is the separation of the visibility factor:

$$H(y) \approx \frac{E}{A_S} \underbrace{\int_S \frac{\cos\theta\cos\theta'}{\pi d^2} dx}_{F_S(y)} \underbrace{\int_S v(x,y)\,dx}_{V(y)} \qquad (12)$$

The first term $F_S(y)$ is the unoccluded point–to–area form factor from y to the source, and the second term $V(y)$ is the visible area of the source as seen from y. The unoccluded form factor can be computed using integration formulae.

Computing $V(y)$ is equivalent to projecting the blocker onto the source from y. The following characteristic functions of the source and the blocker in their respective planes are defined as:

$$S(x) = \begin{cases} 1 & \text{if x is on the source} \\ 0 & \text{else} \end{cases}$$

$$P(x) = \begin{cases} 0 & \text{if x is on the occluder} \\ 1 & \text{else} \end{cases}$$

The visibility factor can then be written as:

$$V(y) = \int_S P\left(\frac{d_1 x + d_2 y}{d_1 + d_2}\right) dx \qquad (13)$$

Now the integration is extended over the entire plane to show that $V(y)$ is a convolution:

$$V(y) = \int_{\mathbf{R}^2} S(x) P\left(\frac{d_1 x + d_2 y}{d_1 + d_2}\right) dx \qquad (14)$$

$$= \left(\frac{d_2}{d_1}\right)^2 \int_{\mathbf{R}^2} s_\alpha(-t)\, p_{1+\alpha}(t+y)\,dt = \frac{1}{\alpha^2}(s_\alpha \star p_{1+\alpha})(y) \qquad (15)$$

where

$$\alpha = \frac{d_1}{d_2} \qquad s_\alpha(x) = S\left(-\frac{1}{\alpha}x\right) \qquad p_{1+\alpha}(x) = P\left(\frac{1}{1+\alpha}x\right) \qquad (16)$$

and $\star$ denotes a convolution operation. The radiosity function B can now be expressed:

$$B(y) \approx \frac{\rho(y)E}{A_S \alpha^2} F_S(y)(s_\alpha \star p_{1+\alpha})(y) \qquad (17)$$

Soft shadows can be rendered by computing a shadow map using the convolution formula, and the shadow map is used as a texture to modulate the illumination function $\rho(y)EF_S(y)$. The method can be generalized for non–parallel configurations and non–uniform light sources. Practically the convolution is computed by the use of the two dimensional FFT (Fast Fourier transform).

4.4. Contribution to Vegetation Scenes

The hierarchical clustering method introduced a new approach to hierarchical radiosity, and made it possible to calculate energy exchanges in very complex scenes. Since vegetation scenes can be reckoned among the most complex scenes, the hierarchical clustering method was at least an important step in the direction to a complete simulation of light distribution within scenes of plants.

This method also combines the theory for clusters of surfaces with that for clusters of volumes. This relation raised hopes that clusters of plant organs could be rendered as a volume density distribution of particles with isotropic properties, resulting in a much faster calculation than for surfaces. Unfortunately (spherical) isotropy in many cases is not the appropriate illumination model for plant canopies. But for the rotationally symmetric model that fits much closer to the nature there do not exist so simple relations between volumes and surfaces.

5. Radiosity for Large Outdoor Scenes

In 1997 Daubert et al tried to integrate all methods that assist in radiosity calculations for outdoor scenes with terrain–maps, plants and buildings within a single hierarchical algorithm [Daub 97]. Also the sky and the sun are included in the hierarchy.

At the heart of the hierarchical simulation framework is the identification of a scene with a hierarchy of H–elements, connected by links which represent light exchanges. The highest level comprises the entire scene. H–elements can be of different nature, such as clusters, or portions of surfaces, or whole surfaces, but they all share a number of properties and characteristics, allowing the specification of many computational operations at an abstract level. All H–elements possess radiosity and reflectance information, and a set of links representing energy exchange at a certain level of hierarchy. Specialized H–elements deal with the specifics of clusters or polygons.

Before the linking phase can start, input objects (surfaces, meshes, ...) are grouped into clusters, and an initial self–link of the root node to itself is established. This is a very coarse representation of all energy transfer in the scene. During the linking phase links between H–elements are first refined using a refinement engine, resulting in the subdivision of the corresponding H–elements where appropriate, as described in [Hanr 91] and [Sill 94]. Subsequently irradiance is gathered across these links. The refiner estimates how well the link represents the light transfer by an error based criterion with respect to a user–defined tolerance. The criterion can be represented by BF ($radiosity \times form\,factor$) or better using upper and lower bounds on the energy transfer across the link. Therefore the form factor between two H–elements or the radiance received has to be estimated by sampling of a kernel function according to [Sill 94].

5.1. Geometric Properties

By a generic hierarchical mesh information the complexity of the geometric object is hidden (e. g., a terrain mesh or a tensor–product spline surface). As hierarchical subdivision becomes necessary, the structure uses the hidden geometric complexity to create finer hierarchically subdivided levels. At the lowest level of subdivision the structure reverts to classical regular subdivision (e. g. quadtrees).

It is necessary that all connectivity information is maintained at every level of hierarchy. In the quadtree or triangular structures created below the finest level of mesh subdivision it must be possible to find their neighbours across boundaries. An example is shown for quadtrees in Fig. 5. An abstract class *generic edge* is defined, allowing to have a unified interface for neighbour finding, independent of the actual type of mesh in question. Generic edges are defined in an appropriately chosen *parameter space*, mapping the terrain point data to $[0,1]^2$. Neighbourhood information is needed to correctly perform per-vertex shading (Gouraud shading).

5.2. Natural Light Sources

For outdoor scenes the sun can be modeled as a parallel source, and the sky is considered a hemispherical source at infinity. These light sources must become hierarchical elements which provide the same interface as the other elements of the hierarchy.

Attention must be paid to the sampling mechanisms used for these sources, for the calculation of energy transfer and form factor estimates, because the formulae cannot be used directly with sources at infinity. The subdivision of the sky dome is done by a quadtree representation of the hemisphere's parameter space, expressed by the altitude u and the azimuth v. u is the angle above horizon, and v denotes the angle in the plane west from the south (Fig. 6). The initial sky patch is the whole hemisphere, $(u,v) \in \left[0,\frac{\pi}{2}\right] \times [0,2\pi]$. Every hierarchical sky patch is therefore defined by a parameter range and a constant radiance value. Assuming full visibility, the contribution of a sky patch q to the irradiance on a scene element p is obtained as follows:

$$L_i^p = R_{pq} S_p \Omega_q J_q \tag{18}$$

where R_{pq} is the receiver factor of p, S_p is the scale factor of p, Ω_q is the solid angle subtended by q, and J_q is the radiance of q. For surfaces R_{pq} is $\cos\theta_i$ and S_p is 1, for volumes $R_{pq} = 1$ and $S_p = \frac{1}{4}$. The solid angle Ω_q is given by

$$\Omega_q = (v_1 - v_0)(\sin u_1 - \sin u_0) \tag{19}$$

Parallel sources never need to be split, and the irradiance cast on an element is just the source's flux density modified

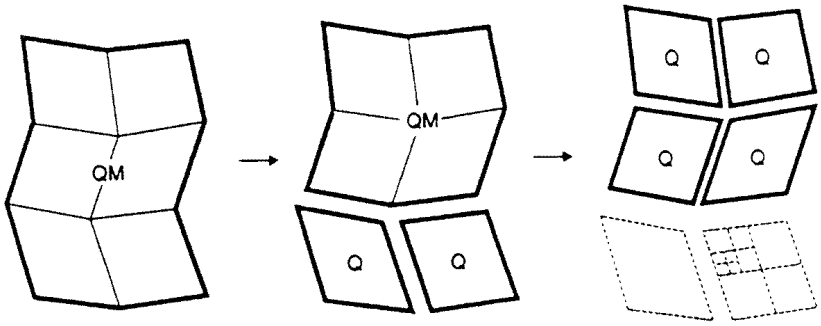

Figure 5: *A sequence of splitting operations starting with one quad mesh (QM) and resulting in quadrilaterals (Q), which are subdivided into regular quadtrees.*

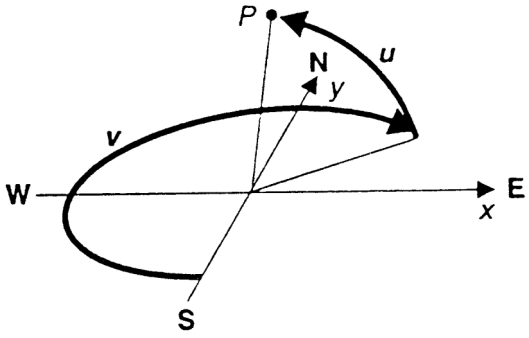

Figure 6: *Parameter space for a skylight object.*

by the receiver's orientation and type. The irradiance of a parallel light q on a scene element p is given by

$$L_i^p = R_{pq} S_p \Phi_q \tag{20}$$

where Φ_q is the source's flux density.

External light sources are not included in the top–level cluster of the scene hierarchy. In the presence of n external lights, the simulation would start with $n+1$ initial links. Fig. 7 shows an initial link setup for a scene with sunlight and skylight.

5.3. Contribution to Vegetation Scenes

There is just another way the volume scattering method can be used for the rendering of large vegetation scenes. Very dense plant accumulations like high grass, bushes, brushwood, or leaf canopy can be rendered as the periodic occurence of one pattern in x- and y-direction without significant loss of impression of a natural scene. Like the design of a wallpaper that can be reduced to an rectangular area and then repeated in both directions, a cubic volume of one or more plants can represent the whole scene if it is repeated in x- and y-direction upto infinity.

The scene will now be built up in a hierarchical way, consisting of the light sources (direct sun and skylight), the accumulation of the plants with a self–link representing all light exchanges within it, and the ground with certain reflection properties. In the first step of hierarchical subdivision the infinite accumulation of plants is replaced by one central volume that contains all necessary geometrical information for the scene and eight identical volumes surrounding the central one. Those surrounding volumes are only copies of the central volume, so that there can be drawn links for the light interactions with the neighbour volumes.

It is necessary for this method to work that the scene is potentially infinite. So during the subdivision process no outer surfaces are created. The surrounding volumes should be thought rather in a torus topology so that the leftmost surface is close to the rightmost one, and the foremost surface is close to the rearmost one.

During further subdivisions the central volume will be refined until the given accuracy is reached. It should be noted that only light exchanges with the neighbour volumes are

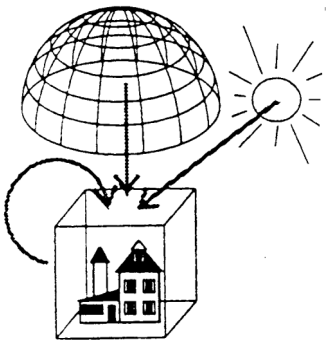

Figure 7: *Initial links for a scene with a skylight and a sunlight object.*

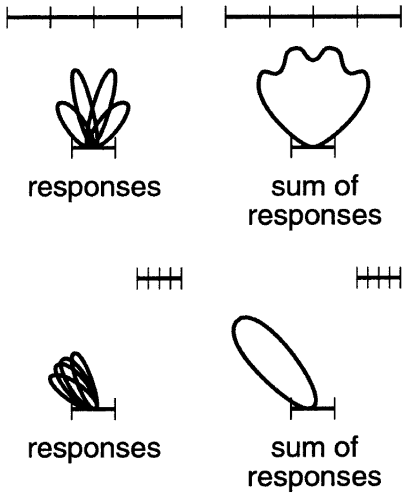

Figure 8: *Incident light portions at a reflector from four senders for two sender configurations. Although the single responses are similar for each configuration, the sum differs significantly.*

considered by this method. Light contributions from far volumes are neglected. This systematic error can be ignored if the vegetation accumulation is dense enough or if the central volume will be chosen sufficiently large.

6. Three Point Clustering

In 1998 Stamminger et al presented a three point clustering algorithm in [Stam 98] that combines the ideas of clustering [Sill 94], hierarchical radiosity [Hanr 91], line space hierarchy [Dret 97] and bounded radiosity [Stam 97]. The presented method allows radiosity computations for diffuse and non–diffuse scenes of high complexity, even those scenes that were previously calculable only with Monte–Carlo sampling methods.

The new method exploits the so called three–point parameterization of the rendering equation:

$$L(y,z) = L^e(y,z) + \int_S f(x,y,z)G(x,y)L(x,y)dA_x \qquad (21)$$

where S is the set of all surfaces in the scene that send light to z reflected at y. $L^e(y,z)$ is the emission from y in direction to z, $f(x,y,z)$ is the BRDF (bidirectional reflectance distribution function) at a point y for light coming from x reflected towards z. $G(x,y)$ describes the geometric relation between the surfaces at x and y

$$G(x,y) = \frac{\cos(n(x),y-x)\cos(n(y),x-y)}{\pi \mid x-y \mid^2} V(x,y) \qquad (22)$$

where V(x,y) is the visibility between x and y. $L(x,y)$ is

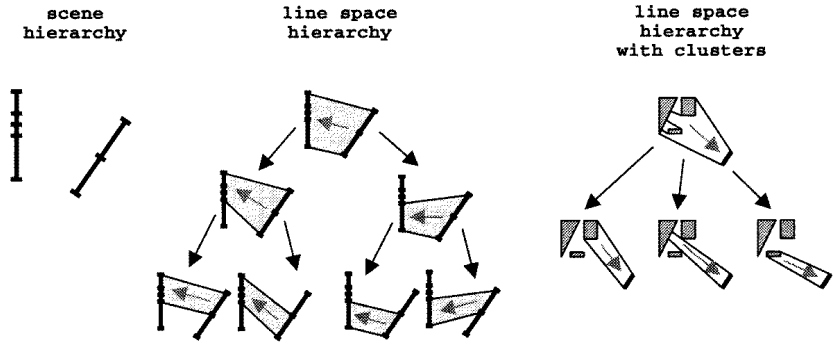

Figure 9: *Creating a line space hierarchy. Center: Hierarchy on the set of connections. The root is the set of all connections. Right: Hierarchy transformed to clusters.*

the equilibrium radiance distribution. The rendering equation can also be written as

$$L = L^e + \mathcal{K}\mathcal{G}L \tag{23}$$

with the *local reflection operator* $\mathcal{K}$ and the *field radiance operator* $\mathcal{G}$. For purely diffuse surfaces (with constant BRDF) instead of the four dimensional radiance distribution only a two dimensional radiosity has to be computed.

The *three point clustering* algorithm avoids the problem of hierarchical representation of different responses (Fig. 8). The hierarchical representation of L is stored in a *line space hierarchy* of all visible connections between scene objects. Fig. 9 shows the set of line segments $[s,t]$ that connects a sender s and a receiver t. $[s,t]$ is split into two subsets $[s,t_1]$ and $[s,t_2]$.

Starting with the radiance distribution $\tilde{L}^0$, the self emission of the objects, Jacobi or Gauss–Seidel iteration is used to approximate the global solution:

$$\tilde{L}^{i+1} = \mathcal{K}\mathcal{G}\tilde{L}^i + \tilde{L}^0 \tag{24}$$

For a single value $\tilde{L}^{i+1}(x,y)$ all incident light at surface point x has to be collected from the line space hierarchy. The incident light portions then have to be reflected towards y and added to the sum $(\mathcal{K}\mathcal{G}\tilde{L}^i)(x,y)$.

In Fig. 10 the dotted arrows show the line space nodes illuminating the grandfather, father and s itself. The box on the right shows how the set of all interactions adaptively partitions the set of all directions around s. The recursion starts at the line space root $[s^0,s^0]$. If the representation of $\tilde{L}^{i+1}$ within a given line space range is not possible by a constant value, $\tilde{L}^{i+1}$ is determined recursively on the level of the children. Otherwise, recursion ends. To avoid sampling problems and to obtain accurate results, not only exitant radiance values are computed, but also their bounds.

6.1. Line Space Hierarchy

The set of all oriented line segments $\mathbf{R}$ is the set of all mutually visible surface points:

$$\mathbf{R} = \{(x,y) \mid x,y \in S \wedge V(x,y) = 1\} \tag{25}$$

While a *scene hierarchy* is defined on the surfaces, a *line space hierarchy* is defined on $\mathbf{R}$, starting with the line space root $[s^0,s^0]$. As a refiner function D_{area} is used:

$$D_{area}(s,t) = \begin{cases} sender & if \quad s \neq t \wedge area(s) \geq area(t) \\ receiver & if \quad s \neq t \wedge area(s) < area(t) \\ both & if \quad s = t \quad (selflink) \end{cases}$$

For the computation of the light transport by a line space node, a measure on the line segments $\mathbf{R}$ is defined. The line segments between two scene object nodes s and t have the measure:

$$\|[s,t]\| = \int_s \int_t \frac{\cos(n_u, v-u)\cos(n_v, u-v)}{|\, v-u \,|^2} V(u,v)\,du\,dv \tag{26}$$

This measure is equivalent to the form factor from s to t times the area of the receiver t. If the radiance L between s and t is assumed to be constant, the flux from s to t is

$$\Phi(s,t) = L \cdot \|[s,t]\| \tag{27}$$

and the irradiance at t due to s is

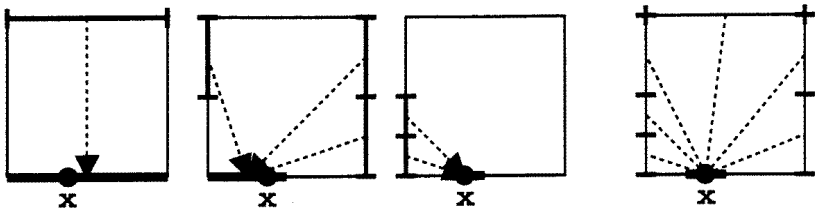

Figure 10: *Three walls of a box hierarchically illuminating a patch on the bottom line.*

$$E^s(t) = L \cdot \frac{\|[s,t]\|}{area(t)} \tag{28}$$

6.2. Three Point Clustering Algorithm

Light transport in the scene is computed by a top–down traversal of the line space hierarchy. Each full traversal computes one iteration of a Jacobi or Gauss–Seidel iteration. During one traversal , incident light is reflected in the scene, and an adaptive representation of the reflected light is computed. The main loop can be described as follows:

```
main loop:
 until converged
  traverse line space hierarchy
   recompute radiance L of current node [s,t]
   if refinement criterion fulfilled for [s,t]
    refine
    set L to average radiance of children nodes
   else
    store L in [s,t]
```

The radiance for $[s,t]$ is the sum of the emission of the sender towards the receiver plus the radiance reflected off at the sender towards the receiver:

```
recompute radiance of current node [s,t]
 L = emission of s towards t
 traverse line space hierarchy for all
       nodes [r,s'] illuminating s
  L += reflection of [r,s'] towards t
return L
```

The set of all directions from s to t is bounded with a cone. With this approach, light sources that emit light in a very small directional range can be used without sampling problems. A similar approach is used for obtaining the value of the BRDF.

The measure of a line space node is required during reflection computations. First, the measure is approximated without considering visibility $V(x,y)$. Also an upper and a lower bound are computed. The method works for arbitrary objects including curved surfaces. In the second phase, the occlusion between the two objects is approximated separately by shooting a number of sample rays between the objects. The product of the sampled visibility factor and the unoccluded line space measure is then used as a measure for the line space node.

In the case of purely diffuse objects an irradiance field is added, so irradiance can be used instead of radiance. Thus, the partial traversal to gather incident light can be omitted for diffuse surfaces.

6.3. Contribution to Vegetation Scenes

Three point clustering is the most recent development in hierarchical methods. The whole scene can be seen as the root of the hierarchy. This can be an advantage for large vegetation scenes, if the refinement process resembles the natural structure of the scene.

In three point clustering curved surfaces can be treated as whole objects. It is not necessary to approximate curved surfaces by planar patches in advance. Since plants mainly consist of curved surfaces, many unnecessary subdivisions can be avoided with three point clustering. Specular and glossy effects can easily be integrated with radiosity calculations by this method. Three point clustering can therefore be considered as a very good base for further specialized algorithms for large vegetation scenes [Mast 99].

7. Plane–Parallel Radiance Transport for Vegetation Scenes

Nelson Max presented in [Max 97] a quite different, very important approach for the investigation of the global illumination in large vegetation scenes consisting of clusters of similar plants. They considered the leaves, stems, and branches as volume density of scattering surfaces. Ordinary differential equations (ODE) are written for the multiply scattered radiance as a function of height above the ground, with the sky radiance and ground reflectance as boundary conditions.

The resulting radiance distribution can be used to precompute diffuse and specular ambient shading tables. Ambient light in [Max 97] has not the meaning of constant approximation of global illumination, but rather is a function of height and surface normal and can be directly used in rendering the clusters of leaves or branches with high accuracy.

7.1. Plane–Parallel Radiance Transport Equations

The angular variation of the radiance will be represented by the division of the unit sphere into angular bins which are discrete in both directions. First the general equation of radiance transport is presented:

$$\frac{dI(X,\omega)}{ds} = -\kappa(X,\omega)I(X,\omega) + \int_{4\pi} \kappa(X,\omega') \cdot a \cdot r(X,\omega',\omega) I(X,\omega') d\omega' \tag{29}$$

where X is the position (x,y,z), ω and ω' are ray directions on the unit sphere, $I(X,\omega)$ is the radiance at position X flowing along a ray in direction ω, s is the length along this ray, $r(X,\omega',\omega)$ is the scattering phase function from direction ω' to direction ω (representing both reflection from and transmission to the leaves). κ is the extinction or beam attenuation coefficient representing absorption plus scattering, and a is the albedo, the fraction of the attenuation that is scattered rather than absorbed. κ represents geometric occlusion from the polygon surface area (of a leaf) projected normal to ω, per unit volume. It depends on ω unless the distribution of the polygon normals is completely random.

In spherical coordinates ω is (θ,φ), where θ is the polar angle, measured from the vertical z axis, and φ is the azimuthal angle, measured from the x axis in the xy-plane, and $\omega' = (\theta',\varphi')$. Then $r(X,\omega',\omega)$ depends only on X,θ,θ', and $|\varphi - \varphi'|$ as long as the BRDF and BTDF of the leaves satisfy similar assumptions, and the distribution of the surface normals is rotationally symmetric around the z axis. These conditions are fulfilled for rotationally symmetric plants, or random rotational placement of arbitrary plants.

7.2. Discrete Division of the Unit Sphere

The unit sphere is divided into a number of discrete ω direction bins, $2m$ in θ direction, and n in φ direction. The first and the last θ bins are joined into a pair of polar caps, giving a total of $(2m-2)n+2$ bins. A square matrix of this size is precomputed where the product $\kappa(X,\omega') \cdot a \cdot r(X,\omega',\omega)$ is stored, built up from the diffuse and specular reflection and transmission coefficients, for each color band.

Because of the the plane–parallel case, and because of the rotational symmetry the integro–differential equation can now be written as a linear system of ordinary differential equations:

$$\mu_i \frac{dI_i(z)}{dz} = \sum_j I_j(z) r_{ji}(z) \tag{30}$$

where μ_i is $\cos\theta_i$ of the i^{th} direction bin. $r_{ji}(z)$ is the bin–to–bin scattering matrix $\kappa(X,\omega') \cdot a \cdot r(X,\omega',\omega)$ with $r_{ii}(z) = -\kappa(X,\omega)$ the extinction term.

This system of equations is again replaced by two coupled vector differential equations

$$\frac{dI_u(z)}{dz} = I_u(z)\tau_{uu}(z) + I_d(z)\rho_{du}(z) \tag{31}$$

$$\frac{dI_d(z)}{dz} = I_u(z)\rho_{ud}(z) + I_d(z)\tau_{dd}(z) \tag{32}$$

where $I_u(z)$ and $I_d(z)$ are the radiance vectors for the upward flow and the downward flow respectively. One boundary condition is known: $I_d(h)$ is the illumination from the sun and the sky. As a further condition the BRDF of the ground at $z = 0$ is given, resulting in

$$I_u(0) = I_d(0) \cdot F \tag{33}$$

where F is the bin–to–bin matrix approximating the BRDF of the ground.

Before solving the equations for I_u and I_d an auxiliary differential equation for $F(z)$ with $I_u(z) = I_d(z) \cdot F(z)$ has to be solved:

$$\frac{dF(z)}{dz} = \rho_{du} + F(z)\tau_{uu} + \tau_{dd}F(z) + F(z)\rho_{ud}F(z) \tag{34}$$

with the initial condition $F(0) = F$, the known BRDF of the ground. Thus it can be integrated upwards incrementally from $z = 0$ to $z = h$.

When $F(h)$ is calculated, $I_u(h)$ can be determined, and finally the coupled vector differential equations can be integrated downwards from from $z = h$ to $z = 0$. Alternatively $I_u(z)$ and $I_d(z)$ can be expanded as finite Fourier series in φ, resulting in a faster numerical integration.

7.3. Shading

The actual simulations were performed with $2m = 20$ latitude bins and $n = 24$ longitude bins, giving $2M = 434$ bins totally. As BRDF values not the evaluated matrix values were used for shading, instead a Phong BRDF was used, consisting of a diffuse reflection plus a specular term that depends only on the angle between the viewing direction and the perfect mirror reflection of the incident ray:

$$f(L,V,N,\lambda)\cos\theta = d(\lambda)\cos\theta + s\{R \cdot V\}^c \tag{35}$$

where f is the BRDF, L is the direction to the light source opposite to the incoming ray direction ω, V is the outgoing

ray direction, N is the surface normal, λ is the wavelength of the light, θ is the angle between L and N, d is the wavelength dependent diffuse reflection coefficient, s is a wavelength independent specular reflection coefficient, R is the mirror reflection direction, the rotation of L about N, c is an arbitrary positive exponent (actually 20), and the special dot product $\{R \cdot V\}$ is the dot product $R \cdot V$, if positive, and 0 otherwise.

With $I(\omega,\lambda) = I(-L,\lambda)$ the surface radiance $S(V,\lambda)$ in viewing direction V becomes:

$$S(V,\lambda) = \int_{2\pi} I(\omega,\lambda)\, f(-\omega,V,N,\lambda)\cos\theta d\omega \qquad (36)$$

$$= d(\lambda)\int_{2\pi} I(\omega,\lambda)\cos\theta d\omega + s\int_{2\pi} I(\omega,\lambda)\{-\omega\cdot Q\}^c d\omega \qquad (37)$$

where Q is the rotation of V about N.

During the shading process the effect of attenuated sunlight is added by the use of the Phong BRDF for the sun as a discrete parallel source, only if the surface point is not in shadow.

7.4. Contribution to Vegetation Scenes

By the use of the plane–parallel radiance transport method one can precompute diffuse and specular "ambient" shading tables, as a function of height and surface normal. The values can then be used in rendering vegetation scenes with large numbers of plants. In the moment there are some restrictive assumptions for the plane–parallel transport method: the trees or groups of trees must meet the condition of cylindrical symmetry.

If it is possible to alleviate the restrictions to trees with some regular shape, one could precompute shading tables for many different species of trees, say one shading table for spruce trees, one for typical fir trees, and so on. Then one can imagine the following approach for fully heterogeneous vegetation scenes: The individual plants are considered as complete objects, and energy exchange between the objects is determined by a hierarchical method like three point transport. The light distribution within the objects is calculated from the precomputed shading tables and the direct illumination.

8. Nested Radiosity

Hondermarck et al present in [Hond 98] a study concerning radiosity for vegetation scenes, where they distinguish between far and close radiosity. The method is called nested radiosity and is thought to be used in computer graphics as well as in environmental physics for crop simulation.

A polygon A_j is considered to be close to A_i, if it lies within a sphere S_i of a diameter D_S that is centered at the center of A_i. For the similar sized elements of canopy the diameter D_S defines a high bound for the form factor between two polygons (e. g. leaves). The radiosity equation can now be written as follows:

$$B_i = B_i^0 + \rho_i \sum_{j\in S_{i+}} B_j F_{ij} + \tau_i \sum_{j\in S_{i-}} B_j F_{ij} + B_i^{far} \qquad (38)$$

where S_{i+} is the upper half of the sphere S_i, S_{i-} is the lower half, B_i^0 corresponds to the diffusion of uncollided light from sun and sky, B_i^{far} is the radiosity due to the light by scattered polygons outside S_i. Far radiosity is approximated by a light distribution. Let Ω^* be the solid angle domain for which there is no occlusion between A_i and S_i. Then B_i^{far} can be expressed as follows:

$$B_i^{far} = \rho_i \int_{\Omega^*_{i+}} L(P,-\omega)\cos\theta_i d\omega + \tau_i \int_{\Omega^*_{i-}} L(P,-\omega)\cos\theta_i d\omega \qquad (39)$$

where $L(P,-\omega)$ is the radiance at a point P of the sphere in a direction $-\omega$, and θ_i is the angle between n_i and ω.

The radiative equilibrium of all surfaces of the canopy can now be written

$$M \cdot B = B^{far} + B^0 \qquad (40)$$

with some matrix M that can be calculated from F_{ij}, ρ_i and τ_i, and B, B^{far} and B^0 being the vectors containing B_i, B_i^{far} and B_i^0.

Light sources are the sun and the sky that are located at infinity. The angular distribution of radiance is approximated by discretizing the sky hemisphere in solid angles Ω_n. A multilayer model is used to compute the field of radiance $L(P,\omega)$. By this method the canopy is described as a set of homogeneous infinite layers, and diffuse fluxes are assumed isotropic. Irradiance at any arbitrary point does only depend on the elevation z, and is computed by linear interpolation.

For each polygon i the set of inner polygons j partially or completely inside the sphere S_i surrounding i is determined. For this purpose the scene is divided into a regular cubic grid, whose edge is equal to the diameter of the sphere.

The differentiation between far and close radiosity allows for a parallelization of the algorithm on computers with distributed memory. Especially the form factor calculation can be distributed over a number of processors because of the locality property of the exchanges within the sphere S_i. But the entire canopy has to fit into the memory of a master machine. The other machines only hold those parts temporarily that belong to objects within the proximity spheres. The speedup of

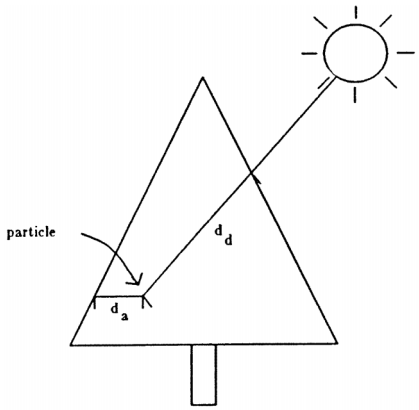

Figure 11: *Distances used by Tree Shading Algorithms.*

the parallel nested radiosity algorithm is good only for scenes with a homogeneous distribution of polygons.

9. Realistic Modelling of Vegetation Scenes

Prusinkiewicz and Lindenmayer had already presented fine pictures of single plants and medium sized plant scenes in [Prus 90] by the use of a raytracing algorithm. The pictures show accurate glossy effects, but no global illumination effects of diffuse reflection.

Deussen et al enhanced Prusinkiewicz's method for very large vegetation scenes [Deus 98]. They use sophisticated methods for building the terrain and for positioning the plants in an artificial or in an ecological way. Great care is applied for instancing and efficient storing of the objects. Rendering is done by raycasting and a Z–buffer algorithm with no global effects.

9.1. Approximate Rendering Algorithms

As early as 1985 Reeves and Blau began to solve the problem of modelling and of global illumination for large vegetation scenes [Reev 85] by particle systems. The results of this paper were used in the film *The Adventures of André and Wally B.* The so called diffuse shading component for a particle varies with the distance into the tree from the light source, d_d, as shown in Fig. 11. An exponential attenuation is assumed:

$$D = e^{-\alpha d_d} \tag{41}$$

Random highlights are added by stochastically turning on a specular component whenever d_d is small and the cosine of the angle between the light direction and the branch direction is close to zero. The so called ambient component varies exponentially with the distance from a particle to the tree's bounding volume in a direction parallel to the ground:

$$A = \max\left(e^{-\beta d_a}, A_{min}\right) \tag{42}$$

where A_{min} is the minimum ambient light in the tree. It guarantees that there is some light even in the deepest interior of the tree.

Even though Reeves' method is rather heuristic than mathematically based, the animated pictures of groups of different trees as well as of complete forests look quite natural and give the impression of an accurate global illumination.

10. Conclusion

True heterogeneous vegetation scenes cannot be rendered with current algorithms yet. Such scenes can occur in real mixed forests consisting e. g. of pine trees, spruce trees, oak trees and beech trees. Also if the trees are of the same species, but with different shapes and sizes, the analysis of light distribution within the canopy will be very complicated. The authors expect the solution of the global illumination problem for very large and general vegetation scenes to be reached on the base of a sophisticated hierarchical algorithm like three point clustering.

They believe that the hierarchical method can be enhanced by a natural subdivision algorithm that splits the whole scene into single plants or groups of entire plants. At a deeper level a single tree can be subdivided into e. g. the cluster of leaves, the cluster of branches, and the solid of the trunc. For the form factor calculations between clusters and other objects the stochastic properties of the clusters should be used, like rotational symmetries. So further subdivisions of clusters can be avoided. Light distribution within the clusters (self–link) can be determined by typical radiance values for those species of plants that occur in the scene.

About Authors

Helmut Mastal is a PhD. student at the Institute of Computer Graphics, Vienna University of Technology, Vienna, Austria. His main research topic is radiosity methods for plants.

Robert F. Tobler is Researcher of Computer Science at the Institute of Computer Graphics, Vienna University of technology. His main research topics are photorealistic images, global illumination techniques, especially radiosity.

Werner Purgathofer is Professor of Practical Informatics at the Vienna University of Technology, where he is the head of Visualization and Animation Group at the Institute of Computer Graphics. His main research topic is global illumination, especially stochastic radiosity methods. His other interests include color science and visualization.

References

[Cohe 85] M. Cohen, D. Greenberg: The Hemi-Cube – A Radiosity Solution for Complex Environments. *Computer Graphics*, 19(3), 1985

[Cohe 86] M. Cohen, D. Greenberg, D. Immel, P. Brock: An Efficient Radiosity Approach for Realistic Image Synthesis. *IEEE Computer Graphics and Applications*, 6(2), 1986

[Cohe 88] M. Cohen, S. Chen, J. Wallace, D. Greenberg: A Progressive Refinement Approach for Fast Radiosity Image Generation. *Computer Graphics*, 22(4), 1988

[Daub 97] K. Daubert, H. Schirmacher, F. Sillion, G. Drettakis: Hierarchical Lighting Simulation for Outdoor Scenes. *Rendering Techniques '97 (Proceedings of the 8th Eurographics Workshop on Rendering)*, St. Etienne, France, Springer, 1997

[Deus 98] O. Deussen, P. Hanrahan, B. Lintermann, R. Mech, M. Pharr, P. Prusinkiewicz: Realistic modeling and rendering of plant ecosystems. *Computer Graphics (SIGGRAPH '98 Proceedings)*, Orlando, July 1998

[Dret 97] G. Drettakis and F. Sillion: Interactive Update of Global Illumination using a Line–Space Hierarchy. *Computer Graphics (SIGGRAPH '97Proceedings)*, August 1997

[Gora 84] C. Goral, K. Torrance, D. Greenberg, B. Battaille: Modelling the Interaction of Light Between Diffuse Surfaces. *Computer Graphics*, 18(3), 1984

[Hanr 91] P. Hanrahan, D. Salzmann, L. Aupperle: A Rapid Hierarchical Radiosity Algorithm. *Computer Graphics*, 25(4), 1991

[Hond 98] J. Hondermarck, M. Chelle, C. Renaud, B. Andrieu: Parallel Form Factors Computation for Radiative Transfers in Vegetation. *Proceedings of the 2nd Eurographics Workshop on Parallel Graphics and Visualization*, Rennes, France, 1998

[Mast 99] H. Mastal, R. F. Tobler, W. Purgathofer: Radiosity Calculations for Rotational Surfaces. *Proceedings of the 7th International Conference in Central Europe on Computer Graphics, Visualization and Interactive Digital Media '99*, Plzen, Czech Republic, February 1999

[Max 97] N. Max, C. Mobley, B. Keating, En-Hua Wu: Plane–Parallel Radiance Transport for Global Illumination in Vegetation. *Rendering Techniques '97 (Proceedings of the 8th Eurographics Workshop on Rendering)*, St. Etienne, France, Springer, 1997

[Prus 90] P. Prusinkiewicz, A. Lindenmayer: The Algorithmic Beauty of Plants. Springer, New York, 1990. With J. S. Hanan, F. D. Fraccia, D. R. Fowler, M. J. M. de Boer, and L. Mercer.

[Reev 85] W. Reeves, R. Blau: Approximate and Probabilistic Algorithms for Shading and Rendering Structured Particle Systems. *Computer Graphics*, 19(3), 1985

[Shir 91] P. Shirley: Radiosity via Ray Tracing. *Graphics Gems II* (Editor: J. Arvo), Academic Press, London, 1991

[Sill 94] F. Sillion: Clustering and Volume Scattering for Hierarchical Radiosity Calculations. *Rendering Techniques '94 (Proceedings of the 5th Eurographics Workshop on Rendering)*, Darmstadt, Springer, 1994

[Sole 98] C. Soler, F. Sillion: Fast Calculation of Soft Shadow Textures Using Convolution. *Computer Graphics (SIGGRAPH '98 Proceedings)*, Orlando, July 1998

[Stam 97] M. Stamminger, P. Slusallek, H. Seidel: Bounded Radiosity – Illumination on General Surfaces and Clusters. *Computer Graphics Forum (EUROGRAPHICS '97 Proceedings)*, 16(3), September 1997

[Stam 98] M. Stamminger, P. Slusallek, H. Seidel: Three Point Clustering for Radiance Computations. *Rendering Techniques '98 (Proceedings of the 9th Eurographics Workshop on Rendering)*, Vienna, Austria, Springer 1998

[Wall 89] J. Wallace, K. Elmquist, E. Haines: A Ray Tracing Algorithm for Progressive Radiosity. *Computer Graphics Proceedings*, Boston, 1989