Hierarchical Lighting Simulation for Outdoor Scenes

Katja Daubert, Hartmut Schirmacher*
François X. Sillion, George Drettakis**

*i*MAGIS ***
Laboratoire GRAVIR/IMAG-INRIA

Abstract: Lighting algorithms for outdoor scenes suffer from the sheer geometric and lighting complexity of such environments. In this paper we introduce an efficient, *hierarchical* solution to the problem of outdoor illumination. Data structures and sampling algorithms are presented, permitting the integration of complex and natural objects in a hierarchical radiosity simulation system. This new approach allows the hierarchical simulation of radiant energy exchanges in outdoor scenes for the first time, including terrain and botanical models as well as sunlight and skylight. This is accomplished by providing the necessary tools to treat terrain meshes as a hierarchy of light-exchanging objects, as well as an efficient hierarchical representation for the sky dome. In addition, refinement criteria are adapted to the particular characteristics of natural lighting. Results of our implementation are presented including naturally-lit images of terrain-maps, trees and buildings.

1 Introduction

Rendering technology has made tremendous progress in the last decade, and it has become possible to perform simulations of light exchanges for moderately complex environments, yielding images of impressive realism. These algorithms exploit hierarchical solution methods to correctly account for secondary illumination and result in view-independent solutions. A vast majority of this research has been concentrated on indoor scenes.

As realistic simulated images receive more widespread attention, new potential applications for lighting simulations constantly appear, and present new challenges for state-of-the-art technology. One such challenge is the accurate and efficient simulation of lighting effects for *outdoor* scenes. Such simulations are needed for instance in visual simulation applications for various driving or flight simulators, for visual site planning in the context of architectural projects or urban modeling, or for the simulation of radiant exchanges in vegetation with applications in remote sensing or in the simulation of botanical processes.

Outdoor scenes tend to be especially challenging to simulate, because they present a combination of several difficulties that are poorly handled by existing radiosity algorithms, and in particular hierarchical solution methods.

First, outdoor scenes exhibit great geometric complexity: on the one hand, the sheer scale of outdoor scenes is significantly larger than that of indoor scenes. Furthermore, outdoor scenes often contain natural objects (terrain or trees) that have complex shapes.

A second major challenge with outdoor scenes is the complexity of the illumination itself. Natural illumination (i.e. from the sun and the sky) has specific properties such as

the concentration of a very large energy flux in a small solid angle (from the sun), and the fact that the sky effectively "surrounds" all objects. These properties require particular care when sampling for illumination simulation. Moreover, because outdoor scenes are always complex, efficient algorithms must be developed, since brute force approaches are bound to be intractable.

In this paper we propose solutions to the problems of geometric complexity and natural lighting. Our approach is based on *hierarchical* lighting algorithms, which permit efficient and elegant solutions. In particular, we show that complex geometry can be handled using a generic hierarchical surface description mechanism in the context of a clustering algorithm. This mechanism is integrated into an object-oriented framework, and its generic nature opens the way for the definition of other hierarchical objects for lighting simulation (such as complex meshes representing curved surfaces). We introduce appropriate representation, sampling and refinement machinery for hierarchical treatment of natural lights (sun and skylight). The resulting system is, to our knowledge, the first capable of providing *hierarchical* lighting solutions of complex outdoor scenes.

2 Previous work

Previous work in lighting simulation for outdoor scenes has been restricted mainly to the modeling of natural light. Similarly, the treatment of the geometric complexity, specifically for the problems posed by certain outdoor structures has received limited attention. We briefly review the relevant literature.

2.1 Previous Work in Natural Lighting

Much work has been done on modeling daylight with different atmospheric effects and designing global illumination models including such light sources. Several models with different complexity were proposed to compute skylight intensities [1, 13, 8, 7, 3]. The public domain system *Radiance* [20] provides source code ("gensky" and "skybright") to compute sky luminance using the CIE standard sky distribution model [1] for a given setup of hour, day, month, position on earth, and several weather-specific parameters. The sun position can also be specified directly by providing the solar angles.

To represent the skylight distribution, Nishita et al. [13] proposed a fixed set of band sources. In [3] subsampled band sources are inserted where the skylight distribution varies strongly, resulting in a representation with adaptive precision.

To speed up the time consuming integration of bands for global illumination, Tadamura et. al [19] use a regularly subdivided parallelepiped in a hemicube-like manner to determine the skylight contribution including visibility.

For indoor lighting, a set of "interface" surfaces can be used to represent the skylight coming through the windows of a room. In [9] the skylight contribution on window sample points is calculated in a preprocessing step and then window patches are used as area light sources for global illumination. This method is of course limited to indoor scenes.

The most closely related approach to our solution is that of Müller et al. [12]. In their algorithm the skylight distribution is directly stored in a set of radiosity patches which are placed far away from the rest of the scene to satisfy error bounds. This is achieved by the use of a regular subdivision of the sky hemisphere. The sky patches are then used for shooting in the first iteration of the radiosity simulation. Their method is based on the progressive refinement radiosity algorithm. The initial shooting phase is expensive (computation time of three hours was reported for a scene of 47,500 patches), but permits rapid change of the sky lighting condition, by storing a list of form-factors.

2.2 Treating Complex Geometry

Many outdoor models, such as terrains or mesh structures resulting from scanned-data, or complex man-built structures (bridges, dams etc.) are typically represented with complex meshes. Level-of-detail approaches for the treatment of such complex geometries are well known in the modeling community (e.g., [14, 4]). Despite this wealth of algorithms, the hierarchical treatment of these mesh structures has not previously found its way into lighting simulation algorithms.

2.3 Limitations of the Previous Approaches

The vast majority of previously described natural lighting algorithms require time consuming preprocessing steps. The method proposed in [12] is the first to integrate the sky as a radiosity object, but however a fixed subdivision is used. This is a severe restriction, lacking the benefits of both hierarchical solution [6] and clustering [18] for skylight objects. This also results in expensive calculations and lack of an overall representation of *global* illumination. To achieve efficient integration of natural lighting in a radiosity system, it is important first to provide multiresolution representations of natural lights, and also to ensure that the entire sky is always taken into account. It is important to note that when progressive refinement radiosity is used, this condition forces us to shoot energy from *all* sky patches, at a significant expense, before any meaningful solution can be obtained.

Similarly, in the case of complex meshes representing outdoor structures, a true hierarchical representation is required. A generic clustering algorithm [18, 16] applied to the polygons constituting the mesh would however be unsuitable, since we would lose all neighbouring information amongst mesh elements, needed for the smooth reconstruction of the radiosity function.

3 Unified Lighting Framework

As was presented in [18], a natural complete framework for hierarchical radiosity with clusters can be based on the concept of an abstract *Hierarchical Element* or *H-element*. These elements can be clusters (groups of objects) or surfaces. In the context of an object-oriented system, this abstract class hierarchy permits the definition of generic operators (refine, gather, push-pull) on H-elements, thus entirely defining the hierarchical radiosity process.

We briefly overview the central concepts in what follows, which will allow the reader to understand the extensions required to seamlessly integrate the new features required for outdoor scenes.

3.1 Hierarchical Elements

At the heart of our hierarchical simulation framework is the identification of a scene with a *complete* hierarchy of *H-elements*, connected by *links* which represent light exchanges. The hierarchy is complete in the sense that its highest level encompasses the entire scene. H-elements can be of different nature, such as clusters (i.e., groups of objects), or portions of surfaces, but they all share a number of properties and characteristics, allowing the specification of many computational operations at an abstract level. H-elements all possess radiosity and reflectance information, as well as a set of links representing

energy received at their level of the hierarchy. Specialised H-elements dealing with the specifics of clusters, or polygons, are obtained by subclassing the abstract level and overloading appropriate functionality.

3.2 Computational Operators

In the case of our clustering approach, input objects (e.g., surfaces, meshes etc.) are first grouped into clusters, and an initial self-link of the root node to itself is established. This link is a (very coarse) representation of all energy transfers in the scene. To achieve the desired quality of lighting simulation, the following sequence of operations needs to be performed [17, 2, 18].

1. Links between H-elements are first refined using a *refinement engine*, resulting in the subdivision of the corresponding H-elements where appropriate. This is described in more detail in the following paragraph.
2. Once the refinement process is complete, links exist between H-elements at various levels. Irradiance is now transported (*gathered*) across these links.
3. To maintain a consistent view of radiosity at every level of the hierarchy, irradiance is pushed (added) down the hierarchy, and radiosity is pulled (averaged) up.
4. After an appropriate number of iterations of the refine-gather-push/pull operations, the subdivided H-elements are displayed using the computed radiosity values.

Each one of these steps is defined at the highest possible, and thus most abstract, level of the class hierarchy. For operations such as gather and push-pull, few parts of the process require the definition of specialised methods.

3.3 Hierarchical Refinement Engine

The refinement engine (or "refiner") is the core of the hierarchical lighting simulation process. The refiner operates on links between two hierarchical elements.

The refiner first estimates how well the link represents the light transfer in question. This is typically performed by an error based criterion with respect to a user-defined tolerance. An example is the "BF" (radiosity times form-factor) criterion [6] Improved approaches exist [10, 5], using upper and lower bounds on the energy transfer across the link. If the link is judged of insufficient quality, it is refined, and thus one of the H-elements is typically subdivided, resulting in the creation of *sub-links*.

The estimation of the error criteria, and the subsequent gather operation on a link requires the calculation of an estimate of the form-factor between two H-elements or the irradiance received on an element due to the illumination from the other. This in turn requires an appropriate sampling of a kernel function involving the position, orientation, distance and relative visibility of the two elements. As shown in [18] the calculation of form factors can be expressed at an abstract level using virtual functions with specialised behaviour for clusters or surfaces. A typical refiner therefore operates solely on H-elements and relies on virtual functions to perform the necessary subdivision (split).

In what follows, we show how the treatment of generic complex meshes and natural lighting can be integrated into this unified framework.

4 Geometry

4.1 Dealing with Geometric Complexity of Outdoor Scenes

For many outdoor scenes, we are presented with complex mesh representations (terrains, large man-made structures etc.). A simple, direct solution would be to pass these elements in an unstructured manner to a clustering algorithm [18]. As mentioned previously, this could result in inappropriate groupings of surfaces since the implicit mesh structure and connectivity information is lost. In addition, this would encumber the system with a large total number of polygons, even in the case where a vast majority are never lit or never participate in lighting.

We present a solution to this problem in the form of a general structure which encapsulates hierarchical mesh information. The power of this structure is its ability to hide the complexity of the geometric object (e.g., a terrain mesh or a tensor-product spline surface), by presenting a generic hierarchical interface. As hierarchical subdivision becomes necessary, the structure uses the hidden geometric complexity to create finer hierarchically subdivided levels. As an additional benefit, at the lowest level of subdivision the structure reverts to classical regular subdivision (e.g., quadtrees), with all the advantages this implies in terms of implicit representation and information sharing.

4.2 A Generic Hierarchical Mesh Structure

The definition of a generic hierarchical mesh class may include information permitting subdivision up to a "finest" level. For example, a terrain mesh will be given as an array of vertices describing its finest level of detail. Our generic interface hides this complex geometric information in what concerns the lighting simulation.

In our generic structure no mesh element uses all the geometric information (e.g., vertex array for terrains) constituting the mesh, since each element is approximated using only a small subset. Nevertheless each element has access to the geometric detail data. When subdivided, a mesh element at a given level constructs child elements which, by using more detailed geometric information, are a better local approximation of the mesh.

After each consecutive split, more geometric detail is used by each child element leading to a stage where finer geometric detail is no longer available in the original data. To allow for finer detail the splitting process does not construct additional mesh objects but regular hierarchical polygon elements (quadtrees) instead. This is possible in the context of our uniform framework because meshes and polygons are both derived from a single abstract hierarchical surface class.

Generic Surface Operations To successfully integrate generic mesh operations into our hierarchical radiosity system, we need to be able to maintain all connectivity information at every level of the hierarchy. In addition, the regular (e.g., quadtree or triangular) structures created below the finest level of mesh subdivision must be able to find their neighbours across boundaries. An example is shown in the case of quadtrees in Fig. 1, although a similar problem exists for smooth surfaces etc.

To deal with this problem, we define an abstract class *generic edge*, allowing us 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*, e.g. the

mapping of the grid of terrain point data to the interval $[0, \ 1]^2$. Neighbourhood information is most notably needed to correctly perform per-vertex shading (Gouraud shading), and to correctly maintain a restricted subdivision when splitting elements.

A generic edge is defined by two points in parameter space which we call its *support* and parameters on this support defining the beginning and end of the generic edge. During mesh element subdivision, generic edges are defined between the newly constructed children.
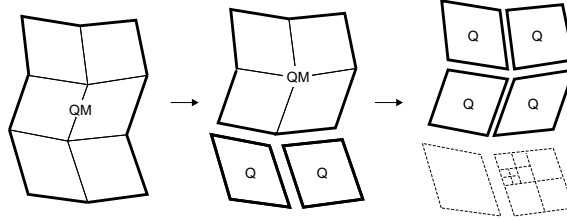

Fig. 1. A sequence of splitting operations, starting with one quad mesh (QM) and resulting in quadrilaterals (Q), which are subdivided into regular quadtrees. Connectivity is preserved across boundaries at *each* level of the hierarchy, due to the generic edge structure.

Neighbour Finding and T-vertex Detection Given the generic edge structure, neighbour finding can be correctly performed using the hierarchical mesh representation. Consider a patch with a generic edge e consisting of the support s and the parameter interval $[t_0 : t_1]$. To find the appropriate neighbouring patch sharing e, the following steps are performed:

1. Step up the mesh element hierarchy until the current element is using the edge e_0, consisting of s and the parameters zero and one.
2. There must be a sibling of that patch sharing e_0, which would not have been constructed otherwise. Find this sibling.
3. Step down the sibling's hierarchy to find the smallest element sharing the entire interval $[t_0 : t_1]$ of s.

Given this neighbourhood finding process, we can easily perform mesh restriction and T-vertex identification [15], thus allowing correct colour interpolation across vertices at different levels in the hierarchical mesh.

5 Natural Lighting

The incorporation of natural light sources is of course a key requirement for the creation of realistic simulations in outdoor scenes. The first problem to address is modeling these sources. For outdoor scenes with moderate scale (such as a house or some part of a city) the sun can be modeled as a parallel source, and the sky is considered a hemispherical source at infinity.

In order to fully integrate sunlight and skylight in a hierarchical radiosity process with clustering, these light sources must become integral hierarchical elements which

provide the same interface as the other elements in the simulation [18]. Thus we need to define corresponding H-elements (using specialised classes) which can "split" (be subdivided) to create child elements which represent the source in more detail and with less approximation error.

In addition, particular attention must be paid to the sampling mechanisms used for these sources, for the calculation of energy transfer and form factor estimates. For instance, form factor formulas based on distance calculations cannot be used directly with sources placed at infinity.

5.1 Hierarchical Representation of Skylight

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 the horizon, and v denotes the angle in the plane west from south (see Fig. 2). The initial sky patch is the whole hemisphere, $(u, v) \in [0, \pi/2] \times [0, 2\pi]$. Every hierarchical sky patch is therefore defined by a parameter range, and the constant radiance value associated with that range. Note that sky patches are characterised by *radiance* as opposed to radiosity, since they have no area and are positioned at an infinite distance.

The radiance is computed by sampling any given skylight distribution at a fine level of detail and creating a pyramidal representation of the averages of the sampled values. If an element is split, it retrieves the associated radiance value from the pyramid. If an element splits below the finest level of the pyramid, a new value is obtained by subsampling.

The skylight intensity values can also be modulated by a texture. The texture value is simply applied every time the skylight is sampled. When using monochromatic skylight models like the one provided by *Radiance* [20], the texture can be used to assign colours and draw clouds which will affect the simulation results.

Assuming full visibility, the contribution of a sky patch q with parameter range $[u_0, u_1] \times [v_0, v_1]$ to the irradiance on a given scene element p is obtained as follows:

$$\delta L_i^{(p)} = \mathcal{R}_{p,q} \mathcal{S}_p \Omega_q J_q \tag{1}$$

where $\mathcal{R}_{p,q}$ is p's *receiver factor*, $\mathcal{S}_p$ is p's *scale factor*, Ω_q is the solid angle subtended by q when viewed from the any point of the scene, and J denotes radiance. $\mathcal{R}$ and $\mathcal{S}$ depend on p's element type: For surfaces, $\mathcal{R}$ is the cosine between the surface normal and the line through the two elements' samples and $\mathcal{S}$ is 1. For volumes and clusters, $\mathcal{R}$ is 1 and $\mathcal{S}$ is $1/4$ [18]. In the case of partial visibility, a multiplicative correction factor is applied to model the fraction of mutual visibility [6].

The solid angle Ω_q is given by

$$\Omega_p = (v_1 - v_0) \cdot (\sin u_1 - \sin u_0). \tag{2}$$

5.2 Parallel Sources and Sunlight

Parallel sources are easy to implement because they never need to be split, and the irradiance cast on an element is just the source's flux density modified by the receiver's orientation and element type (receiver and scale factor). We thus obtain the irradiance of a parallel light q on a scene element p as follows (again assuming full visibility):

$$\delta L_i^{(p)} = \mathcal{R}_{p,q} \mathcal{S}_p \phi_q \tag{3}$$

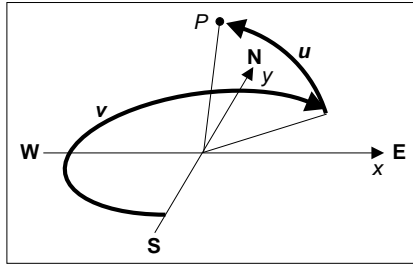where ϕ_q denotes the source's flux density.

Fig. 2. Parameter space for a skylight object with u and v denoting altitude and azimuth
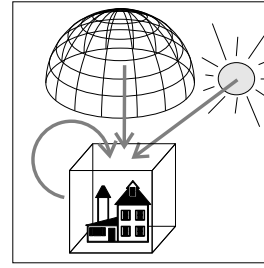

Fig. 3. Initial links for a scene with a skylight and a sunlight object

5.3 Adapting the Refinement Process

We refer to sunlight and skylight as "external" light sources because they are placed outside the scene geometry: therefore they are not included in the extent of the top-level cluster of the scene hierarchy, and their effect is not represented by the top-level "self-link" of the root cluster. To take into account the effect of external light sources, at the beginning of the simulation one link from each external light source to the scene's root cluster is created (in addition to the top-level self-link). In the presence of n external lights, the simulation would thus start with $n + 1$ initial links. At this stage, the link from the skylight is established from the toplevel sky description. Figure 3 shows an initial link setup for a scene with sunlight and skylight.

Next, a refiner is called to recursively examine and subdivide these links. Because a refiner typically uses generic sampling routines to query the geometry of the elements considered, some modifications are needed to properly cope with external light sources. In particular, for special sources such as sunlight and skylight, the estimation and computation of irradiance must be left to the sender. In this way we can ensure that for ordinary surface-to-surface links the well-established form factor formulae are used to compute irradiance, and for external sources our new formulae are applied. Thus, the refinement engine does not need to know the type of the objects interacting with each other and uses a generic refinement procedure.

6 Results

The methods for hierarchical treatment of natural light sources and of terrain meshes have been implemented in BRIGHT, a hierarchical radiosity system with clustering, using a BF refinement engine. The skylight intensities and colours were computed using parts of the *Radiance* software system by Greg Ward [20] and modulating these monochromatic values by a texture. Any other skylight model could be used instead. All timings are on a Silicon Graphics Indy R5000 (150Mhz) machine.

6.1 Terrain

Figure 4 shows a terrain mesh illuminated by a sky dome and a parallel sunlight in the morning of a day in August. The radiosity solution for Figure 4 took about 16 minutes. The terrain scene is built from 30,373 polygons. The hierarchical subdivision resulted in 90,899 links and 33,967 leaf elements.

6.2 Natural Lighting

Figure 5 shows a house and several trees on a smaller terrain mesh. The total number of input polygons is 20,260. The scene is illuminated only by a hierarchical skylight object. The house casts (soft) shadows which are caused by the brighter parts of the sky.

If the same scene is also lit by a parallel sunlight source as in Figure 6, the shadows become sharper and the trees begin to cast shadows on the house walls and the ground. Fig. 7 shows the same scene from a different direction. Note the bright area in the sky which produced the subtle shadows of Fig. 5.

The hierarchical solution for Fig. 5–7 resulted in around 49,017 links and required approximately ten minutes. There are a total of 64,674 leaf elements. By selecting a higher error tolerance, our hierarchical algorithm can compute a coarser approximation of the natural illumination in the same scene in 148 seconds, using a total of only 1800 links (see Fig. 8) and resulting in less than 30,000 leaf elements, but maintaining comparatively good visual quality.

It is interesting to visualise the hierarchical structure of the solution by displaying the links to different levels of the hierarchy. Sky patches with low intensity are linked quite high up in the hierarchy (Fig. 9).

7 Conclusions

A new *hierarchical* solution to the problems of efficient and accurate outdoor lighting has been presented. A hierarchical abstraction was introduced, permitting the integration of moderately complex outdoor objects (such as terrain maps) and natural lighting (sky- and sun-light) into a unified hierarchical illumination framework.

In particular, an object-oriented approach was introduced to treat complex mesh structures using generic operators. The problem of hierarchically sampling the sky dome was then addressed and an appropriate refinement strategy described. These two contributions allow the treatment of complex, naturally-lit outdoor scenes, based on clustering and hierarchical radiosity, resulting in efficient and accurate lighting simulations.

In contrast to the only previous radiosity-based approach to outdoor lighting [12], our new method requires computation times for scenes of comparable size which are lower than those reported in [12]. More importantly, the hierarchical nature of our new algorithm ensures that global illumination effects are always accounted for at the chosen level of approximation, which is not possible in the progressive-refinement approaches previously used.

In future work, we will be investigating the use of the generic mesh framework for general complex mesh structures, which will require the correct definition of the parameterisation for non-regular structures. In terms of natural lighting, the sampling and refinement processes can be greatly improved, in the spirit of the error-based approaches of [11] or [5]. This will result in higher quality images with a smaller number of links and consequently lower computation times.

References

1. CIE. Standardization of luminance distribution of clear skies, 1973. Publication TC 4.2.

2. Michael F. Cohen and John R. Wallace. *Radiosity and Realistic Image Synthesis.* Academic Press Professional, San Diego, CA, 1993.

3. Y. Dobashi, K. Kaneda, T. Nakashima, H. Yamashita, T. Nishita, and K. Tadamura. Skylight for interior lighting design. In *Computer Graphics Forum*, volume 13, pages 85–96. Eurographics, Basil Blackwell Ltd, 1994. Eurographics '94 Conference issue.

4. Matthias Eck, Tony DeRose, Tom Duchamp, Hugues Hoppe, Michael Lounsbery, and Werner Stuetzle. Multiresolution analysis of arbitrary meshes. In R. Cook, editor, *SIGGRAPH 95*, Annual Conference Series, pages 173–182. ACM SIGGRAPH, Addison Wesley, 1995. held in Los Angeles, California, 6-11 August 1995.

5. S. Gibson and R. J. Hubbold. Efficient hierarchical refinement and clustering for radiosity in complex environements. *Computer Graphics Forum*, 15(5):297–310, December 1996.

6. Pat Hanrahan, David Saltzman, and Larry Aupperle. A rapid hierarchical radiosity algorithm. *Computer Graphics*, 25(4):197–206, August 1991. Proceedings SIGGRAPH '91 in Las Vegas (USA).

7. Kazufumi Kaneda, Takashi Okamoto, Eihachiro Nakame, and Tomoyuki Nishita. Photorealistic image synthesis for outdoor scenery under various atmospheric conditions. *The Visual Computer*, 7:247–258, 1991.

8. R. Victor Klassen. Modeling the effect of the atmosphere on light. *ACM Transactions on Graphics*, 6(3):215–237, July 1987.

9. Eric Languénou and Pierre Tellier. Including physical light sources and daylight in a global illumination model. *3rd Eurographics Workshop on Rendering*, pages 217–225, May 1992.

10. Dani Lischinski. *Accurate and Reliable Algorithms for Global Illumination.* Ph.D. thesis, Cornell University, Ithaca, NY, 1994.

11. Dani Lischinski, Brian Smits, and Donald P. Greenberg. Bounds and error estimates for radiosity. In Andrew Glassner, editor, *Proceedings of SIGGRAPH '94 (Orlando, Florida, July 24–29, 1994)*, Computer Graphics Proceedings, Annual Conference Series, pages 67–74. ACM SIGGRAPH, ACM Press, July 1994. ISBN 0-89791-667-0.

12. Stefan Müller, Wolfram Kresse, and Frank Schoeffel. A radiosity approach for the simulation of daylight. In *Eurographics Rendering Workshop 1995*. Eurographics, June 1995.

13. T. Nishita and E. Nakamae. Continuous tone representation of three-dimesional objects illuminated by skylight. *Computer Graphics*, 20(4):125–132, August 1986. Proceedings SIGGRAPH '86 in Dallas (USA).

14. J. Rossignac and P. Borrel. Multi-resolution 3D approximation for rendering complex scenes. In *Second Conference on Geometric Modelling in Computer Graphics*, pages 453–465, June 1993. Genova, Italy.

15. Hanan Samet. *Applications of Spatial Data Structures.* Addison-Wesley, Reading, Massachusetts, 1990.

16. François Sillion and George Drettakis. Feature-based control of visibility error: A multi-resolution clustering algorithm for global illumination. In Robert Cook, editor, *SIGGRAPH 95 Conference Proceedings*, Annual Conference Series, pages 145–152. ACM SIGGRAPH, Addison Wesley, August 1995.

17. François Sillion and Claude Puech. *Radiosity and Global Illumination.* Morgan Kaufmann, San Francisco, 1994.

18. François X. Sillion. A unified hierarchical algorithm for global illumination with scattering volumes and object clusters. *IEEE Transactions on Visualization and Computer Graphics*, 1(3):240–254, September 1995. ISSN 1077-2626.

19. Katsumi Tadamura, Eihachiro Nakamae, Kazufumi Kaneda, Masshi Baba, Hideo Yamashita, and Tomoyuki Nishita. Modeling of skylight and rendering of outdoor scenes. In R. J. Hubbold and R. Juan, editors, *Eurographics '93*, pages 189–200. Blackwell, 1993.

20. Gregory J. Ward. The RADIANCE lighting simulation and rendering system. In Andrew Glassner, editor, *Proc. SIGGRAPH '94 (Orlando, Florida, July 24–29, 1994)*, Computer Graphics Proceedings, Annual Conference Series, pages 459–472. ACM SIGGRAPH.
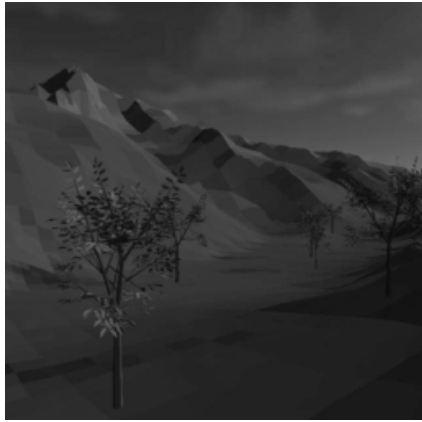
Fig. 4. Terrain, lit by sky-dome and sun.

Fig. 5. House, lit by sky-dome only.

Fig. 6. House lit by sky-dome and sun.

Fig. 7. House from left, view towards the sun.
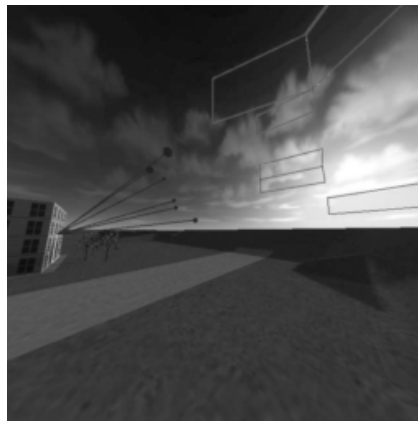

Fig. 8. House, computed with less detail.

Fig. 9. Entire wall linked to large sky patches.