A Hierarchical Algorithm for Radiosity Daylighting

Gonzalo Besuievsky
Ignacio Martin
Institut d'Informática i Aplications
Universitat de Girona

Abstract

A new daylight simulation method for radiosity environments is presented. Daylight illumination is transported in a first step where sun and sky are represented as a set of directional sources. Light is transported using parallel projections of these sources. The scene mesh is adaptively subdivided using energy and visibility criteria obtained from a extended scan-line algorithm that computes a layered depth image for each projection. In a second step, diffuse intereflection is computed through a hierarchical Monte Carlo radiosity algorithm. The method is feasible for it extension for environments containing specular surfaces.

Relative short time view-independent global illumination results can be obtained with the proposed method, which make it suitable for daylight building design and analysis applications for both indoors and outdoors environments.

1 Introduction

Daylight illumination is very important for realism simulation. First of all for increasing the degree of realism in diurnal scenes for classic Image Synthesis applications like communication movies and entertainment. Daylight simulation is also required in building design. The last few years experienced an increase need for architecture design tools concerning with both visual and thermal comfort as well as with lighting assessment and energy saving. Solving the daylight illumination problem efficiently is still a challenge for global illumination. The complexity of the problem is due from the fact that light energy comes from a huge extended area source (the whole hemisphere), and every point in the scene is potentially a receiver. Moreover, the increasingly size of the scenes which may have different kinds of openings, combining reflective, refractive or diffuse materials, makes daylight illumination even more complex. The main problems that daylight available methods present are that they are not general enough to deal with complex lighting situations and that they spend long rendering times for they application in an architectural production design cycle.

In this paper we present a new method for daylight simulation for radiosity environments. The method adaptively computes daylight energy into a finite element mesh obtaining, within a short processing time, a view-independent solution of the simulation. This process is based on an extended Monte Carlo hierarchical radiosity algorithm and it is feasible to be extended for environments containing specular surfaces.

We first overview previous work in daylight simulation and sky models in section 2. Our method is outlined in section 3 and described in sections 4 and 5. The radiosity algorithm used and a proposed final gather algorithm for high-quality rendering is described in section 6, results are presented in section 7 and conclusions and future work in section 8.

2 Previous Work

There are two main topics in daylight simulation. First, the modeling of the sky, and second the use of these models to compute global illumination solutions. There has been a lot of research in sky modeling, and there is a set of well known models widely accepted. However, there has been a limited effort to develop efficient and accurate daylight simulation techniques that can deal with complex environ-

ments.

Models for skylight have been studied in many fields over decades as they are required to aid in simulations for many applications. Several available models are compared in [6] and their effects of daylight distribution prediction. For rendering applications we need a function that returns the spectral radiance given the following inputs: direction, geographical location, date, time and conditions. An interesting analytic sky model is presented in [13]. It is accurate, easy to use in rendering algorithms and can handle smoothly varying atmospheric conditions. Another posible solution instead of analytical sky models is the use of captured high dynamic range images (HDRi) [5].

Less effort has been made in using daylight for global illumination, specially in finite element methods. First approach for indoors scenes was presented in [11]. In [7] daylight illumination is preprocessed for a set of "window" surfaces that are used as area light sources. The method is restricted for indoor scenes. A more general approach is presented by Müller et al. [8] through a progressive refinement radiosity algorithm. The sky dome is subdivided into a fixed number of patches, and a daylight shooting step is performed as a pre-processing pass. Hierarchical radiosity was extended for they use in daylight in [4]. Here, the sky dome is subdivided into a set of hierarchical patches that interact with the scene as if they were regular patches. More recently, Ashdown proposed the use of a combination of a parallel projection approach and a ray tracing technique to compute sun and sky illumination [2]. This is the closer work to the method proposed here. Ray tracing-based approaches were also proposed for daylighting. The well-known Radiance system [16] is capable of producing accurate lighting predictions for complex environments. However, it requires user intervention to identify important sources and also a considerable amount of time of processing.

3 Overview of the Method

Our method consist in the following steps:

- Adaptively subdivide the sky dome into spherical triangles. Each triangle will be represented as a parallel light source using its center. The sun, because of the important amount of power carried, is considered as another parallel light source.
- For each parallel light source:
  - Scan convert the mesh using an orthographic camera corresponding to the direction of the parallel light source.
  - Transport energy to diffuse patches. Hierarchically subdivide the mesh taking into account energy and visibility criteria.

This process ends with a hierarchical subdivision of the scene that accurately captures the direct illumination produced by the sun and the sky. Receiver patches are then converted into diffuse light sources and a hierarchical radiosity solver iterates to obtain the final solution. The following sections describe details of each of these steps.

4 Sky Hemisphere Subdivision

For the sky model we used the model presented in [13], but the method herein described can be adapted to any sky model that returns the incoming radiance for a given direction. The goal of this step is to produce a set of parallel light sources that approximate the sky illumination distribution. Advanced subdivision techniques, developped mainly for HDRi, like the structured importance sampling algorithm ([1]) could also be adapted for our daylight algorithm.

We consider the sky hemisphere initially subdivided into four spherical triangles, and then each triangle is considered for subdivision. The criterion for subdivision uses two parameters to drive the subdivision: the solid angle subtended by the triangle and its irradiance. A minimum solid angle for the spherical triangles as well as a maximum irradiance thresholds can be selected. The irradiance is computed by averaging the radiance at their vertices and its center.

These simple algorithm criterion produces a higher density of triangles in the brighter areas of the sky. The sun is considered as another parallel light source, and the direction used corresponds to the center of the sun. Figure 1 shows a top view of the final spherical triangle subdivision for a sky generated for some given conditions. The irradiance

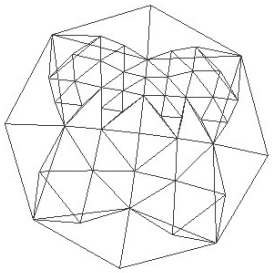

Figure 1: Sky mesh triangles generated at 16:00 GMT, lat. 41, long. 2, 20th June. Using the adaptive refinement algorithm, for this conditions, 70 triangles where generated. Higher density can be appreciated in the bright area of the sky.

that produces each parallel source corresponding to a triangle i in a visible surface is:

$$E_i = \widehat{L}_i \omega_i \quad (1)$$

where $\widehat{L}_i$ is the average radiance over triangle i, and ω_i is the solid angle subtended by triangle i.

5 Parallel Light Source Transport

Once the sky has been modeled as a set of parallel light sources, energy from each light and from the sun light has to be transported into the scene. Parallel light transport computation were also been used for global illumination in [15] and [10]. In these cases it was used to compute diffuse interreflection exploiting coherence. We use parallel projections only for the sky source. We describe each step of the procedure for a single parallel light source.

5.1 Parallel Projection

Parallel projection is used to compute the visibility of the scene patches with respect to the source. All the top-level patches of the hierarchy are projected using a scan-line algorithm. At each pixel, all the visible patches along the corresponding line of sight are stored depth-ordered. The final result is a Layered Depth Image [14] that it is used to drive the hierarchical subdivision of the scene for the current parallel light source.

For each parallel projection we first apply the corresponding transformation in order to cover the whole scene into the resulted viewport. The resolution of the projection is proportional to the irradiance of the parallel light source.

5.2 Refinement Subdivision Criteria

Once all the top-level patches of the hierarchy have been projected the refinement of the mesh takes place. Because a parallel light source produces a constant illumination on a planar surface, the refinement is only driven by visibility. A top level patch that is fully visible does not need to be further subdivided.

The LDI provides the information to compute the visibility for each patch. For each patch we can compute:

- The total number of pixels in which the patch is projected (visible or not): n_p. In this case the patch is intersected by the line of sight of the pixel at any depth in the LDI.
- The total number of pixels in which the patch is visible: n_v. In this case the patch is intersected first by the line of sight of the pixel.

We define the *Visibility Ratio*, VR, as the ratio between n_v and n_p. This ratio, that ranges between 1 (patch completely visible) and 0 (patch completely occluded), gives us an approximation of the visibility of the patch and it is used to decide the patches subdivision in the projection. We used a simple criterion: a patch i is refined if $V_{min} < VR_i < V_{max}$. In our empiric tests we have found that values of $V_{min} = 0.05$ and $V_{max} = 0.95$ produces reasonable results. A simplified example of an LDI and the visibility ratio computation is described in Figure 2.

When a patch is refined, first is subdivided into four children (quad-tree subdivision), and each children is projected into the LDI. This projection is not an expensive operation since the four child represent the same surface that the parent, so only the four children have to be scan converted. When a child is projected, at each pixel of the projection we have to search for the parent in the list of patches of the pixel. Then, the parent is exchanged by the corresponding child.

Figure 2 shows an example of patch subdivision. Using the information provided by the LDI, the visibility ratio of patch $P2$ is $VR_2 = 0.6$, so it has to be subdivided. Two children are created: $P21$ and $P22$. The two children together represent exactly the same surface that patch $P2$, so they are scan converted into the same pixels that $P2$. The LDI on the bottom shows the result after subdivision. Patch $P2$ no longer appears in the LDI, and its pixels have now patches $P21$ and $P22$. According to the new LDI, the visibility ratio of the subdivided pixels are $VR_{21} = 0.\widehat{3}$ and $VR_{22} = 1$.

This procedure is applied recursively to the children, until all of them have an acceptable visibility ratio, or a minimum area threshold is reached.

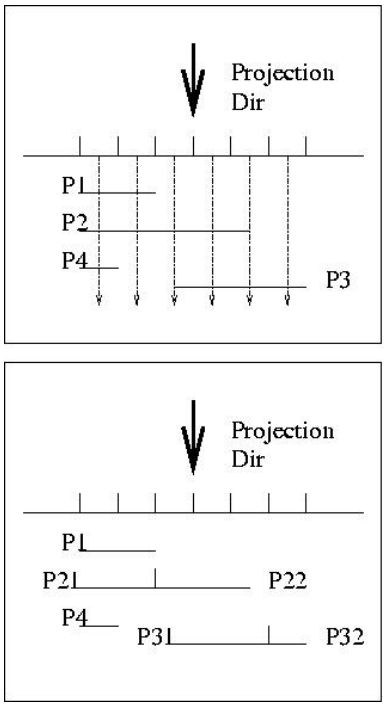

Figure 2: Example of patch subdivision. Patch $P1$ has been projected into 2 pixels, both visible, so its visibility ratio is $VR_1 = 1$. Patch $P2$ has been projected into 5 pixels, 3 visible, and 2 occluded, so its visibility ratio is $VR_2 = 0.6$. Patch $P3$ has been projected into 3 pixels, 1 visible and 2 occluded, so its visibility ratio is $VR_3 = 0.\widehat{3}$. Finally, patch $P4$ has been projected into 1 pixel, which is occluded, so its visibility ratio is $VR_4 = 0$.

5.3 Diffuse Transport Computation

Once the hierarchical refinement has finished, energy is transferred to the patches. Only visible patches that appear in the LDI will receive energy. That means that the energy transfer is performed at different levels in the hierarchy depending on how patches have been subdivided in the LDI.

The radiosity of patch i is updated as follows:

$$B_i += \rho_i E \frac{Ap_i}{A_i} VR_i$$

where ρ_i is the reflectivity of patch i, E is the irradiance corresponding to the parallel light source (equation 1), Ap_i is the projected area of the patch, A_i is the area of the patch, and VR_i is the visibility ratio defined in previous section. Note that the term $\frac{Ap_i}{A_i}$ accounts implicitly for the cosine factor.

6 Hierarchical Radiosity and Visualization

Once direct illumination has been computed all receiver patches with incoming power are used to start a hierarchical radiosity solution. We used the hierarchical Monte Carlo radiosity algorithm [3], which can deal well with moderate complex environments as does not require links storages. Another advantage of this algorithm is that can be extended for accounting for specular surfaces and participating media [12].

The algorithm begins building a shooting-patch list. If any light source is included in the scene description it is added to the list at this step and integrated to the simulation process without any restriction. The algorithm iterates the solution using classical refinements mesh criteria (minimum patch area and minimum power transport) until a global convergence is reached.

Our method, as any radiosity algorithm, provide view-independent results. Walkthroughs in daylight simulated scenarios can be performed in real time with our navigator or can be exported in a standard exchange format as vrml.

6.1 Final Gathering

In order to obtain high-quality render images including hard shadows and textures we also developed a final gathering algorithm that recomputes the

illumination per pixel. This algorithm is similar to the one described in [12] but extended for daylighting.

For a given view, for each pixel of a ray-traced image, energy coming from all patches is re-sampled. The algorithm builds a per-pixel cumulative density function (cdf) that uses more samples for the potentially most contributing patches.

Direct illumination is re-computed at each pixel and daylight simulation results is used to compute the indirect component of the global illumination. For the sun source a single ray is cast in the sun direction. The energy coming from the sky is not re-sampled as it produce a low-variance illumination that can be approximated with the mesh information. For this purpose energy coming from the sky is stored separately at each patch and interpolated in the final gather.

7 Results and Discussion

We implemented the present method within the SIR system, a framework for the development of global illumination algorithms [9]. All timings showed in this section are measured in a Pentium 4 with 1 Gb RAM. Our method was tested for validation with two scenarios: a simple model with walls and floor and a six-floor building composed by 12k polygons. Two daylight design application results are also shown: an underground metro station composed by 7k polygons and a kitchen.

In order to test the daylight parallel projecting algorithm we used a simple model composed only by walls with openings (232 triangles). Table 1 shows execution details for different hours of the scene and figure 3 shows the resulted patch-mesh after each projection step. This step allows to deal efficiently with the visibility occlusion problem for direct daylighting. The illuminated hierarchical mesh is used for starting the hierarchical radiosity solution. Parallel projections for daylighting were also used in [2]. The difference between the approaches is that the referenced one used hardware hemicube for the projections requiring then a ray-tracing algorithm to recover aliasing problems, whereas we uses a adaptive scan-line algorithm.

We tested the whole global illumination algorithm for a six-floor building with furniture. Figure 4 shows resulted images for an afternoon simulation. The first row shows the mesh after the sun shoot and the sky shoot, the second row shows different views for the converged radiosity solution. Time processing for each step were: 2 seconds for the sun shoot, 30 seconds for the sky shoot and 69 for the hierarchical radiosity solution. We observe relative short times processing to obtain both the direct daylighting and the globally-illuminated simulation solution. This is a remarkable advantage of the algorithm comparing it to previous work.

	8:00	9:00	19:00
# of sky triangles	70	58	52
# of patches after sun shoot	4288	5044	2458
# of patches after sky shoot	4672	5482	2968
Time for sun shoot (sec.)	1	1	1
Time for sky shoot (sec.)	7	6	6

Table 1: Details executions of the simple model of figure 3. Two hours in the morning and one in the afternoon were taken for the following conditions: 20th June, latitude 41, longitude 2.

Figure 5 shows the model and simulation results of the interior of an underground metro station illuminated with both sun and sky daylight. Openings are designed with a diffuse devices at the ceiling of the building, and they are the only sources of illumination. For this reason, this is a complex situation. Our method works well for this case without any preprocessing in order to detect openings. Simulation times for these scene are 96 seconds for the daylighting and the HMC radiosity steps and 754 seconds for the final gathering step described in section 6.1 for a 600x600 size image.

Figure 6 shows simulation results for a kitchen design that consists of 25K triangles and includes textures. It has been computed with the following settings: latitude 35, longitude 0, January 1st, 7:30 GMT. From left to right it shows the scene after sun shoot, sky shoot, Hierarchical Monte Carlo convergence and the final gather result using the algorithm described in section 6.1. The timing for these executions are: 120 seconds for the whole daylight shoot (sun and sky), 245 seconds for the HMCR and 1230 seconds for the final gathering for a 600x600 size image.

8 Conclusions and Future Work

A new daylighting method for radiosity environments has been presented. The method performs a fast projection that shoots sun and sky energy into the scene creating a hierarchical mesh that accurately captures the daylight. A hierarchical Monte Carlo radiosity algorithm is then performed to obtain view-independent daylight simulation solution in relative short times. A final gathering step, that allows to generate high quality images is also proposed for the method. We belive that the proposed method is useful for daylight visualization application in early design building where a rapid prototype must be evaluated.

Future work on this project rely on several improvements of the method:

- Use coherence for similar sky projections for daylight simulation at several conditions. For example, compute daylight for several times using the same projections.
- Extend the method for including specular surfaces using the the extended hierarchical Monte Carlo radiosity presented in [12] in combination with the final gathering algorithm presented in section 6.1.

Acknowledges

This work was supported by TIN2004-07672-C03-00 project, dursi project PICS2005-12 and by the "Ramon y Cajal" program. Thanks to Mauricio Molochnik for the metro station model.

References

[1] Sameer Agarwal. Structured Importance Sampling of Environment Maps. In *Computer Graphics Proceedings, Annual Conference Series, 1995 (ACM SIGGRAPH '03 Proceedings)*, pages 605–612, 2003.

[2] Ian Ashdown. Fast daylight simulation and analysis. In *Proceedings of Illuminating Engineering Society 2002 Annual Conference (IESNA '02)*, 17th Floor, 120 Wall Street, New York, NY, August 2002. Illuminating Engineering Society of North America. 177–186.

[3] Philippe Bekaert, László Neumann, Attila Neumann, Mateu Sbert, and Yves D. Willems. Hierarchical monte carlo radiosity. In G. Drettakis and N. Max, editors, *Rendering Techniques '98 (Proceedings of Eurographics Rendering Workshop '98)*, pages 259–268, New York, NY, 1998. Springer Wien.

[4] Katja Daubert, Hartmut Schirmacher, François Sillion, and George Drettakis. Hierarchical lighting simulation for outdoor scenes. In Julie Dorsey and Philipp Slusallek, editors, *Rendering Techniques '97 (Proceedings of the Eighth Eurographics Workshop on Rendering)*, pages 229–238, New York, NY, 1997. Springer Wien. ISBN 3-211-83001-4.

[5] Paul Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Computer Graphics (ACM SIGGRAPH '98 Proceedings)*, pages 189–198, 1998.

[6] K. P. Lam, A. Mahdavi, and V. Pal. The implications of sky model selection for the prediction of daylight distribution in architectural spaces. In J. D. Spitler and J. L. M. Hensen, editors, *Building Simulation '97*, volume I, pages 159–162, Prague, Czech Republic, September 1997. International Building Performance Simulation Association.

[7] Eric Languénou and Pierre Tellier. Including Physical Light Sources and Daylight in Global Illumination. In *Third Eurographics Workshop on Rendering*, pages 217–225, Bristol, UK, May 1992.

[8] Stefan Muller, Wolfram Kresse, Neil Gatenby, and Frank Schoeffel. A Radiosity Approach for the Simulation of Daylight. In P. M. Hanrahan and W. Purgathofer, editors, *Rendering Techniques '95 (Proceedings of the Sixth Eurographics Workshop on Rendering)*, pages 137–146, New York, NY, 1995. Springer Wien.

[9] Ignacio Martín, Frederic Pérez, and Xavier Pueyo. The SIR rendering architecture. *Computers & Graphics*, 22(5):601–609, 1998.

[10] László Neumann. Monte Carlo Radiosity. *Computing*, 55(1):23–42, 1995.

[11] Tomoyuki Nishita and Eihachiro Nakamae. Continuous Tone Representation of Three-Dimensional Objects Illuminated by Sky Light. In *Computer Graphics (ACM SIGGRAPH '86 Proceedings)*, volume 20, pages 125–132, August 1986.

[12] Frederic Pérez, Ignacio Martín, and Xavier Pueyo. High quality final gathering for hierarchical monte carlo radiosity for general environments. In J. Vince and R. Earnshaw, editors, *Advances in Modelling, Animation and Rendering (Proceedings of Computer Graphics International 2002)*, pages 425–437. Springer, 2002.

[13] Arcot J. Preetham, Peter Shirley, and Brian E. Smits. A practical analytic model for daylight. In *Computer Graphics Proceedings, Annual Conference Series (Proc. SIGGRAPH '99)*, pages 91–100, August 1999.

[14] J. Shade, S. Gortler, L. He, and R. Szeliski. Layered Depth Images. In *Computer Graphics Proceedings, Annual Conference Series, (Proc. SIGGRAPH '98)*, pages 231–242, July 1998.

[15] László Szirmay-Kalos, Tibor Foris, László Neumann, and Balázs Csébfalvi. An analysis of quasi-monte carlo integration applied to the transillumination radiosity method. *Computer Graphics Forum (Eurographics '97 Proceedings)*, 16(3), 1997. C271–C281.

[16] Gregory J. Ward. The RADIANCE Lighting Simulation and Rendering System. In *Computer Graphics Proceedings, Annual Conference Series, 1994 (ACM SIGGRAPH '94 Proceedings)*, pages 459–472, 1994.

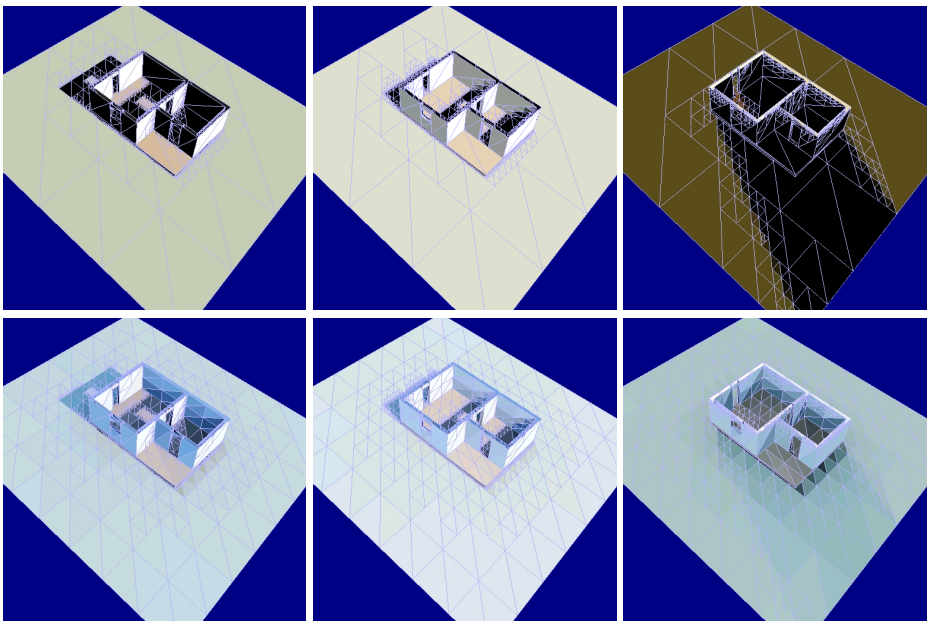

Figure 3: Results of the parallel projection algorithm for a simple model. Top images shows the mesh after sun shoot computed at the following conditions: 20th June, latitude 41, longitude 2, time 8:00, 9:00 and 19:00 GMT respectively. Bottom images correspond to the mesh after the sky shoot for the same conditions.

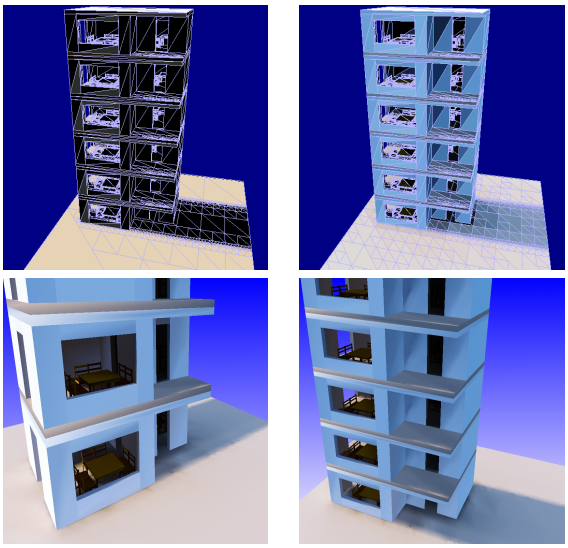

Figure 4: Daylighting simulation on a building at 16:00h. First row shows the resulted mesh after sun (left) and sky (right) projection. Bottom images show snaps for different views for the converged solution.

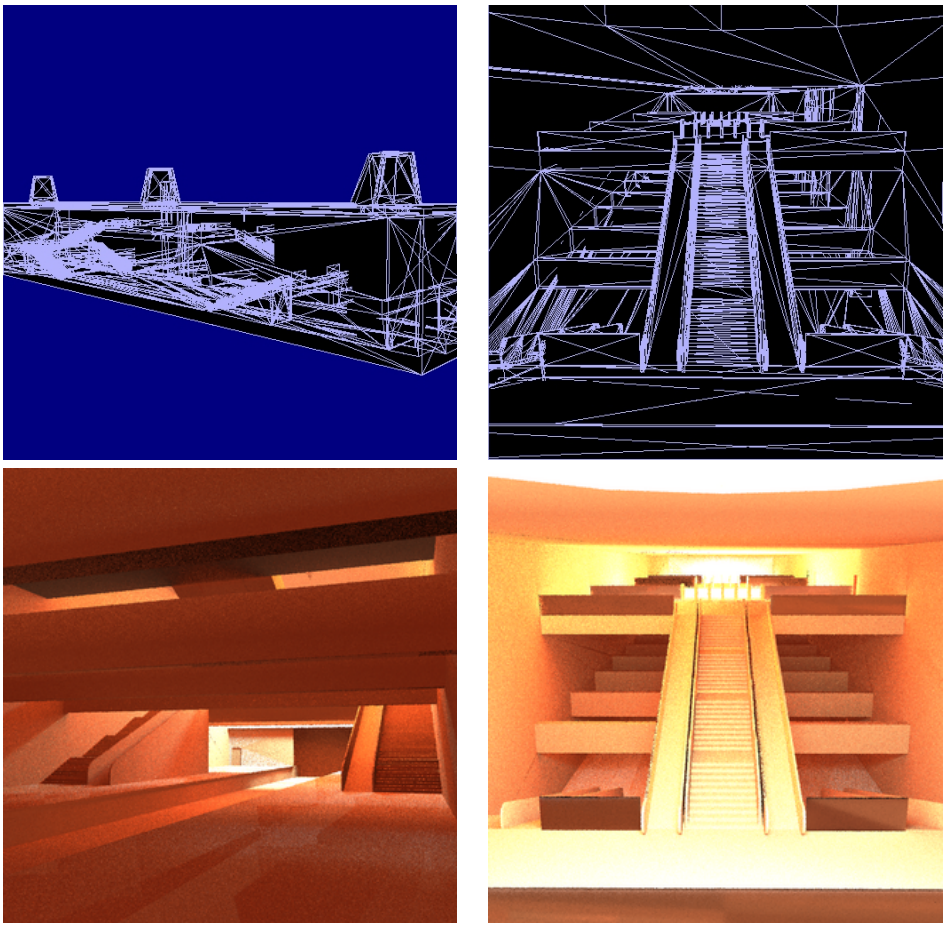

Figure 5: Results for an underground metro station. Top images show the scene mesh, the model it is designed to improve natural lighting in the building that enters by the openings of the roof. Bottom images show the final gather after daylight simulation of the interior of the station.

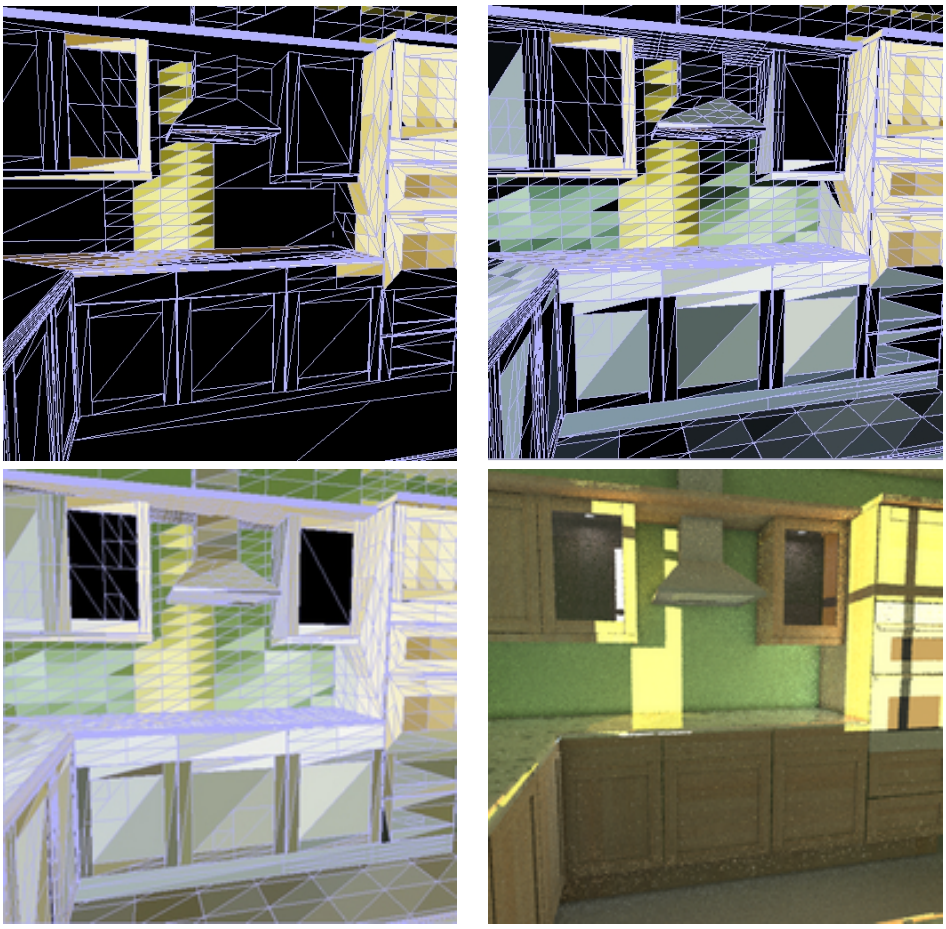

Figure 6: Results for the kitchen scene. From left to right show the mesh after the sun shoot, the mesh after the sky projection, the mesh after the Hierarchical Monte Carlo Radiosity convergence and the final gather result.