On Scene Complexity Definition for Rendering

Ludovit Niepel, Jozef Martinka, Andrej Ferko, Pavol Eliass
Department of Computer Graphics and Image Processing,
Comenius University, 842 15 Bratislava, Slovakia
e-mail: niepel, ferko, elias@fmph.uniba.sk,
Jozef.Martinka@st.fmph.uniba.sk

Abstract

This paper introduces one way how to define in a mathematically correct way the complexity of the polygonal scene for rendering. This complexity allows to compare previously uncomparable rendering algorithms via computing the complexity of various scenes, and give the more rigid solution of other related problems. The advantages of this unified approach are discussed.

1 Introduction

Let S be a given polygonal scene with N patches. The inventor of the notion "scene" is not known, for some ancient references, which are suspect to introduce this notion, see e. g. [Fole90], [Watt92]. The rendering R(S) of S depends on the complexity of S. In many papers the complexity of the scene is simply identified with N. In contradiction to this "dealing with environmental complexity and with human perceptual issues" are the 1994 answers to the question of Peter Shirley: "What are the key issues that are still unanswered or not answered well in realistic image synthesis?", [Shir94]. It is suitable to distinguish between geometric and perceptual or visual complexity of the scene, as discussed in [ChSh94].

Our research focused to the geometric complexity of the scene. Intuitively, it is possible to formulate following observations. (The last pair of them remain questionable.)

1. Empty room is not complex.
2. Adding objects increases the complexity of the scene.
3. Refining mesh doesn't change the complexity.
4. The meshed sphere has only a little bit higher complexity than meshed cube. (Simple topology means low complexity: e.g. thorus has higher complexity than the sphere.)
5. The complexity depends on the size of objects.
6. The complexity depends on the number of objects.

A number of patches (N) became the most heavily used complexity measure for rendering algorithms based on polygonal description of a scene. This is due to the fact, that most algorithms require the computation of entities, which number depends directly on the number of patches. As an example one can look at hemicube radiosity algorithm [CoGr85], where the evaluation of O(N^2) formfactors is necessary in order to solve an equation system for energy equilibrum for the given scene. A similar situation concerns the progressive refinement algorithm [Cohe88]. One iteration step requires the computation of N formfactors. On the other hand, the total number of the iteration steps depends more on the material properties of the scene (self emmitance of light sources, reflective properties of objects) and therefore the visual complexity of the scene is involved instead of geometric complexity.

A problem of the visual and mathematical quality of a solution to global illumination problem depends mostly on the meshing of the given scene. This is mainly due to the latest advances in evaluation of energy transfers between two polygonal patches (see closed form expression of polygon-to-polygon formfactor in [ScHa93]). If meshing is used only as a preprocessing method, a lot of information gained by a rendering method is wasted (e.g. information on shadow borders). Therefore, new methods have been developed, which take advantage of these additional information. To mention at least the most important ones: adaptive patch subdivision [Cohe86] (the patch is subdivided if a illumination gradient between vertices of the patch is too high), discontinuity meshing [Lisc92] (the patch is subdivided along the shadow boundaries caused by light sources), hierarchical radiosity [Hanr91] (the energy interaction between two patches occurs on those levels of subdivision hierarchy, where the error is acceptable).

As long as the number of energy transfers depends on the geometrical properties of the scene, the number of meshing patches is not enough to be able to compare the quality of different methods. Moreover, a good geometrical complexity measure is necessary framework for comparison of different meshing strategies. This motivates the following formulation of research problem:

Let S1, S2 are the scenes both with N patches. Let S1 seems inuitively more complex than S2. Give the function f, which formalizes the intuition (as approximately summarized in 6 numbered observations above) so that

$$\mathbf{f(S1) < f(S2).}$$

The paper is organized as follows. Section 2 gives some trivial complexity measures, which can be used for very rough orientation in the topic and gives some first observations. Section 3 gives the overview of necessary graphtheoretic notions, and definition of scene complexity. Section 4 explains the algorithm for complexity measurement and results obtained. Section 5 contains the discussion of the results, and conclusion.

2 Trivial Complexity Measures

2.1 Area Approach

The number of patches is still the most popular complexity measure, but we will consider some other approaches.

If revisited the patch data, there are available patch corners 3D coordinates, normal vector to the patch, light energy of the patch, and reflectivity properties. Let the scene S has extents in all dimensions denoted by intervals EX, EY, EZ, where EX = [xmin, xmax], EY = [ymin, ymax], EZ = [zmin, zmax]. If the scene envelope (bounding box) is not a paralelogram, we can bound it by the bounding box of volume EX * EY * EZ. Then the empty scene, having no interior objects, has the total area equal to the size of surface of volume EX * EY * EZ

$$\mathbf{A(S) = 2\,(EX{*}EY + EY{*}EZ + EZ{*}EX)}$$

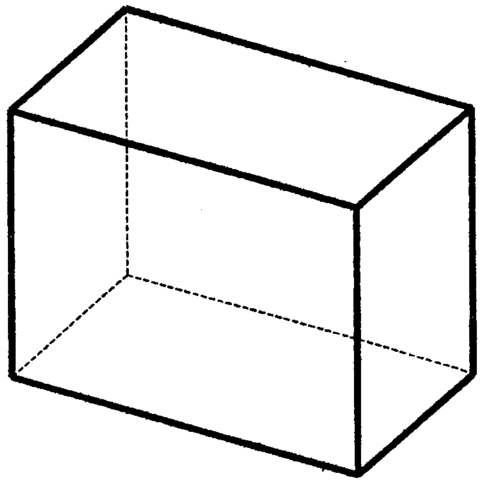


Fig. 1. The scene with trivial complexity.

If the scene is nonempty, what represents the practically used cases, then there is simply computable the total sum of areas of all patches

P(S) = sum of areas of all patches.

Observation of fraction P(S)/A(S) - 1 behaviour gives the simplest complexity measure. It is zero for empty scene. It grows with the total area of the scene (together with the area of boundary). However, this cannot distinguish rather differently complex scenes. This is illustrated in Fig. 2.

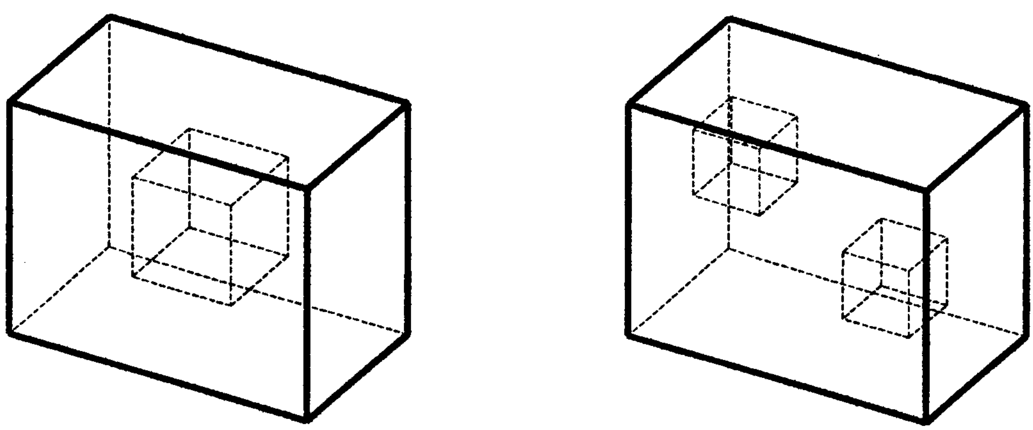


Fig. 2. Two scenes with equal area complexity.

The area approach gives for scene complexity measure the function

$$f1(S) = P(S)/A(S) - 1.$$

2.2 Normals of Pairs of Patches

The set of patch normals gives another characterization of the scene. The distribution of normals doesn't seem to be a measure of scene complexity, but we can observe the pairs of patches, too. There are 3 possible cases for their normals:

1. Patches are coplanar, the normals are equal, with some tolerance.
2. Patches are visible to each other, the normal halflines can intersect.
3. Patches are invisible.

This gives N(N-1)/2 pairs, and their qualification into the three groups characterizes the scene from the aspect of normals statistics. Denote NC number of pairwise coplanar patches, NV number of visible pairs, NI number of invisible pairs of patches. The numbers NC, NV, NI must be normalized by dividing by N(N-1) to give three numbers independent from N, their .sum equals to 2. The function

$$\mathbf{f2(S) = (NC/N(N\text{-}1), NV/N(N\text{-}1), NI/N(N\text{-}1))}$$

gives a simple statistical measure, which can distinguish scenes according to distribution of normals of pairs of patches.

2.3 Three or more patches

The previous approach can be generalized to triplets or quadruplets of normals of patches. The combinatorial number of possible cases increases - and the redundancy, too. If we take into account only the adjacent patches, we can decrease the volume of results. In particular, for triangulated scene there is useful to consider quadruplets, because each triangle has 3 neigbours in "well done" scenes. Another problem arises if two objects touch themselves. There occurs "new adjacency" not recorded in data structure.

To conclude this kind of observations - it seems that the normals statistics gives too much unnecesary information and it needs to be improved (or restricted) by adjacency considerations. The crucial disadvantage of this approach is that it gives only very rough estimation of light reflections.

2.4 Visibility Approach

Another straightforward way how to obtain scene complexity definition is the patch visibility precomputation. In fact, it seems that the scene complexity cannot be expressed adequately only with patch area measure or normals statistics information. The appropriate definition of scene complexity cannot be reduced from global topology description aspect and light reflection model. One possible answer to this question is outlined in next two sections.

3 Graphtheoretical Preliminaries

The fundamental notion used here to evaluate the scene complexity is the reachability graph G: the vertices of G mean single patches of polygonal scene and the edges represent their bilateral visibility. For the sake of uniqueness it is suitable to assume that any pair of patches is either totally visible or invisible. Thus we consider the scene with visibility obstacles blocking the two given patches completely.

For any scene it is possible to satisfy the assumption by appropriate subdivision of patches, but we can't do this in practice.

For given patch P the illumination contribution from other patch depends on the number of refections necessary for the ray to reach one patch from the other. In graphtheoretic language this means the distance d(u,v) of two representing vertices u, v of reachability graph G. The distance is the minimum number of edges of a path connecting the pair of vertices. (The path may be not unique!) If such a path does not exist, the distance is infinite. The reachability graph G is a connected one and therefore all distances are finite. (For two separate rooms the infinite distances are not taken into account.)

The graphtheoretic distance induces some other notions, which we will need for further considerations. The excentricity of a vertex v is a maximum distance d(v,u) for all u in G. Maximal excentricity is diameter of G, diam (G), minimal excentricity is radius of G, r(G). An important parameter, describing the metric structure of G, is the average distance of vertices, it is the average of N(N-1) possible distances in G.

When applying to the scene complexity, from the graph parameters described above, the average distance of reachability graph G could give the suitable scene complexity measure, because it is independent from the number of patches, and from possible singular cases (unlike diameter), and from the localization of light sources and observer (unlike radius).

4 Algorithm and Results

If the reachability graph G(S) of given scene S is not very large, and it is possible to store the graph in given memory, there is no problem to compute the distance of a fixed pair of vertices, e.g. by Dijkstra's algorithm, respectively to compute all distances, e.g. by Floyd's algorithm. Both classical algorithms are described in many books, e.g. in [SwTh84].

The complications arise if G is too large to be stored in given memory. The scene with N patches may be storable in memory, while the O(N^2) edges of reachability graph could cause memory overflow. This problem can be solved using the following idea.

The distances of given vertex v in G is possible to compute without having all G in memory. Evidently, the computational cost increases, because we must test if two vertices are adjacent or not. The algorithm of computing the distances of vertices from given fixed vertex v we can roughly described as follows. Each vertex will be assigned distance value DIST, initially zero for starting vertex v and infinite for all others. The strategy is breadth first search with starting vertex v. We test each vertex u of G, if it is joined with v by an edge. For adjacent vertex u we set DIST(u) = 1.

In the next phase we test the vertices with DIST(u) = 1. If a vertex u with DIST(u) = 1 is connected with any vertex w with infinite distance value, we set the DIST(w) = 2. This phase finishes if we have investigated all vertices u with DIST(u) = 1, or if all vertices have finite distance value. The process is repeated. In the k-th phase we search the vertices with DIST(u) = k, searching for adjacent vertices w, connected with some other vertex with infinite distance value. In this case we set DIST(w) = k+1. This process finishes if all vertices have the finite evaluation. The average distance of the reachability graph G(S) of given scene S is evidently the average of DIST(u) for all u in G. Next figure illustrates the reachability graph.

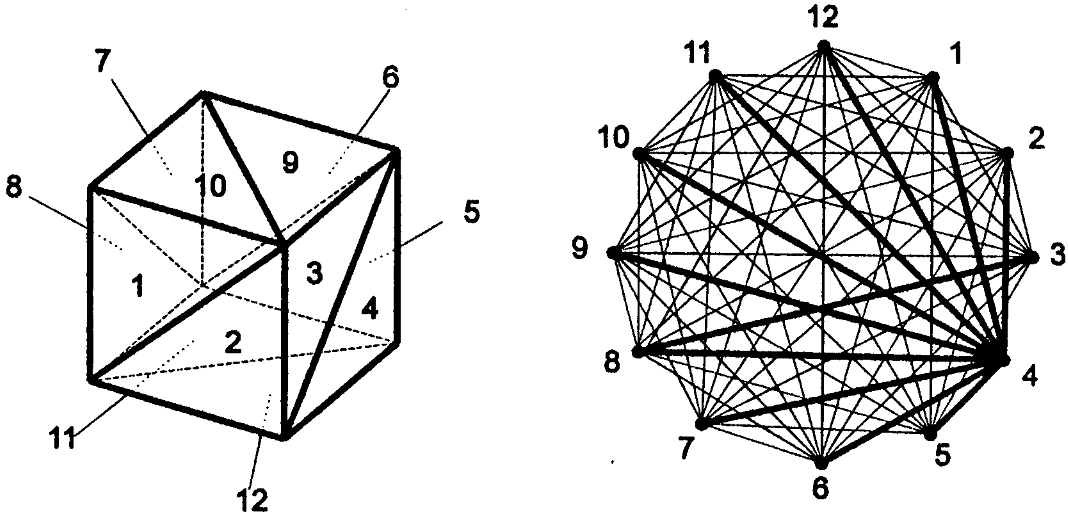
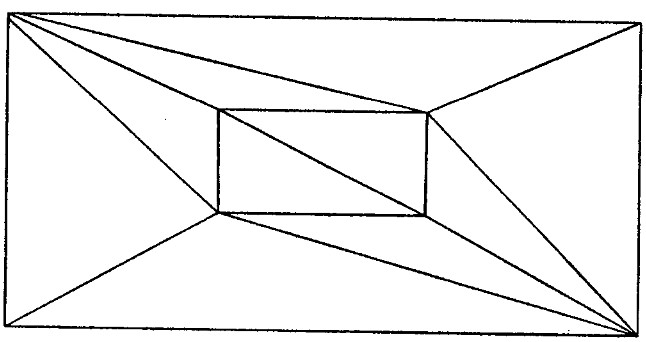


Fig. 3 Scene S1, and the reachability graph G(S1).

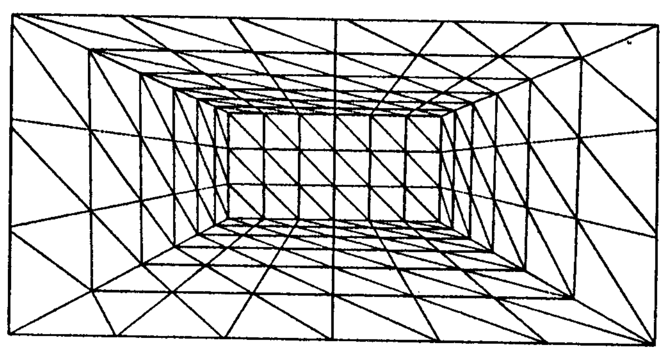
Using the approach as described above, we have obtained following results:

	1	2	3	4	5	6	7	8	9
average ditance	1.09090	1.18469	1.19679	1.27562	1.50609	1.41932	1.98710	1.90463	2.04483
patches	12	288	1 152	320	1 344	1 344	92	1 264	538

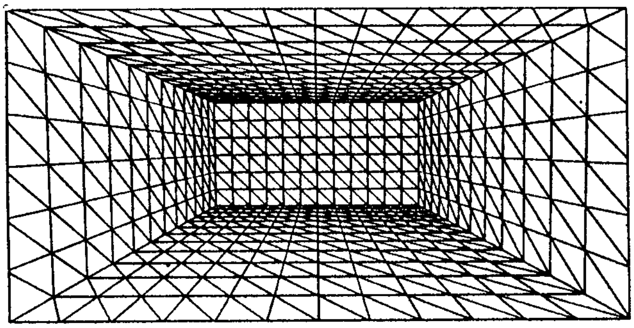
Tab. 1 The complexity of given scenes.



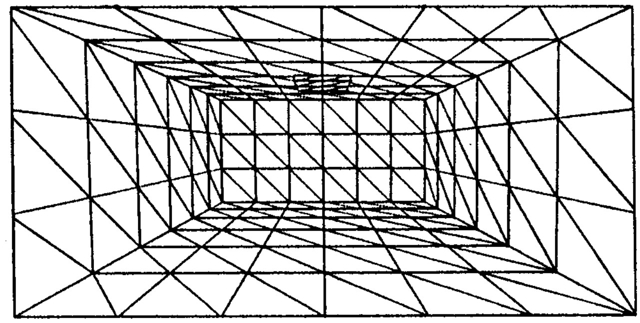
scene 1 *empty room.*



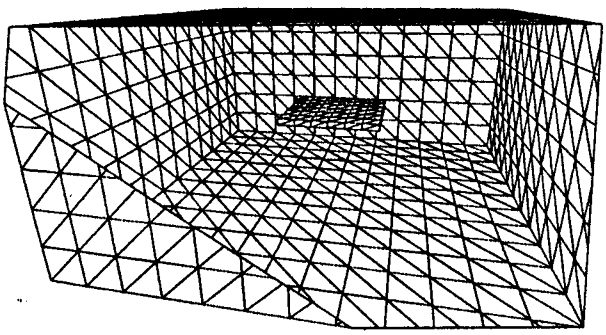
scene 2 *scene 1 after refining the mesh.*



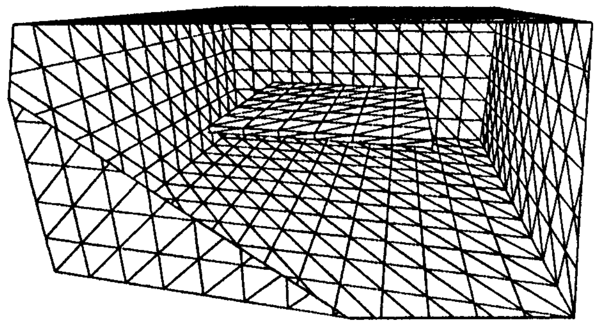
scene 3 *scene 1 after further refining the mesh.*



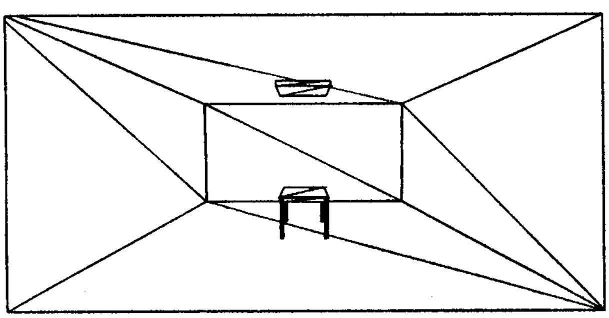
scene 4 *scene 1 with aded object for light source.*



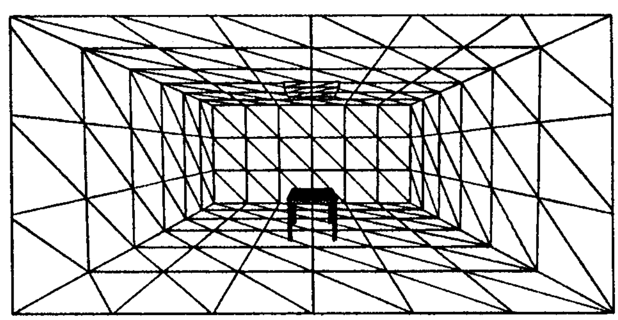
scene 5 *empty room with small board in center.*



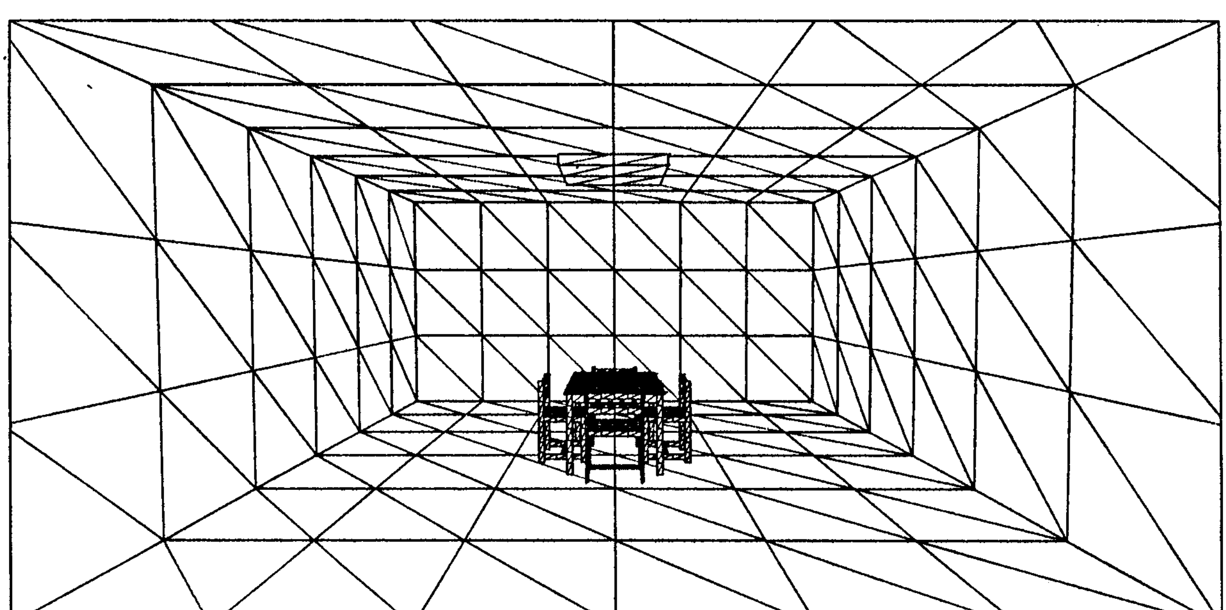
scene 6 *as scene 6 but with four times bigger board in center.*



scene 7 *scene 4 with aded a table.*



scene 8 *scene 7 after refining the mesh.*



scene 9 *as scene 7 but four chairs have been aded.*

5 Conclusion

We have calculated the complexity for a set of scenes, which are illustrated in appendix. The results show that the formalization of scene complexity using the average distance in reachability graph gives a good evaluation of intuitively estimated complexity. The computational price for this is very high for real scenes.

There are many further questions unsolved. Should we take instead of the shortest one all ways of light? How compute this for partially visible patches without subdivision? How to express various (non-triangle) patches? Is this complexity measure sufficient enough?

We have introduced a new approach to define the complexity of given polygonal scene. This complexity allows to compare previously uncomparable rendering algorithms via computing the various scenes complexity. It seems it will be possible to generate the scene with given complexity, and give the more rigid solution of other related problems. The main advantage of this way of scene complexity definition is the formalization of less or more intuitively used notion scene complexity. This research will continue with many well known scenes evaluation. We would like to ask the scene users community to send us their scenes, or any scene complexity results and ideas.

References

[SwTh84] Swamy, M. N. S. - Thulasiraman, K.:
"Graphs, Networks, and Algorithms"
Wiley 1984

[CoGr85] Cohen M., Greenberg D.:
"The Hemi-Cube: A Radiosity Solution for Complex Environments"
Computer Graphics vol. 19, num. 3 (Siggraph'85 Proc.)

[Cohe88] Cohen M., Chen S., Wallace J., Greenberg D.:
"A Progressive Refinement Approach to Fast Radiosity Image Generation"
Computer Graphics vol. 22, num 4, (Siggraph'88 Proc.)

[Cohe93] Cohen M., Wallace J.:
"Radiosity and Realistic Image Synthesis"
Cambridge, MA: Academic Press Professional, 1993

[Lisc92] Lischinski D., Tampieri F., Greenberg D.:
"Discontinuity Meshing for Accurate Radiosity"
IEEE CG & A , 1992, Vol 12, Num 6

[ScHa93] Schroeder P., Hanrahan P.:
"On the Form Factor between Two Polygons"
Computer Graphics, Siggraph '93 Proc.

[Cohe86] Cohen M., Greenberg D., Immel D., Brock P.:
"An Efficient Radiosity Approach for Realistic Image Synthesis"
IEEE CG&A, 1986, Vol. 6, Num 3.

Shirley P.:
"Rendering database"
available by anonymous ftp from "moose.cs.indiana.edu"

[Shir94] Shirley P.:
"Results of the 1994 Survey on Image Synthesis"
Fifth EUROGRAPHICS Workshop on Rendering, pp. 3 - 6,
Darmstadt, Germany, June 13-15, 1994

[Elia94] Elias, P.:
"Proposal for Perceptually Based Error Metrics for Global Illumination Algorithms"
Spring School on Computer Graphics, pp. 68 - 74
Bratislava, Slovakia, June 6 - 9, 1994

[ChSh94] Chiu, K. - Shirley P.:
"Rendering, Complexity, and Perception"
Fifth EUROGRAPHICS Workshop on Rendering, pp. 19 - 33,
Darmstadt, Germany, June 13-15, 1994

[Fole90] Foley, J. D. et al.:
"Computer Graphics, Principles and Practice"
Addison Wesley 1990

[Watt92] Watt, A. - Watt, M.:
"Advanced Animation and Rendering Techniques"
ACM Press, Addison Wesley, New York 1992

[Hanr91] Hanrahan, P. - Salzman, D. - Aupperle, L.:
"A Rapid Hierarchical Radiosity Algorithm"
Computer Graphics, Vol. 25, No. 4, Siggraph'91 Proc.