Feature-based Control of Visibility Error: A Multi-Resolution Clustering Algorithm for Global Illumination
François X. Sillion, George Drettakis

To cite this version:

HAL Id: inria-00510125
https://inria.hal.science/inria-00510125
Submitted on 17 Aug 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers. L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Feature-based Control of Visibility Error:  
A Multi-resolution Clustering Algorithm for Global Illumination

François Sillion\textsuperscript{1}  George Drettakis\textsuperscript{2}

\textsuperscript{1}CNRS  \textsuperscript{2}ERCIM-INRIA  
iMAGIS  
B.P. 53, 38041 Grenoble Cedex 9, France.

Abstract

In this paper we introduce a new approach to controlling error in hierarchical clustering algorithms for radiosity. The new method ensures that just enough work is done to meet the user's quality criteria. To this end the importance of traditionally ignored visibility error is identified, and the concept of \textit{features} is introduced as a way to evaluate the quality of an image. A methodology to evaluate error based on features is presented, which leads to the development of a \textit{multi-resolution visibility} algorithm. An algorithm to construct a suitable hierarchy for clustering and multi-resolution visibility is also proposed. Results of the implementation show that the multi-resolution approach has the potential of providing significant computational savings depending on the choice of feature size the user is interested in. They also illustrate the relevance of the feature-based error analysis. The proposed algorithms are well suited to the development of interactive lighting simulation systems since they allow more user control. Two additional mechanisms to control the quality of a simulation are presented: The evaluation of internal visibility in a cluster produces more accurate solutions for a given error bound; a progressive multi-gridding approach is introduced for hierarchical radiosity, allowing continuous refinement of a solution in an interactive session.

Keywords: Visibility error, Clustering, Feature-based error metric, Multi-resolution visibility, Hierarchical radiosity, Progressive multi-gridding, Global Illumination.

1 Introduction

Modern global illumination algorithms allow the precise simulation of interreflection effects, penumbras caused by extended light sources, and subtle shading variations caused by complex reflectance properties [2, 15]. Lighting simulation systems operate under very tight and often contradictory constraints: users typically require guaranteed and easily controllable precision levels, with maximum speed for interactive design. An important goal of rendering research is thus to enable the user to reduce the solution error where such reduction is deemed desirable, while at the same time limiting the time spent to achieve this reduction.

Unfortunately, the algorithmic complexity of radiosity methods (quadratic in the number of objects) in effect impairs their use for scenes containing more than a few thousands objects, while Monte-Carlo methods are unable to provide low and medium-quality solutions without too much noise. Therefore means must be found to focus the effort on the most important parts of the calculation.

This paper presents new algorithms and criteria that together allow very fine and efficient user control of the perceived quality of a solution. This is accomplished by first acknowledging the importance of \textit{visibility error}, and using the concept of \textit{features} to evaluate the quality of a solution. This leads to the introduction of \textit{multi-resolution visibility}, which allows precise control of the quality vs. time tradeoff. Additional mechanisms are then discussed to control the quality of a simulation in a working system.

Previous work: error-driven computation and visibility

The introduction of the hierarchical radiosity algorithm [5] was a major step towards the design of practical lighting simulation systems. First, it reduces the overall resource requirements for a given solution. Second, it uses a surface subdivision criterion as an explicit control mechanism. This criterion embodies the priorities used to guide the simulation, as it directs the computational effort to “areas of interest”, introducing a natural tool for error estimation.

Hierarchical radiosity (HR) remains quadratic in the number of input objects (since each pair of objects must be linked before hierarchical subdivision begins), and therefore is not suited to large collections of small objects. Clustering, the operation of grouping objects together into composite objects that can interact, provides a means to eliminate the quadratic complexity term. Such clustering can be performed manually [11, 7] or automatically [16, 13].

Historically, subdivision criteria for HR first consisted of simple bounds on either the form factor or the exchange of radiosity between two surface patches [5], under the assumption that the error incurred is proportional to the magnitude of the transfer. Using the concept of importance these bounds can be made dependent on the user’s interest for each region [17]. However such bounds tend to be quite conservative and thus produce unnecessary subdivision [6].

Recent work has attempted to characterize possible sources of error in global illumination [1], and establish error bounds on radiosity solutions [8]. These error bounds can then be used in the subdivision criterion of a hierarchical algorithm. Since the estimation of the error is decoupled from that of the actual transfer, subdivision can be avoided in regions where significant transfers take place without much error, resulting in better focus of the computational expense.

Existing error controls however typically ignore visibility as a possible source of error, or simply increase the error estimate by a constant factor in situations of partial visibility. Trivial bounds of 0 (total occlusion) and 1 (total visibility) are often used. While these
bounds are always valid, their use results in unnecessary work being
done to narrow down other error bounds by increasing the subdivi-
sion. Global visibility algorithms can be used to exploit the struc-
ture of architectural scenes and produce guaranteed visibility in-
formation [18], but they are not suited to large collections of inde-
pendent objects. For exchanges between surfaces, discontinuity mesh-
ing also provides explicit visibility information, and indeed consid-
erably improves the efficiency of HR [10]. However for Monte-
Carlo or clustering approaches it is either impossible or impractical
to calculate analytic visibility and error bounds must be used. For
exchanges between clusters, an approximate visibility estimate can
be derived using equivalent volume extinction properties [13], but
the error introduced in the process has not yet been analyzed.

Visibility error is admittedly difficult to evaluate, since the com-
putation of visibility itself is a costly process. Still, controlling this
source of error is imperative since the quality of shadows plays a sig-
nificant role in determining the user’s perception of image quality.
In complex environments where clustering is most useful, a domi-
nant part of computation time is spent in visibility calculations in-
volving small, geometrically complex objects. Resulting visibility
variations produce fine detail shadows, which may be of little inter-
est to the user, or may be lost in the implicit averaging over a surface
patch.

Paper overview

The preceding discussion has shown that a key issue in designing
efficient lighting simulation systems is to provide adequate con-
trol mechanisms to ensure that just enough work is done to meet
the user’s quality criteria. It appears that control of visibility error
has not yet been attempted, despite its great potential for tightening
global bounds and reducing computation costs. The goal of this pa-
ter is twofold: first, a new approach to visibility error estimation
is proposed, based on features, that legitimates the use of a multi-
resolution visibility algorithm. Second, quality control mechanisms
are discussed for interactive simulation systems development.

We begin in Section 2 with the introduction of features to evalu-
ate image quality, and show why existing error metrics are incapable
of determining when a given level of detail is satisfactorily repre-
sented. A simple metric is then proposed to illustrate how to take
into account the user’s interest in a minimal feature size. This leads
to Section 3 where we explain how to compute multi-resolution vis-
ibility information using a spatial hierarchy augmented with equiva-
 lent extinction properties. Selection of a hierarchical level for vis-
ibility computation can then be based on the resulting feature size
on the receiver. In this paper an application to clustering algorithms
is discussed, but multi-resolution visibility is equally promising for
Monte Carlo techniques. The construction of a suitable hierarchy
is discussed in Section 4. In Section 5 we show that the multi-
resolution visibility algorithm successfully generates images (cur-
cently for isotropic clusters) in which only selected features sizes
are accurately represented, resulting in computational savings.
Section 6 presents more quality controls for clustering algorithms,
specifically intra-cluster visibility determination in linear time and
progressive multi-gridding. We conclude in Section 7.

2 Feature-Based Error Analysis

To a large extent the quality of an image is judged based on how well
features of different sizes are represented. It is not easy to charac-
terize what constitutes an illumination feature. For the purposes of
this paper, we will consider image features to be the connected re-
gions of varying illumination related to shadows (regions in umbra
or penumbra).

2.1 \(L^p\) metrics are inadequate for “feature detection”

A major difficulty for accurate lighting simulation is that in general
the exact solution is not known at the time of computation. Thus the
estimation of the error in a proposed approximation is particularly
difficult, and must rely on the computation of error bounds for all
algorithmic operations. Even in the case where an exact solution is
available, it is not a simple task to define the quality of a given
approximation. This is done by choosing a particular error metric
to quantify the distance between an approximate solution and the
true solution. A “good” metric should therefore convey a sense of
the user’s requirements. A central observation in this paper is that
when simulating a complex scene, the user is typically interested in
capturing illumination variations down to a certain scale. Very small
details are not as important, or at least not in all areas of the scene.
We strive to define a control mechanism that will avoid any work
that would only generate such small details.

![In each column below a cluster of cubes similar to this one is placed between a light source and a wall. The size of the cubes increases from left to right.](image)

Figure 1: Comparison of approximate illumination solutions using
different clusters. Top: reference images (illumination of the wall).
Middle: approximate images using a coarse mesh. Bottom: \(L^2\) er-
ror norms. Note that the four images have similar \(L^2\) error values,
and all hide some illumination information. However the varying
size of the missing features cannot be discovered.

Figure 1 illustrates the issue by showing shadows cast on a wall
by four different groups of objects. Four approximate images, all
computed using the same mesh size, are shown below the “exact”
images. Consider a user who is interested in shadows of a specific
size, e.g. those of the image on the extreme right, but is satisfied
by the averaging of the smaller, detailed shadows on the left\(^1\). The
user thus does not wish more work to be done for the detail shad-
ows, but wishes to have a more accurate representation at the larger
scale. The subdivision criterion used in a HR algorithm for instance
should be capable of halting the subdivision for the left-hand group,
while ordering further computation for the group on the right. Thus
an error measure should distinguish between the four cases.

Traditional error metrics are incapable of making such a distinc-
tion. As an example consider the commonly used family of error
metrics expressing the distance between a reference function \(f\) and
an approximate function \(\hat{f}\) as the \(L^p\) norm

\[
||\hat{f} - f||_p = \left( \int |\hat{f}(x) - f(x)|^p dx \right)^{\frac{1}{p}}
\]

\(^1\)Perhaps a more realistic example would be a situation where a user is
viewing an office scene from the doorway, and in which accurate shadows
for chairs and desks are important, but averaged, low quality shadows from
details such as pens on a desk are satisfactory.


$L^p$ norms simply add error contributions from all points on a surface (or in an image), and do not take into account higher-level properties of the radiance distributions, such as the size and shape of illumination features. This is illustrated by the similar values obtained for the four groups in Figure 1. Appendix A shows that in fact for a point light source the $L^1$ or $L^2$ error introduced by averaging all visibility variations depends only on the average visibility, and not on the size or shape of the shadows.

2.2 A proposal for an error metric based on feature size

Our hypothesis is that illumination features (shadows or bright areas) are important only as far as they have a significant visual impact. Therefore it is possible to define a feature size on a receiving surface, and decide that features smaller than that size are “unimportant”: their absence should not contribute to the error.

The remainder of the paper refers to the radiosity function over a surface as an “image”. This terminology should not mask the important fact that the entire discussion takes place in three-dimensional object space. In order to demonstrate the relevance of this hypothesis, we assume for now that we have access to all the information in a reference solution. The multi-resolution visibility technique of Section 3 will show how the ideas developed here can still be used in the absence of such a reference.

A simple error metric based on features is defined by segmenting the image $f$ into two components by means of a feature mask $\mathcal{F}^s(f, x)$: a binary function that equals one at points $x$ that belong to a “feature” of size greater than $s$ of function $f$. Computation of feature masks from the reference solution is described in the next section. For points in the mask region we compute an $L^p$ norm of the difference between the approximate function and the reference function. For points outside the feature mask, we are content with an average value (since features present there are smaller than $s$). Thus in our current implementation we compute average values at each point, for both the approximate and reference functions, using a box filter of size $s$ around the point of interest, and compute an $L^p$ norm of the difference between the averages.

The feature-based error metric (FBEM) is summarized by the following formula, where $\hat{f}$ represents the filtered version of $f$:

$$
\|\hat{f} - f\|_p = \left( \int |\hat{f}^s(x) - \mathcal{F}^s(f, x)|^p \right)^{\frac{1}{p}}
$$

2.3 Examples

Table 1 shows the FBEM values computed for the four groups of Figure 1 and different values of the minimum feature size $s$. The object-space size of typical shadows in these images is respectively 11, 16.5, 22 and 31. For small $s$ values, all FBEM values are high since the metric is equivalent to an $L^2$ metric in the limit of $s = 0$. As $s$ increases, FBEM values decrease more rapidly for the groups containing smaller objects, as expected. There appears to be a residual error of about 3 due to the mesh size used for the approximate solutions.

Assume the user is interested in clearly seeing features of size 30 or greater, while being content with an average for all features smaller that this size. The extreme right-hand image of Figure 1 requires more work since the FBEM value for $s = 30$ is high. The approximation for the other three images is deemed satisfactory since the error is low.

Thus, using the FBEM presented above, it is possible to reveal the presence of features greater than a given threshold in the approximated images, opening the way for selective subdivision based on the user’s minimum feature size of interest. Of course this could not be used as is in a subdivision criterion for HR, since it uses a reference solution, but it is useful for a posteriori validation of control mechanisms.

2.4 Computation of feature masks

According to the definition of features given above, computing a feature mask amounts to identifying connected regions of “significant” size. Mathematical morphology provides tools to isolate features based on their size [12]. Consider a binary image, representing for example the characteristic function of an object. We define the action of an Erosion operator as follows: all points outside the object (white) are untouched. All points inside the object that have a neighbor outside become white. All other points remain black. An Expansion operator is defined similarly by including in the objects all outside points that have a neighbor in the object. Figure 2 shows a reference image and images obtained after a number of erosions (top) or expansions (middle).

Clearly an object of diameter $2d$ will disappear after $d$ erosions are applied in sequence. Thus applying a sequence of $n$ erosions followed by $n$ expansions will successfully eliminate all small regions, but keep larger regions (slightly modifying their shape in the process). This process is illustrated in the bottom row of Figure 2.

Computing the effect of the erosion operator on a binary image is straightforward using bitwise operations: the result is the logical OR of the image and the four translated copies of itself (by one pixel) in the $+x$, $-x$, $+y$ and $-y$ directions. For the expansion operator the logical operator AND is used.

In our examples, the original binary image is computed by recording all areas of the receiver that have a partial or occluded

<table>
<thead>
<tr>
<th>Feature size</th>
<th>5</th>
<th>14.76</th>
<th>16.34</th>
<th>17.25</th>
<th>17.31</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>9.37</td>
<td>12.24</td>
<td>13.76</td>
<td>15.80</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>4.78</td>
<td>6.50</td>
<td>9.06</td>
<td>14.74</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>4.23</td>
<td>3.16</td>
<td>6.90</td>
<td>13.37</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>3.65</td>
<td>2.33</td>
<td>3.35</td>
<td>6.94</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Feature-based error metric (FBEM) for the four approximate images of Figure 1 and five different feature sizes. The four measures are equivalent for small feature sizes, and decrease at different rates as a function of $s$. Images are shown again for clarity.
view of the light source. This expensive operation was performed only once during the creation of the reference image. Feature masks are computed by applying the proper number (p/2 for a feature size of p) of successive erosions and expansions to eliminate unwanted features. Figure 3 shows some feature masks for the four groups used above.

<table>
<thead>
<tr>
<th>Original mask</th>
<th>F Size 12</th>
<th>F Size 18</th>
<th>F Size 24</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Image](65x463 to 111x508)</td>
<td><img src="65x509" alt="Image" /></td>
<td>![Image](65x512 to 111x556)</td>
<td><img src="65x558" alt="Image" /></td>
</tr>
<tr>
<td>![Image](65x560 to 111x604)</td>
<td><img src="65x606" alt="Image" /></td>
<td>![Image](65x608 to 111x653)</td>
<td><img src="65x654" alt="Image" /></td>
</tr>
</tbody>
</table>

Figure 3: Some feature masks for the images in Figure 1.

3 A Multi-resolution Visibility Algorithm

In the previous section we presented the concept of a feature size and introduced an error metric which permits the evaluation of image quality determined by how well illumination features are represented. We now use these fundamental concepts to develop a multi-resolution (MR) visibility algorithm. With this algorithm, expensive high quality visibility calculations are only performed when they are expected to help in the accurate representation of features deemed “interesting” by the user.

Hierarchical spatial subdivision structures are often used in the calculation of global illumination algorithms, in particular when form-factor estimation is performed with ray-tracing [19, 4, 5, etc.]. In radiosity clustering algorithms the hierarchy of clusters is also used for radiometric calculations, by letting clusters represent their contents for some energy transfers [13, 16]. The following multi-resolution visibility algorithm naturally extends previous clustering approaches by allowing clusters to also represent their contents in some visibility calculations. If a specific feature size s has been chosen, it is unnecessary to consider the contents of a cluster for visibility if these contents will produce features smaller than s.

3.1 Approximate visibility computation between clusters using an extinction model

Let us assume that we have grouped all objects in the scene into a hierarchy of clusters. Approximate visibility calculations can be performed using an analogy between clusters and absorbing volumes [13]. The approximation (asymptotically exact for homogeneous isotropic clusters when the size of the objects goes to zero) consists of associating an extinction coefficient \( \kappa \) with each cluster. The transmittance function between two points \( P \) and \( Q \) in the scene is then given by

\[
T(P, Q) = e^{-\int_{PQ} \kappa(u) du} - \sum_{i \in C(PQ)} \kappa_i l_i
\]

where \( C(PQ) \) is the set of clusters traversed by the ray joining \( P \) and \( Q \), \( \kappa_i \) is the extinction coefficient of cluster \( i \), and \( l_i \) is the length traveled inside cluster \( i \) by the ray.

Extinction coefficients express the probability that a random ray is intercepted in the cluster, and are computed as

\[
\kappa_i = \frac{\sum_j A_j}{4V_i}
\]

where the area of all surface patches contained in cluster \( i \) is summed and divided by the cluster’s volume. Since a surface contributes to the extinction of only one cluster, the attenuation due to overlapping clusters is correctly obtained by adding their extinction contributions.

3.2 Multi-Resolution Visibility

In the rest of this section we consider the emitter-blocker-receiver configuration shown in Figure 4, which consists of two surfaces, the emitter \( E \) and the receiver \( R \), in two-dimensions. This restriction is for presentation purposes only and is removed later.

![Figure 4: Definition of shadow features created by a blocker. (a) The umbra region is unbounded since the blocker is larger than the emitter: there is always an umbra region on the receiver. (b) For some positions of the blocker the receiver has no umbra region.](Image)

If a blocker (which for now we also consider to be a surface) is placed between the emitter and the receiver, umbra and penumbra regions are created in space. Depending on the position of the blocker, there may or may not be an umbra region on the receiver. (Figure 4). Given the definition discussed above the size of the umbra zone on the receiver ±AB in Figure 4(a)−, if it exists, is the feature size.

The blocker may actually be a hierarchical representation of a collection of objects (a cluster) as pictured in Figure 5(a). In this case, at each level of the hierarchy an extinction coefficient is stored allowing the approximate calculation of the attenuation of a ray if it passes through the cluster, as described previously.

Multi-resolution visibility can be performed by avoiding the descent into the hierarchy after a certain level. When the required conditions are met the extinction coefficient is used instead, thus avoiding the intersection of the ray with all the descendants of this cluster. Evidently, the effect is that visibility is no longer exact, but an average estimation of transmittance. It is here that a large potential gain in computation time can be achieved. In scenes where the small detail objects (e.g., models of phones, keyboards, small objects on a desk etc.), comprise the largest part of the geometric complexity, the intersection with these objects can quickly become the overwhelming expense of visibility (and overall) computation. By considering
the higher level clusters for visibility computation instead of the numerous contents, when such a choice is dictated by the chosen feature size, this expense can be completely avoided.

![Figure 5: Visibility estimation through a cluster. (a) the blocker is a hierarchy of clusters. (b) an "equivalent blocker" is used to estimate the maximum feature size on the receiver.](image)

4 A Hierarchical Structure for Clustering and Multi-Resolution Visibility

Previous automatic clustering approaches have used spatial data structures developed for ray-tracing (hierarchical bounding boxes [3] were used in [16], while in [13] a K-D tree was used). In this section we show that given the calculation of average visibility based on extinction coefficients in the manner of [13], it is beneficial to develop a special-purpose hierarchical data structure, such that the resulting clusters have properties suitable for cluster-based hierarchical radiosity and multi-resolution visibility.

By definition, clusters are constructed to represent as accurately as possible the collection of objects they contain. By introducing computation of visibility using extinction coefficients and also multi-resolution visibility, apart from the representation of energy transfer of the contained objects as a whole, the clusters also need to correctly represent the transmission properties of the collection of contained objects.

These two modes of representation place different constraints on the cluster hierarchy. From the point of view of energy exchanges, good clusters allow tight bracketing of radiance or visibility functions (thus surfaces with similar orientation that do not shadow each other are preferred). From the point of view of visibility approximation, good clusters are ones for which the extinction property is plausible (thus homogeneous isotropic clusters are preferred). Given these constraints, we have identified two key properties for clusters: (a) proximity and (b) homogeneity of the contained objects. Maintaining proximity is a natural way to group objects when the cluster is used to represent radiative transfers. Also, for multi-resolution computation it is important that objects contained in a cluster are close so that the averaging performed does not introduce unacceptable artifacts. Homogeneity here means that we want a cluster to group objects of similar size, and is crucial for the resulting quality of the average visibility computation.

As a simple measure of proximity, we use the percentage of empty space resulting from a clustering operation (i.e., the addition of an object or a cluster to another cluster). Thus we prefer clusters in which the empty space is minimized.

To efficiently group objects of similar size we use a hierarchy of $n$ levels of uniform grids. We start with level 0, which is a single voxel the size of the bounding box of the scene and then at each level $i$ we create a grid which is subdivided into $2^i$ voxels along each axis. We then insert each object into the level for which its bounding box fits the voxel size.

Once these grids have been constructed, we start at the lowest level $n$, containing the smallest objects. We group the objects entirely contained in each voxel, by attempting to minimize the empty space, in accordance to the proximity criterion described above. In addition, objects which are very small compared to the grid size are grouped into an appropriate cluster, even if the resulting cluster is largely empty. Once all the voxels of a level have been treated, we attempt to add the objects not entirely contained in a single voxel at this level to the clusters already constructed, again using the same criteria. We then insert the clusters created to the grid of the level immediately above, and iterate.

Once the cluster hierarchy has been created, the data structure is augmented with average transmission behavior by propagating the average extinction values up the hierarchical structure as in [13].
When multi-resolution visibility is used, the $BSIZE$ estimation is also performed for each cluster in the hierarchy in the manner described in Section 3.

Figure 6 presents results obtained with the new hierarchy using first a surface visibility algorithm similar to that of [16], and then the average visibility proposed in [13]. The scene consists of 5380 polygons. It is interesting to observe the significant time gain achievable by the average visibility algorithm given a suitable hierarchy (we observe a factor of 4), while approximate shadows are preserved.

![3D view of test scene.](image)

![Reference sol. (2069 s).](image)

Figure 7: Reference image used in the error comparisons.

As we increase $s$ the $L^2$ norm for all clusters increases, as more and more averaging is being performed. Still the increase appears later for larger objects, as expected. The FBEM values are always of similar magnitude, despite the fact that very different levels of averaging are being used in different clusters in a given image. This shows that the multi-resolution visibility algorithm accomplishes its purpose: given a desired feature size, it ensures that the corresponding FBEM remains low while allowing time gains.

Figure 9 shows that even greater speedups can be achieved when a medium error threshold allows MR visibility to reduce the amount of subdivision. The explicit incorporation of MR visibility in refinement criteria is a promising path for further acceleration.

6 Control of Image Quality for Clustering

Recent algorithms separate the computation of high-quality images into two phases: a coarse quality global illumination calculation is first performed using elaborate algorithms such as discontinuity meshing or clustering in a global pass. A view-dependent, potentially very expensive local pass follows [9, 16]. This local pass is typically a ray-casting operation: at each pixel the energy from all the links is collected, allowing the calculation of high-quality shadows. The cost of this local pass is often many times larger than that of the light-transfer calculation using clusters. In essence this pass may eradicate all computation time benefit achieved by using the clusters in the first place, and exclude any possibility for interactivity with quality and error control.

In contrast, we maintain a “progressive refinement” philosophy, by providing explicit quality controls, allowing computational cost to be focused on desired characteristics of the resulting image. The first component of this approach is the multi-resolution visibility presented above. This technique, coupled with the use of importance [17] to assign appropriate feature sizes to different objects could plausibly replace the global/local pass approach while affording more interactivity. We present next two supplementary quality controls: first, the correct treatment of intra-cluster visibility and second, progressive multi-gridding permitting rapid interactive response for hierarchical radiosity.

6.1 Intra-cluster visibility

Previous clustering algorithms compute a bound on energy transfer that ignores visibility (bound of $1$ on the visibility error), both between the two clusters but also in the distribution of light on each side [16, 13]. This potentially results in light leaks at the scale of the cluster. This behavior is not only visually displeasing but also flawed: since bounds are computed on irradiance values, that irradiance is distributed to many surfaces which should be shadowed,
thereby creating energy.

If visibility information inside the cluster with respect to a source cluster can be computed (with some approximation) in time linear in the number of contained objects, the overall time and space complexity of \( O(s \log s) \) for clustering is not modified [16].

We propose the use of an item buffer to quickly evaluate this visibility. The cluster’s contents are projected in the direction of the light source using a z-buffer to determine visible surfaces from that direction. By counting instances of an item number in the buffer we obtain an estimate of the projected area of each patch visible from the direction of the source. This is used as the projected area in kernel calculations when computing energy bounds. Note that the resolution of the item-buffer can be adapted to the contents of each cluster, provided we know the size of the smallest object in each cluster. Thus the aliasing problems inherent to the item-buffer approach can be reduced. The same technique is also used at the other end of a link, to evaluate the energy leaving a cluster.

In the images of Figure 10 we present an example where a link (shown in purple) has been created from the light source to a cluster of books. Ignoring intra-cluster visibility (left) results in the creation of energy since all books are fully illuminated. Using the visibility buffer to modulate the energy distribution (right), energy is preserved while improving the appearance of the image.

6.2 Progressive multi-gridding

In hierarchical radiosity algorithms subdivision is controlled by an error threshold on individual interactions. A global bound on error is difficult to determine and it is consequently difficult for the user to choose an error threshold so as to achieve a certain quality. The problem is exacerbated with clustering, since the subdivision of links is amortized with time, and thus successive iterations may become much more expensive as the allowed error decreases. This sharp and unpredictable increase in iteration time may then destroy interactivity.

As a remedy we develop a progressive multi-gridding approach. By analyzing the distribution of error bounds on the links created, we can predict how many of these links would survive a given decrease in the error threshold, and thus estimate the time required for a subsequent iteration with the new error threshold. In a manner more intuitive to the user the amount of computation can be specified (in the form of a maximum number of links to refine) and the system proceeds to deduce the new threshold to use for hierarchical refinement.

This analysis can be performed at a marginal cost by recording a histogram of the distribution of error bounds computed. Specifically, the interval between 0 and the current error threshold \( \varepsilon \) is divided into a number of bins, each associated with a counter. Every time a link is created (or left unchanged) during the hierarchical subdivision procedure, we increment the counter for the bin corresponding to the link’s error bound. At the start of the next progressive multi-gridding iteration, the new error threshold is chosen such that the sum of all counter for bins with higher error levels is less than a user-specified limit \( k \). This effectively chooses an error threshold such that at most \( k \) links are refined. This multi-gridding algorithm does not accelerate the computation but guarantees a continuous update of the simulation.
7 Conclusions

Important advances towards the goal of providing interactive systems capable of treating very complex environments have been made by hierarchical radiosity and clustering algorithms. Nonetheless several important shortcomings of previous approaches were identified in this paper: (a) visibility error is typically ignored, (b) traditional error metrics do not allow the user to specify a desired level of detail and (c) progressive refinement of the simulation is difficult to achieve.

In this paper we introduced a new approach to error estimation based on illumination features, which allows the user to choose a level of detail relevant to a given simulation. The quality of a solution then relates to how well features of the user-determined size have been represented.

The principles introduced by the feature-based analysis were used to develop a multi-resolution visibility algorithm. The hierarchy constructed for clustering contains transmission information as in [13] and is further augmented with an estimate of the largest equivalent blocker size from its contents. This information is used to limit the cost of visibility calculations. An algorithm which efficiently constructs a suitable hierarchy was also presented. The results of the implementation for isotropic environments show significant computational speedup using MR visibility when the user does not require the accurate representation of small features.

Two additional quality control mechanisms were introduced: intra-cluster visibility which corrects potential light-transfer error suffered by previous clustering algorithms, and progressive multi-gridding which is essential for interactive clustering systems.

We believe that the introduction of feature-based error and quality evaluation is an important step which will lead to significant acceleration of global illumination algorithms. Multi-resolution visibility is an example of such an achievement.

In future work the extension of our approach to non-isotropic environments must be completely developed. Promising first results in representing directional information for clustering have been obtained [14]. We have not yet addressed the analysis of error caused by the use of extinction coefficients and the effect of visibility correlations between clusters and their contents. Research in these areas is extremely important for the development of reliable quality controls. It will be interesting to observe the results of the application of our approach to Monte Carlo methods. A more in-depth study of feature-based error metrics must be performed. Finally better algorithms for hierarchy construction should be investigated.

8 Acknowledgements

George Drettakis was hosted successively by INRIA (Grenoble, France), UPC (Barcelona, Spain) and GMD (St. Augustin, Germany) as an ERCIM postdoctoral fellow, and was financed in part by the Commission of the European Communities. Pat Hanrahan provided parts of the hierarchical radiosity code; Jean-Daniel Boissonnat suggested the use of erosion/expansion operations. The scene in Figure 6 was assembled using pieces of the Berkeley Soda Hall model; thanks to Seth Teller and all members of the UC Berkeley walkthrough group.

References


A Visibility error using $L^1$ and $L^2$ norms

Consider a patch $P$ illuminated by a point source at point $y$. To quantify the visibility error on the receiver patch, we compute the $L^1$ and $L^2$ norms of the difference between the visibility function $v(x, y)$ defined for $x \in P$, and its average value $\bar{v}$ over patch $P$. If $P^+$ is the region of patch $P$ that receives light, $\bar{v}$ is equal to the ratio of the areas of $P^+$ and $P$. Separating the integrals into one over $P^+$ and one over $(P - P^+)$, we find

$$\|v - \bar{v}\|_1 = 2\bar{v}(1 - \bar{v})$$

and similarly for the $L^2$ norm

$$\|v - \bar{v}\|_2 = \sqrt{\bar{v}(1 - \bar{v})}$$

Note that both estimates only depend on the average visibility across patch $P$, not on the distribution of the visibility function. Also note the dependency in $\bar{v}(1 - \bar{v})$, yielding a small error for either almost complete occlusion or almost complete visibility.