

View-Dependent Information Theory Quality Measures for Pixel Sampling and Scene Discretization in Flatland

Jaume Rigau
Institut d'Informàtica i Aplicacions
Universitat de Girona
jaume.rigau@udg.es

Philippe Bekaert
Max-Planck-Institut fuer Informatik
philippe@mpi-sb.mpg.de

Miquel Feixas
Institut d'Informàtica i Aplicacions
Universitat de Girona
miquel.feixas@udg.es

Mateu Sbert
Institut d'Informàtica i Aplicacions
Universitat de Girona
mateu.sbert@udg.es

Abstract

In this paper we present view-dependent information theory quality measures for pixel sampling and scene discretization in flatland. The measures are based on a definition for the mutual information of a line, and have a purely geometrical basis. Several algorithms exploiting them are presented and compare well with an existing one based on depth differences.

1. Introduction

Recently from the image based rendering and interactive rendering fields the need to measure the pixel quality or the triangle quality out of a mesh has appeared.

In [6], several images were blended to get the new one corresponding to the virtual camera. The color of a pixel of each image to be blended was weighted with a quality factor, which included the cosines of the angle between the normal to the surface and the viewing direction.

Darsa [3] uses an adaptive sampling strategy based on a priority schema that takes two factors into consideration, the difference in color between adjacent samples and the size of cells. The Gouraud-shaded constant color cells are triangles constructed using a Voronoi diagram or Delaunay triangulation, where each Voronoi edge corresponds to a pair of adjacent samples. In [4] they introduce into the sampling strategy the inverse of the depth, this is, they say "the mesh should approximate the object edges, and its sampling density should be inversely proportional to the depth".

In [11] a spherical Delaunay triangular mesh was created according to a priority schema. The priority was an estimate

on how well a Gouraud-shaded triangle approximates the corresponding portion of the sample environment relative to the current view. The priority value was computed weighting two measures: a contrast measure for colour differences and a depth value based on depth differences relative to the current view. This scheme was used to select the next pixels to sample.

We will try to give here a unifying approach to deal with both pixel quality and priority schema sampling, based on information theory. Our approach is based on the definition for the mutual information of a line in flatland. For the moment being our approach covers only the geometry measure, but we are working on an extension to colour. We will show how the mutual information based measures are a natural representation for the quality of a pixel or a mesh, and that previous measures can be considered as approximating steps.

The organization of this paper is as follows: In section 2 we present our previous work about information theory measures in a 2D scene. In section 3 we introduce a view-dependent discretization algorithm. This algorithm can also work as a priority schema. In section 4 two methods are presented in order to obtain which pixel should be sampled next based on the quality measure for a pixel. In section 5 we present our conclusions and future work.

2. Previous work

In our previous work [5, 8, 9, 7], we introduced discrete and continuous information theory measures [1, 2] of the *visibility complexity* in a scene. Next, we review the different definitions and results needed in this paper.

Discrete mutual information in flatland is defined by

$$I_s^d(S) = \sum_{i \in S} \sum_{j \in S} \ell_i F_{ij} \log \frac{F_{ij}}{\ell_j} \quad (1)$$

where S is the set of patches of the scene ($|S| = n_p$), F_{ij} is the form factor between the patches i and j , and $\ell_i = \frac{L_i}{L_T}$ is the relative length of patch i (L_i is the length of patch i and L_T is the total length of the scene or the sum of segment lengths).

Continuous mutual information in flatland is defined by

$$I_s^c = \iint_{x,y \in \mathcal{L}} \frac{F(x,y)}{L_T} \log(L_T F(x,y)) dL_x dL_y \quad (2)$$

where \mathcal{L} is the set of segments that forms the environment, x and y are points on segments of the environment and $F(x,y)$ is the differential form factor between x and y . This integral can be solved by Monte Carlo integration [5] and the computation can be done efficiently by casting global lines uniformly distributed upon segments [10] (see figure 1). Thus, continuous mutual information can be approximated by

$$\begin{aligned} I_s^c &\simeq \frac{1}{N} \sum_{k=1}^N \log(L_T F(x_k, y_k)) \\ &= \frac{1}{N} \sum_{k=1}^N \log\left(\frac{L_T \cos \theta_{x_k} \cos \theta_{y_k}}{2d(x_k, y_k)}\right) \end{aligned} \quad (3)$$

where θ_{x_k} and θ_{y_k} are the angles which the normals at x_k and y_k form with the segment joining x_k and y_k , $d(x_k, y_k)$ is the distance between x_k and y_k , N is the total number of pairs of points considered, which is equal to the total number of intersections divided by two, and the value of $F(x,y)$ is $\frac{\cos \theta_x \cos \theta_y}{2d(x,y)}$ for mutually visible points and zero otherwise.

Both measures, discrete and continuous mutual information, express the average information transfer in a scene. In [5, 8] continuous scene visibility mutual information has been proposed as an absolute measure of the complexity of scene visibility and discrete mutual information as a complexity measure of visibility of a discretized scene.

We define next *discrete* and *continuous mutual information field at a point* in flatland [7] which is related to the increase of complexity when an object is inserted at this point (see [9]). Note that we deal with the interior points of a 2D scene, which belong to the region strictly contained between the enclosure and the objects included in this one.

The contribution of patch $i \in S$ to the *discrete mutual information field at point* x is

$$I_p^d(x, i) = F_p(x, i) \log\left(\frac{F_p(x, i)}{\ell_i}\right) \quad (4)$$

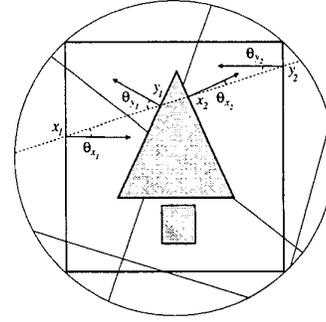


Figure 1. Global lines are used to compute I_s^c . Lines are generated using a random point on a random diameter.

where

$$F_p(x, i) = \bar{\phi}(x, i) \quad (5)$$

represents the fraction of random lines that exiting from point x hit patch i or the normalized visible angle $\bar{\phi}(x, i)$ subtended by patch i [7]. Thus, given a discretization S , the *discrete mutual information field at point* x is given by

$$I_p^d(x, S) = \sum_{i \in S} I_p^d(x, i) \quad (6)$$

The contribution of patch $i \in S$ to the *continuous mutual information field at point* x is

$$I_p^c(x, i) = \int_{y \in \mathcal{L}_i} F_p(x, y) \log(L_T F_p(x, y)) dL_y \quad (7)$$

where $F_p(x, y)$ represents the fraction of random lines that exiting from point x hit differential dL_y or the relative visible differential angle subtended by differential dL_y [7].

$I_p^c(x, i)$ can be computed approximately by casting random lines from x in all directions (similar to rays from a point light source, see figure 2):

$$I_p^c(x, i) \simeq \frac{1}{N} \sum_{k=1}^{N(i)} \log\left(\frac{L_T \cos \theta_{y_k}}{2\pi d(x, y_k)}\right) \quad (8)$$

where N is the total number of lines cast and $N(i)$ is the number of lines that hit patch i .

Thus, given any discretization S of a scene, *continuous visibility mutual information field at point* x is given by

$$I_p^c(x) = \sum_{i \in S} I_p^c(x, i) \quad (9)$$

Note that I_p^c is independent of any discretization and can be computed by the above method:

$$I_p^c(x) \simeq \frac{1}{N} \sum_{k=1}^N \log\left(\frac{L_T \cos \theta_{y_k}}{2\pi d(x, y_k)}\right) \quad (10)$$

where N is the number of lines that hit the scene and $N = \sum_{i \in S} N(i)$.

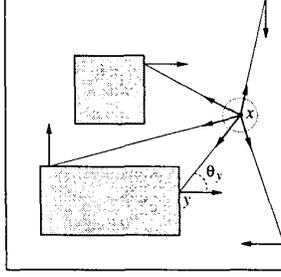


Figure 2. Random lines cast from a point x .

3. View-dependent discretization

In this section we present a proposition that allows us to derive a subdivision algorithm based on the mutual information field at a point (6,9). This algorithm will be applied at different points of two different 2D scenes in order to show the feasibility of our approach.

3.1. Continuous versus discrete mutual information at point x

In [5] we showed that when a patch is refined into n sub-patches, discrete mutual information of a scene increases or remains the same, and continuous mutual information is the least upper bound to discrete mutual information: $I_s^d \leq I_s^c$. The optimal discretization corresponds with the one with minimal loss of information, or vice versa, the one with the highest mutual information.

Now, from the complexity measures at a point, we present a similar proposition:

Proposition 1 *When a patch $i \in S$ is refined into n sub-patches, the discrete mutual information at point x increases or remains the same.*

This proof is shown in appendix A.

Corollary 1 *For any discretization S of a scene, the continuous mutual information at point x is the least upper bound to discrete mutual information at point x : $I_p^d(x, S) \leq I_p^c(x)$.*

Proof: From the above proposition and as

$$\lim_{|S| \rightarrow \infty} I_p^d(x, S) = I_p^c(x)$$

we have that $I_p^c(x)$ is the least upper bound to $I_p^d(x, S)$.

3.2. Subdivision algorithm

From the results in the previous section, we propose a subdivision algorithm from a viewpoint (interior point of a scene).

As the difference $I_p^c(x) - I_p^d(x, S)$ represents the loss of information due to the discretization and our goal is to obtain the maximum information with the minimum number of patches, we choose to subdivide the patch i that has the maximum loss of information, that is, the maximum potential information gain ($I_p^c(x, i) - I_p^d(x, i)$).

We define the gradient over patch i by

$$\Delta(x, i) = I_p^c(x, i) - I_p^d(x, i) \quad (11)$$

and the global gradient of the scene by

$$\Delta(x, S) = \sum_{i \in S} \Delta(x, i) \quad (12)$$

We also introduce the *discretization accuracy* as the quotient

$$\delta(x, S) = \frac{I_p^d(x, S)}{I_p^c(x)} \quad (13)$$

which takes values from 0 to 1.

In the following subdivision algorithm the patches are successively divided into two equal parts. The discretization process will finish when a given discretization accuracy is reached. Also it could be possible to end the algorithm when a given number of patches is achieved.

1. Input set of scene patches S
2. Input point of view x
3. Input accuracy ϵ
4. $I_p^d(x, S) \leftarrow 0$
5. $I_p^c(x) \leftarrow 0$
6. For each $i \in S$, do
 - (a) Compute $F_p(x, i)$ from (5)
 - (b) Compute $I_p^d(x, i)$ from (4)
 - (c) Compute $I_p^c(x, i)$ from (8)
 - (d) $\Delta(x, i) \leftarrow I_p^c(x, i) - I_p^d(x, i)$ from (11)
 - (e) $I_p^d(x, S) \leftarrow I_p^d(x, S) + I_p^d(x, i)$ from (6)
 - (f) $I_p^c(x) \leftarrow I_p^c(x) + I_p^c(x, i)$ from (9)
7. $\delta \leftarrow \frac{I_p^d(x, S)}{I_p^c(x)}$ from (13)
8. While $\delta < \epsilon$, do

- (a) $k \leftarrow \text{Index of } \max_{i \in \{1..|S|\}} (\Delta(x, i))$
- (b) Subdivision of patch $k \in S$ into $\{k_0, k_1\}$
- (c) $S \leftarrow (S - \{k\}) \cup \{k_0, k_1\}$
- (d) Compute $(F_p(x, k_0), F_p(x, k_1))$
- (e) Compute $(I_p^d(x, k_0), I_p^d(x, k_1))$
- (f) Compute $(I_p^c(x, k_0), I_p^c(x, k_1))$
- (g) $I_p^d(x, S) \leftarrow I_p^d(x, S) - I_p^d(x, k) + I_p^d(x, k_0) + I_p^d(x, k_1)$
- (h) $\Delta(x, k_0) \leftarrow I_p^c(x, k_0) - I_p^d(x, k_0)$
- (i) $\Delta(x, k_1) \leftarrow I_p^c(x, k_1) - I_p^d(x, k_1)$
- (j) $\delta \leftarrow \frac{I_p^d(x, S)}{I_p^c(x)}$

9. Output new set of scene patches S

3.3. Results

The behaviour of this algorithm is shown in figures 3, 4 and 5, and table 1. These figures represent different discretizations from different viewpoints. As we can observe, the discretization is finer in the discontinuity regions and also on the patches which are not well oriented with respect to the point. Over these patches we obtain (11) a higher gradient Δ , basically due to the difference of distances between the viewpoint and the points on a patch.

scene	lines cast	n_p	I_p^d	I_p^c	100δ
3(a)	10^4	120	1.6105	1.6122	99.90
3(b)	10^4	120	1.6266	1.6279	99.92
3(c)	10^4	120	2.2987	2.2999	99.95
4(a)	10^5	224	2.7548	2.7681	99.52
4(b)	10^5	224	3.1847	3.1884	99.89

Table 1. Final values obtained by the subdivision algorithm for the scenes of figures 3 and 4. n_p is the final number of patches, I_p^d and I_p^c are the discrete and continuous mutual information field at a point respectively, and 100δ is the accuracy achieved.

In figure 5(a) and 5(b) we show the convergence of I_p^d towards I_p^c for the scenes of figures 3 and 4 respectively. Table 1 shows the values corresponding to the final discretization obtained. In figure 5, the points marked represent the number of patches in which $\delta = 0.99$ is obtained. Note that, in the examples presented, a good discretization is achieved first from the most complex viewpoints. To compute the subdivision of figures 3 and 4 we cast 10^4 and 10^5 random lines respectively.

4. Pixel sampling

In this section we define the quality of a pixel and give two heuristics based on this definition. These are applied to pixel sampling and compared with a heuristic based on depth difference [11].

4.1. Pixel quality measures

We consider here pixels defined on a circle instead of a line, although there is practically no difference in the final results. First we define continuous mutual information for a pixel.

Pixel continuous mutual information or *pixel quality* is defined by

$$\begin{aligned}
 & I_p^c(x, p_i) \\
 &= I_p^c(x, S(x, p_i)) \\
 &= \int_{y \in S(x, p_i)} F_p(x, y) \log(L_T F_p(x, y)) dL_y \quad (14)
 \end{aligned}$$

where p_i is a pixel and $S(x, p_i)$ is the set of patches visible through pixel p_i from point x .

In a similar way we have shown in the previous work, pixel quality can be computed casting random lines from viewpoint x to the scene through pixel p_i :

$$I_p^c(x, p_i) \simeq \frac{1}{N} \sum_{k=1}^{N(x, p_i)} \log\left(\frac{L_T \cos \theta_{y_k}}{2\pi d(x, y_k)}\right) \quad (15)$$

where N is the total number of lines cast, $N(x, p_i)$ is the number of random lines through pixel p_i from point x , and $N = \sum_i N(x, p_i)$.

Pixel line quality is given by

$$I_p^c(x, p_i, k) = \log\left(\frac{L_T \cos \theta_{y_k}}{2\pi d(x, y_k)}\right) \quad (16)$$

where $k \in \{1..N(x, p_i)\}$, and represents the contribution of each line to the pixel quality.

4.2. Sampling heuristics

We present two different heuristics which will be evaluated in comparison to a depth heuristic defined in [11] as

$$\Gamma(x, p_i) = 1 - \frac{D_{\min}}{D_{\max}} \quad (17)$$

where D_{\min} and D_{\max} are respectively the minimum and maximum distances from viewpoint x to the scene through pixel p_i .

This heuristic was used to drive a priority schema for pixel sampling purposes. It was the geometrical part of a combined color and depth difference measure.

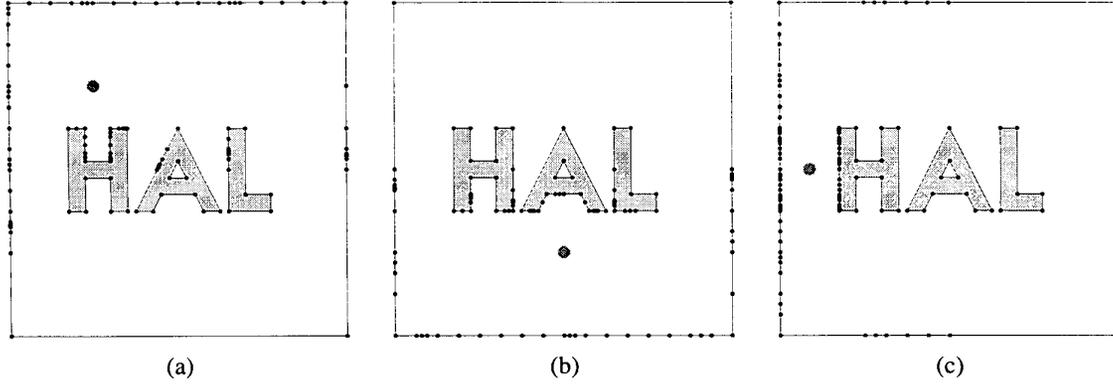


Figure 3. Three different discretizations of the same scene from different viewpoints (represented by a little circle) obtained by casting 10^4 lines. The final values are shown in table 1.

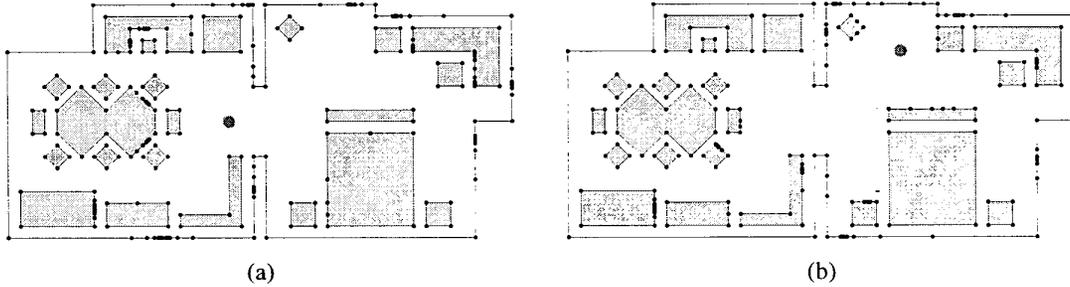


Figure 4. Two different discretizations of the same scene from different viewpoints (represented by a little circle) obtained by casting 10^5 lines. The final values are shown in table 1.

Our first approach calculates for each pixel the sum of differences between successive pixel line qualities which can be expressed by

$$\begin{aligned}
 & \Lambda(x, p_i) \\
 &= \sum_{k=1}^{N(x, p_i)-1} |I_p^c(x, p_i, k) - I_p^c(x, p_i, k+1)| \\
 &= \sum_{k=1}^{N(x, p_i)-1} \left| \log\left(\frac{L_T \cos \theta_{y_k}}{2\pi d(x, y_k)}\right) - \log\left(\frac{L_T \cos \theta_{y_{k+1}}}{2\pi d(x, y_{k+1})}\right) \right| \\
 &= \sum_{k=1}^{N(x, p_i)-1} \left| \log\left(\frac{d(x, y_{k+1}) \cos \theta_{y_k}}{d(x, y_k) \cos \theta_{y_{k+1}}}\right) \right| \quad (18)
 \end{aligned}$$

This measure can be interpreted as representing the orography of the region seen through the pixel and will be used to obtain the proportion of lines cast for each pixel.

The second approach takes only into account the difference between maximum and minimum line quality for each

pixel and is expressed by

$$\Upsilon(x, p_i) = \max_{k=1}^{N(x, p_i)} (I_p^c(x, p_i, k)) - \min_{k=1}^{N(x, p_i)} (I_p^c(x, p_i, k)) \quad (19)$$

This is an analogous to the depth difference measure and can be interpreted as representing the gradient of the region seen through the pixel. As in the first approach, the number of lines cast for each pixel is proportional to this value.

While the distance heuristic only considers the distance between the viewpoint and the segment, in our approaches, pixel line complexity conveys two kinds of information: distance and orientation. So, our two heuristics discriminate discontinuities and also orientation changes (vertices). To see this, pixel line quality can be split in the following way

$$\begin{aligned}
 & I_p^c(x, p_i, k) \\
 &= \log\left(\frac{L_T \cos \theta_{y_k}}{2\pi d(x, y_k)}\right)
 \end{aligned}$$

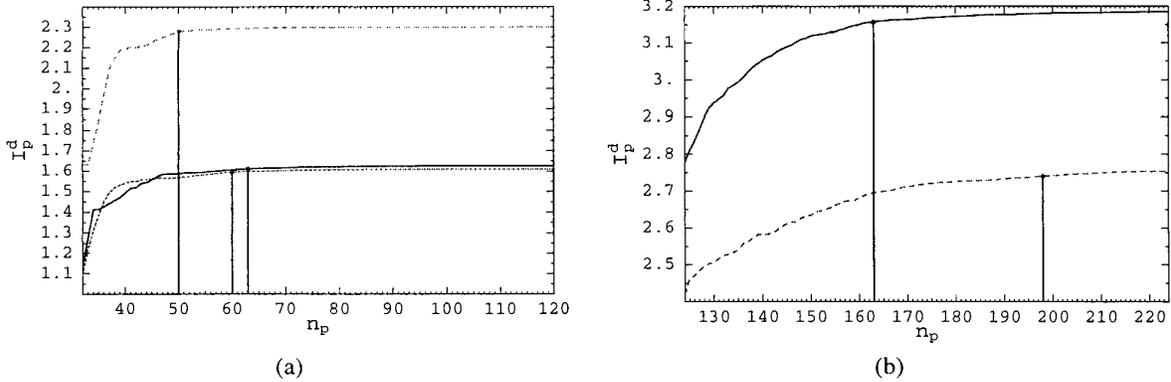


Figure 5. Plots (a) and (b) correspond to figures 3 and 4 respectively. In (a), dot, continuous, and dashed lines correspond to scenes 3(a,b,c), and an accuracy $\delta = 0.99$ is achieved with $n_p = (60, 63, 50)$ respectively. In (b), dashed, and continuous lines correspond to scenes 4(a,b), and the same accuracy is achieved with $n_p = (198, 163)$ respectively.

$$= \log(\cos \theta_{y_k}) + \log\left(\frac{L_T}{2\pi d(x, y_k)}\right) \quad (20)$$

where the first term in (20) represents the patch orientation contribution and the second represents the distance contribution. In this way our approach can be seen as the combination of the ones based on orientation and depth, respectively.

4.3. Results

In figures 6 and 7 we show the results of applying the above three heuristics. Figures (a), (b), and (c) are obtained respectively from the depth (Γ), sum of differences (Λ), and gradient (Υ) heuristics.

In the images that correspond to the depth heuristic (figures 6(a) and 7(a)) we observe that some edges remain unsampled as, contrary to ours, the depth heuristic does not capture a change in orientation from edge to edge. Compare for instance the corners in the bedroom in figures 7(ii). In general our heuristics behave better by obtaining a finer sampling where it is more needed. On the other hand a possible drawback in our approaches can appear when the distance factor is balanced with the orientation factor. This happens for instance at the **L** corner in figures 6(i). We are currently considering a solution to this problem by taking separately the two components of the measure (20).

5. Conclusions and future work

We have presented in this paper several view-dependent information theory based quality measures for pixel sampling and scene discretization in flatland. The measures are

based on the definition for the mutual information of a line. Results show that they are very promising and compare well with depth measures. Although this is a purely geometrical approach we plan in our future work to extend it to deal also with color. The extension to 3D scenes, although not yet implemented, is straightforward.

6. Acknowledgements

This project has been funded in part with grant numbers TIC-98-586-C03 and TIC-98-973-C03 of the Spanish Government and with a Spanish-Austrian joint action number HU-1998-0015 of the *Ministerio de Educación y Cultura*.

A Proof of the proposition

Let us imagine a point x in a discretized scene where $|S| = n_p$ and discrete mutual information field at point x is $I_p^d(x, S)$. If any patch i is divided into $n > 0$ subpatches $S_i = \{i_k | k \in \{1..n\}\}$, we have to show that discrete mutual information at point x of the new discretization $S' = (S - \{i\}) \cup S_i$ (where $|S'| = n_p + n - 1$) fulfills $I_p^d(x, S) \leq I_p^d(x, S')$.

First, we show that $I_p^d(x, i) \leq I_p^d(x, S_i)$. From the following equalities $F_p(x, i) = \sum_{k=1}^n F_p(x, i_k)$ and $\ell_i = \sum_{k=1}^n \ell_{i_k}$ we obtain

$$\begin{aligned} I_p^d(x, i) &\leq I_p^d(x, S_i) \\ &\equiv F_p(x, i) \log\left(\frac{F_p(x, i)}{\ell_i}\right) \leq \sum_{k=1}^n I_p^d(x, i_k) \end{aligned}$$

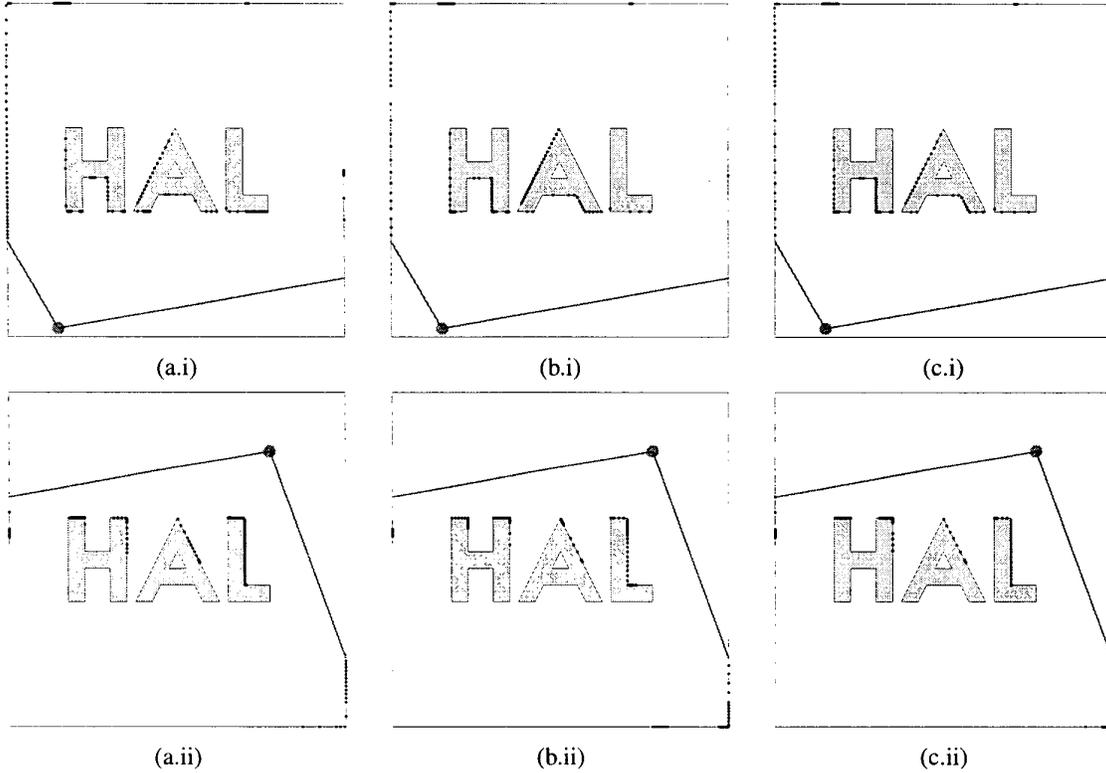


Figure 6. The same scene with six different pixel samplings. Viewpoint is represented by a little circle. Field of view has 110° and 100° for figures (i) and (ii) respectively, 150 lines have been cast and 30 pixels are used. Figures (a), (b), and (c) correspond to Γ , Λ , and Υ heuristics respectively.

$$\begin{aligned}
 &\equiv \left(\sum_{k=1}^n F_p(x, i_k) \right) \log \left(\frac{\sum_{k=1}^n F_p(x, i_k)}{\sum_{k=1}^n \ell_{i_k}} \right) && \equiv \left(\sum_{k \in S - \{i\}} I_p^d(x, k) \right) + I_p^d(x, i) \\
 &\leq \sum_{k=1}^n F_p(x, i_k) \log \left(\frac{F_p(x, i_k)}{\ell_{i_k}} \right) && \leq \left(\sum_{k \in S - \{i\}} I_p^d(x, k) \right) + \left(\sum_{k \in S_i} I_p^d(x, k) \right) \\
 &\equiv True && \equiv I_p^d(x, i) \leq I_p^d(x, S_i) \\
 & && \equiv True
 \end{aligned}$$

The last step is true for the concavity of the logarithm function for non-negative numbers:

$$\left(\sum_{k=1}^n a_k \right) \log \left(\frac{\sum_{k=1}^n a_k}{\sum_{k=1}^n b_k} \right) \leq \sum_{k=1}^n a_k \log \left(\frac{a_k}{b_k} \right)$$

where $(a_k, b_k) = (F_p(x, i_k), \ell_{i_k})$.

Now we have

$$\begin{aligned}
 &I_p^d(x, S) \leq I_p^d(x, S') \\
 &\equiv \sum_{k \in S} I_p^d(x, k) \leq \sum_{k \in S'} I_p^d(x, k) \\
 &\equiv \sum_{k \in (S - \{i\}) \cup \{i\}} I_p^d(x, k) \leq \sum_{k \in (S - \{i\}) \cup S_i} I_p^d(x, k)
 \end{aligned}$$

References

- [1] R. Blahut. *Principles and Practice of Information Theory*. Addison-Wesley, 1987.
- [2] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications, 1991.
- [3] L. Darsa and B. C. Silva. Multi-resolution representation and reconstruction of adaptively sampled images. In *Proceedings of IX Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'96)*, pages 321–328, October 1996.
- [4] L. Darsa, B. C. Silva, and A. Varshney. Navigating static environments using image-space simplification and morphing.

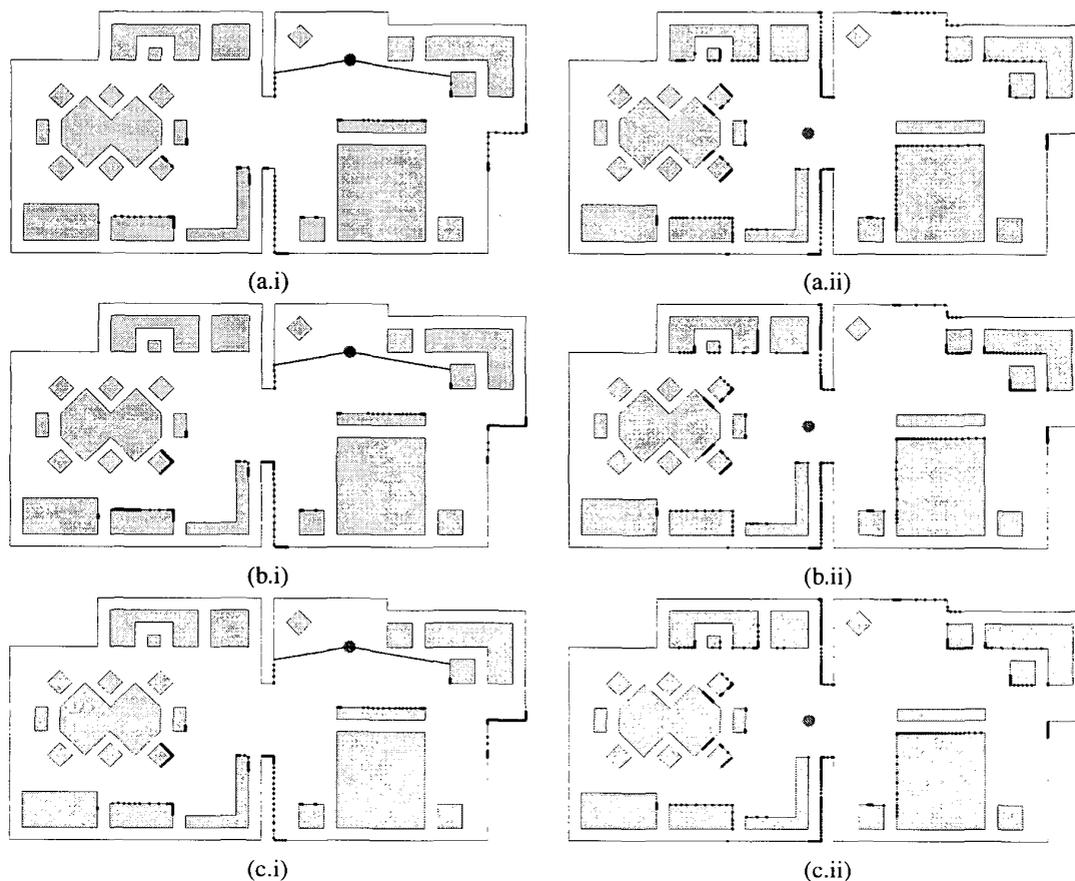


Figure 7. The same scene with six pixel samplings. Viewpoint is represented by a little circle. For figures (i) the field of view has 190° , 200 lines have been cast, and 40 pixels are used, and for figures (ii) the field of view has 360° , 400 lines have been cast, and 80 pixels are used. Figures (a), (b), and (c) correspond to Γ , Λ , and Υ heuristics respectively.

1997 Symposium on Interactive 3D Graphics, pages 25–34, April 1997.

- [5] M. Feixas, E. del Acebo, P. Bekaert, and M. Sbert. An information theory framework for the analysis of scene complexity. *Computer Graphics Forum (Proceedings of Eurographics'99)*, 18(3):95–106, September 1999.
- [6] K. Pulli, M. F. Cohen, T. Duchamp, H. Hoppe, L. Shapiro, and W. Stuetzle. View-based rendering: Visualizing real objects from scanned range and color data. In J. Dorsey and P. Slusallek, editors, *Rendering Techniques'97 (Proceedings of the 8th Eurographics Workshop on Rendering)*, pages 23–34, Vienna, Austria, June 1997. Springer-Verlag Wien-New York. Held in St. Etienne, France.
- [7] J. Rigau, M. Feixas, and M. Sbert. Information theory point measures in a scene. Research Report IliA-00-08-RR, Institut d'Informàtica i Aplicacions, Universitat de Girona, Girona, Spain, 2000.
- [8] J. Rigau, M. Feixas, and M. Sbert. Scene visibility complexity in flatland. Research Report IliA-00-03-RR, Institut d'Informàtica i Aplicacions, Universitat de Girona, Girona, Spain, 2000.
- [9] J. Rigau, M. Feixas, and M. Sbert. Visibility complexity of a region in flatland. In *Short Presentations of Eurographics'00*, pages 159–163, Interlaken, Switzerland, August 2000.
- [10] M. Sbert. *The Use of Global Random Directions to Compute Radiosity. Global Monte Carlo Methods*. PhD thesis, Universitat Politècnica de Catalunya, Barcelona, Spain, November 1996.
- [11] M. Simmons and C. H. Séquin. Tapestry: A dynamic mesh-based display representation for interactive rendering. In B. Péroche and H. Rushmeier, editors, *Rendering Techniques 2000 (Proceedings of the 11th Eurographics Workshop on Rendering)*, pages 329–340, Vienna, Austria, June 2000. Springer-Verlag Wien-New York.