Multiresolution Representation and Reconstruction of Adaptively Sampled Images

LUCIA DARSA
BRUNO COSTA

Instituto de Matemática Pura e Aplicada - IMPA
Rio de Janeiro, Brazil

SUNY at Stony Brook
New York, USA

{lucia|bruno}@visgraf.impa.br

Abstract. We present a method for sampling, structuring and reconstructing progressively refined images effectively. This has direct applications to progressive refinement of the result of slow renderers, such as ray tracers, and to the progressive transmission of images on slow media, as in world wide web browsers. Sampling images adaptively with respect to frequency at the back end provides a more compact representation. These samples are then structured at the front end using Voronoi or Delaunay tesselations, and reconstructed with C^0 continuity taking advantage of standard 3D acceleration hardware. The structuring scheme is extended to handle multiresolution images, with optional antialiasing, allowing the representation of an arbitrary level of detail without penalizing the reconstruction.

1 Introduction

Images are the result of most processes in computer graphics. The demand for quality has greatly increased the amount of data to be manipulated, transmitted and stored. Proper image representation is a key concern in performing these operations efficiently. One must be able to transform an image into a certain representation and to transform it back, while understanding the characteristics and controlling the effects of the mappings. Different terminology applies to these transformations depending on the application: encoding and decoding, compression and decompression, sampling and reconstruction. Even though the applications may differ in their essential requirements, they build upon a set of common ideas, and have some overlap in their usage.

There are very efficient methods for compression of images available, such as JPEG (Pennebaker and Mitchell, 1993), Wavelet based (Froment and Mallat, 1992) and Fractal based (Barnsley and Hurd, 1993), which emphasize the bits per pixel rate.

In interactive applications, however, if the intermediate steps of the decoding can be displayed quickly as meaningful images—as is the case of image pyramids (Williams, 1983), for instance—the output of the data as it becomes available can result in a faster feedback, even if the compression rate is smaller, or even negative.

The order in which the information of the image is decoded, or the order in which the samples are generated, also plays an important role in the perceptual "speed". If important areas of the image, such as the image edges, are displayed first, the fact that the less important parts of the image are not fully detailed yet may be overlooked. This concept is central to progressive refinement techniques (Bergman et al., 1986).

Most of the image decoding schemes, however, are based on uniform sampling, and represent the image as a matrix of samples. This results in progressive refinement schemes that improve images on a top down scan. Our method, instead, balances the overall improvement of the image with the emphasis on the perceptually important features, in an irregular scanning pattern.

One of the applications that can take advantage from any such representation is the transmission of images on relatively slow media, where the progressive refinement of the quality of the image is a crucial user feedback feature. Some forms of progressive refinement are present in most web browsers, for instance, but they are based on traditional image representations.

Slow renderers used interactively—especially ray tracers, which are based on individual samples—also benefit from progressive techniques. Although progressive refinement is available in the majority of the packages, they often work in a regular fashion, or do not perform a suitable reconstruction for display.

Our method allows a good quality reconstruction to be performed quickly and locally, with the added processing cost hopefully worth the improvement in feedback. It is particularly useful when the placement and ordering of the samples cannot be precomputed by the encoding algorithm—for instance, when the sample positions are generated dynamically for interactive display.

Image Representations

Traditional image representation employs a straightforward regular sampling strategy, which facilitates most of the tasks involved. The regular structuring of the samples in a matrix is conveniently simple, having given rise to the raster display paradigm, which makes this representation specially efficient due to the tight relationship with typical hardware.

The regular sampling strategy, however, does not necessarily match the information contents of the image. If high precision is required, the global sampling resolution must be increased, often resulting in excessive sampling in some areas. Needless to say, this can become very inefficient, especially if the fine/coarse detail ratio is low.

Many image representation schemes address this problem, most notably frequency domain codifications (Pennebaker and Mitchell, 1993) (Froment and Mallat, 1992), quad-tree based image models (Samet, 1984) and fractal image compression (Barnsley and Hurd, 1993).

Most of these schemes use non homogeneous sampling, even though they are based on a regular rectangular grid. This originates aliasing artifacts, which are usually reduced by increasing the sampling rate and/or band-limiting the input image (Gomes and Velho, 1995). Approaches that do not use rectangular grids, such as hexagonal or triangular grids (Glassner, 1992), facilitate the avoidance of aliasing artifacts, but still rely on regular sample distribution. Irregular sampling strategies, such as an aperiodic triangulation (Radin, 1994), are significantly less prone to aliasing, although they introduce some noticeable noise.

In general, images as represented must be manipulated or transformed at least to some degree. Certain forms of transformations can be easier or harder to perform on certain representations. Most image representation schemes require resampling of the image when transforming the domain, due to the fixed sampling grids which are employed. The image transformation cycle therefore has the added cost of reconstruction and resampling phases which must be performed carefully to reduce aliasing.

2 Proposed Solution

We propose a simple and compact image representation scheme based on adaptively sampling the input signal, using multiresolution Delaunay or Voronoi tesselations as structuring primitives and using conventional 3D acceleration hardware to quickly reconstruct images for display.

The transformation of images represented in this form by continuous, one-to-one, warping maps is direct and, in general, avoids the necessity of resampling as the representation is not based on a fixed grid. This approach also results in the ability to represent images to an arbitrary level of detail, unconstrained by traditional pixels.

Moreover, the representation scheme is extended to handle multiple levels of resolution, with or without antialiasing, allowing sub-representations of arbitrary complexity to be obtained directly from the model. This shares the usual advantages of multiresolution methods, mostly due to the ability to easily adapt the amount of information in the image to the way it is used (Heckbert and Garland, 1994).

A central idea in the multiresolution scheme is that the addition of a new appropriate sample will improve the quality of the approximation. This leads naturally to incremental techniques that progressively refine the quality of the image. Also, as mentioned before, the ordering of the samples is essential to the perception of "refinement". Therefore, several perceptual issues must be taken into consideration when selecting the samples, and if that is done carefully, this form of image representation can result in an improved sense of quality progress.

In order to work with irregularly sampled images, we must first choose the sampling positions, and then reconstruct the samples to obtain an approximated image. These processes can be treated as independent, even though knowledge of each others characteristics may help. Some sort of structure must be imposed upon the samples, to allow the sampling decisions to be made in an efficient and consistently distributed way, and also to facilitate the display of the results. Therefore, we distinguish three basic questions in the problem:

- How to structure the samples?
- How to sample?
- How to reconstruct the image?

In the next section, we will review some basic concepts, and in the following sections, each of these issues will be addressed.

3 Image Model

An adequate mathematical model for a *continuous* image (Gomes et al., 1996) is a function that relates points in a subset of the Euclidean plane to colors in a color space, i.e., a continuous image is a function $f : U \subset \mathbb{R}^2 \to C$, where C is a color space, that can be considered a generic vector space containing the color space as a subspace. In practice, U is a rectangle aligned with the Cartesian axes of $\mathbb{R}^2$ and C is a one-, three- or four- dimensional color space.

A *discrete* or *point sampled image* f^* is a representation of that image, through a set of samples of f taken at a finite number of positions p_i:

$$f^* = \{\langle s_i \rangle\}, \quad s_i = (p_i, f(p_i)), \ \ p_i \in U.$$

The sample positions usually possess some form of structuring that makes a consistent representation of the samples easier. A natural structure is a regular rectangular grid, which is by far the most used representation and therefore is naturally associated with the concept of discrete images.

A more efficient representation of the continuous image f would, in general, have more samples in the regions where the function f has higher frequencies and curvature, and less samples in the lower frequency areas. This usually results in less samples for the same error in the approximation.

4 Structuring Irregular Samples

Each sample represents the image with absolute fidelity at that point. To be able to decide new sample positions and to display the samples, more complex information will be required, such as adjacency relations and distances. Thus, the samples must be structured in some way.

A natural geometric concept is the area of dominance of each sample, which is well captured by the Voronoi polygon of that sample, and will be the basis of our representation.

Given a set S of n points in a plane, the set of points p, such that p is closer to one of the given points p_i than to any other p_j of S is called the *Voronoi polygon* of p_i in S (see figure 1). In this way, each of the Voronoi polygons is the intersection of the half planes that contain p_i defined by the medians between the point p_i and each of the other points in S. The union of the Voronoi polygons of all the points in S is a planar subdivision called *Voronoi diagram* (Preparata and Shamos, 1985), such as the example in figure 1.

Note that in the diagram, inside each Voronoi cell, there is a single sample which dominates that region. Not only does this concept acommodate arbitrarily distributed samples, but the resulting shape of the cells is usually not regular, and therefore, less prone to aliasing.

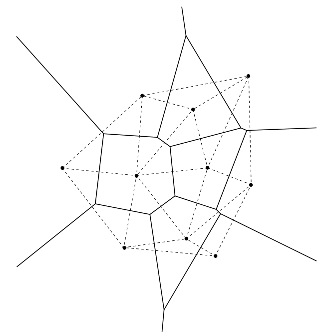


Figure 1: Voronoi diagram and its dual Delaunay triangulation

It is interesting to notice that the dual of a Voronoi diagram, the Delaunay triangulation (shown dotted in figure 1), will also be very useful, and, in fact, can be used as the structuring scheme in place of Voronoi diagrams. Since both structures can be constructed and kept simultaneously in a quad-edge structure (Guibas and Stolfi, 1985), for instance, any of them can be used interchangeably. A Delaunay triangulation is intuitively "good", especially for interpolations: since it maximizes the smallest angle of each triangle, it produces well-behaved triangles, in the sense that they are as close to equilateral as possible. This will reduce numerical interpolation problems and any discontinuities on triangle boundaries caused by C^0 interpolations.

In the next sections, we will show how Voronoi diagrams or Delaunay triangulations can be used in the sampling and reconstruction of adaptively sampled images.

Incremental Construction

There are various algorithms for the construction of Voronoi diagrams, but, in this application new samples must be added incrementally to an existing structure. The addition of each sample should be a quick process that does not interfere, as much as possible, with the overall speed of the display.

The addition of a point to a Voronoi diagram is a local operation, although it may affect the entire diagram in a worst case scenario. On the average, for randomly distributed points, it can be shown that a Voronoi polygon has only six sides, or equivalently, that each vertex of a Delaunay triangulation has six incident edges (Sibson, 1978). Any part that has to be redisplayed is also limited, by the same principle.

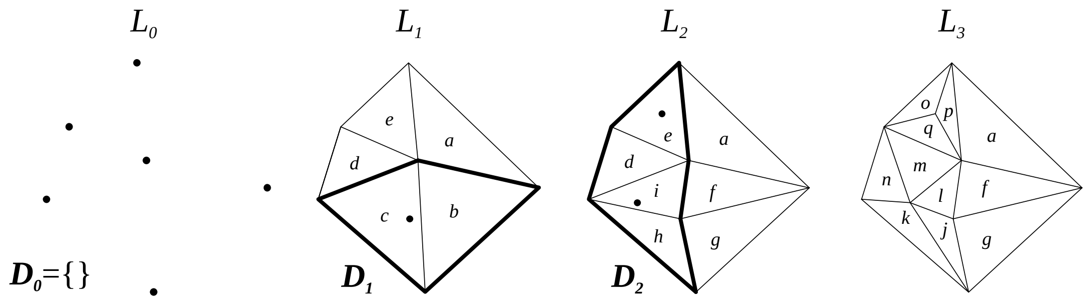


Figure 2: Levels of a multiresolution triangulation and their difference sets.

There are algorithms for incremental construction of Voronoi and Delaunay tesselations; a complete implementation is presented in (Lischinski, 1994).

Multiresolution Representation

The multiresolution representation is an extension of the basic Delaunay triangulation to a pyramidal data structure (Floriani, 1989). The idea is to compute difference sets between each level L_i and the finer level L_{i+1}, each level adding a certain number of samples to the previous level. Levels can be seen conceptually as complete triangulations, although they are not stored as such: to construct any level L_k, we start from L_0, and apply k patches sequentially.

A difference set D_i consists of the triangles of L_i that are not in L_{i+1}. A pyramid is contructed by linking each triangle t in D_i to all the triangles in L_{i+1} that intersect t. A pyramid for the levels in figure 2 is shown in figure 3.

Our representation of multiresolution images is a simple extension of that for point sampled images, obtained by the addition of a set of dividers that split the samples in levels. This can be done by indicating the number of samples for each level of detail. If n_i is the number of samples for the i^{th} level and s_j is the j^{th} sample, a multiresolution image f^μ can be described as

$$f^\mu = \{\langle s_j\rangle, \langle n_i\rangle\}, \quad s_j = (p_j, f(p_j)), \quad p_j \in U.$$

Each level is, then,

$$S_i = \langle s_{n_{i-1}+1}, \ldots, s_{n_i}\rangle.$$

Alternatively, we can say that

$$f^\mu = \langle S_1, S_2, \ldots, S_m\rangle$$

This sequence of samples can be used to construct a Delaunay pyramid as follows. We start from an empty triangulation, which we call L_0, and apply an incremental Delaunay algorithm to add all the samples in S_1 to L_0. By definition, D_0 is empty. We proceed by applying the incremental algorithm to add all the samples in S_2 to L_1 which will result in L_2 (see figure 2). The incremental algorithm can easily determine D_1, which is essentially composed by the triangles modified at that step. This process is repeated until we reach the m^{th} level.

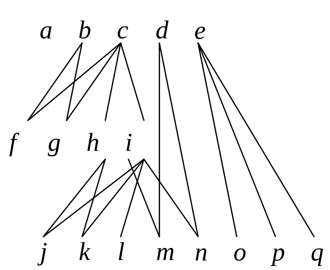
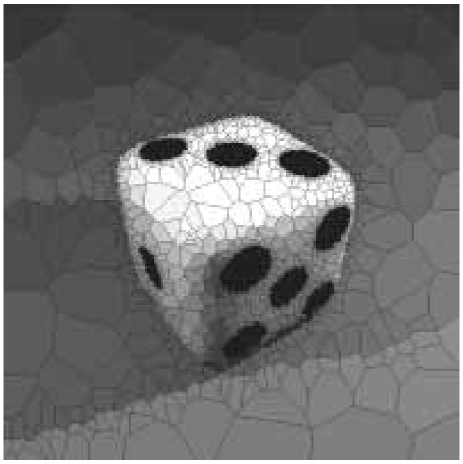


Figure 3: Links in a Delaunay pyramid.

Antialiasing

In the previous discussion, the samples in the finer levels of the multiresolution image are simply ignored in the coarser levels. We extend that technique so that all samples can be used to reduce the aliasing in the coarser levels of the representation. This corresponds to the idea of filtering the finer levels before resampling them to the lower resolution.

This implies, however, that the color of the samples will be different in different levels. With the filtering technique we propose, in going from level L_i to L_{i+1} the only samples that change their color are those adjacent in L_{i+1} to a sample in S_{i+1}, i.e., those on the boundary of the difference set D_i. Since the filtering is invertible, we do not need to change



(a)

(b)

Figure 5: 2,000 samples reconstructed with constant color Voronoi cells.

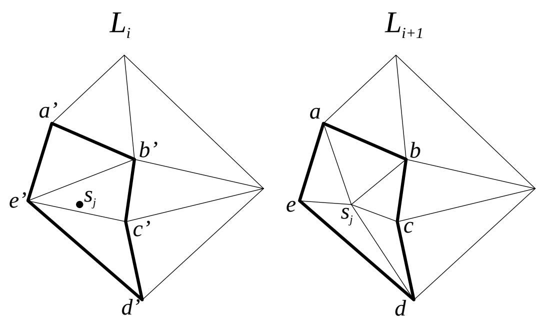


Figure 4: Region modified by the addition of s_j.

the representation of multiresolution images to account for more than one color for each sample. Only the antialiased samples are stored at each level, and when there is a change to a finer level, the colors of the neighbors of each new sample are fixed to remove its influence.

Our approach to antialiasing in the lower levels is therefore a simple propagation of the color of each sample in S_{i+1}—they are not in L_i—to its neighbors which are in L_i. Each sample $s_j \in S_{i+1}$ is considered individually. Let B_j be the set of points on the boundary of the region that was modified by the addition of s_j. In the example in figure 4, $B_j = \{a, b, c, d, e\}$. The samples in B_j will have their color in the previous level L_i modified.

The propagation is based on the Euclidean distances d_k from the sample s_j to each of the samples b_k in B_j as follows:

$$b'_k = \frac{b_k + w_k s_j}{1 + w_k}, \quad b_k \in B_j,$$

where w_k is

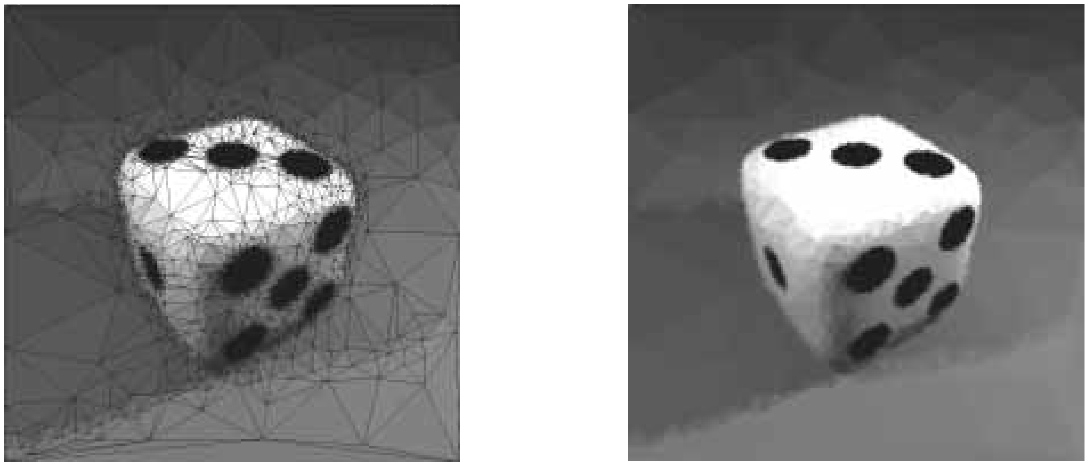
$$w_k = \frac{\frac{1}{1+d_k}}{\sum_k \frac{1}{1+d_k}}.$$

5 Adaptive Sampling Strategy

To sample adaptively, it is necessary to devise a criterion that determines where new samples may be needed. We will start from a basic set of samples, that can be just a few points that cover the domain of interest of the image. The information necessary to make further decisions will be based solely on the current set of samples, as we assume that no other information about the continuous image is available.

One way to decide where samples may be needed is a simple stack, as proposed in (de Figueiredo, 1995) for sampling parametric curves. The stack is initialized with the endpoints of the curve. The pair at the top of the stack is probed to test if inserting a new sample between them is actually necessary. If the two samples and the probe are reasonably collinear, the probe is discarded and that sample pair need not be considered again. Otherwise, two new segments will be pushed onto the stack: one containing the new sample and the last endpoint, the other containing the first endpoint and the new sample. Obviously, this process is not immune to aliasing, and if applied for curves with localized high frequency detail, it will fail to detect it. Note that this scheme generates the samples in the order they occur along the curve. In our case, we want to generate samples in the high frequency area first, which corresponds to the use a breadth first strategy, or a queue, instead of a depth first strategy.

This concept can be extended to a priority queue



(a) (b)

Figure 6: 2,000 samples reconstructed with constant color Delaunay triangles.

that makes the perceptually more important areas of the image be attacked first. We used a simple priority scheme that takes two factors into consideration: the difference in color between adjacent samples, and the size of cells. Between adjacent samples with high difference in color, there is probably an image edge. Large cells should be probed, too, since there is a high probability that significant detail was lost.

Using the Voronoi diagram or the Delaunay triangulation to structure the samples, these criteria can be easily evaluated. Each Voronoi edge corresponds exactly to a pair of adjacent samples. If the priority is evaluated as samples are added, and stored at each Voronoi edge, the edges can be placed in the priority queue. The first edge of the queue is then probed repeatedly, possibly adding more edges to the queue. Once an edge to be broken is selected, the probe position will be chosen randomly between the endpoints of the edge. The rationale behind this decision is that endpoints of Voronoi edges are equidistant from the neighboring samples, yielding a better sample distribution.

6 Reconstruction from Irregular Samples

Reconstructing data from irregular samples is a problem of sparse data interpolation. There is an extensive literature on this subject, including methods using high order interpolation schemes (Cendes and Wong, 1987). Our concern here is more on speed than on quality, since these goals are hard to achieve simultaneously. The applications that motivated this work require good speed in the incremental reconstruction of images specifically for display.

The simplest form of reconstruction of irregularly sampled images is actually based on Voronoi diagrams: each cell is filled with the color information of the sample that dominates it. A similar process has been used for artistic results in (Haeberli, 1990). Although the reconstructed image has discontinuities that translate into artificial high frequencies, it can be efficiently displayed using hardware polygon filling in the screen space to fill each Voronoi cell. This form of reconstruction is illustrated in Figure 5, that shows an 800x800 image sampled at 2,000 positions (0.3% of the final resolution).

Moreover, the addition of a new Voronoi cell causes only the filling of the newly created polygon, as it does not alter the shape of any other cell besides its neighbors. This is important in the refresh

Figure 7: 2,000 samples reconstructed with Gouraud shaded Delaunay triangles.

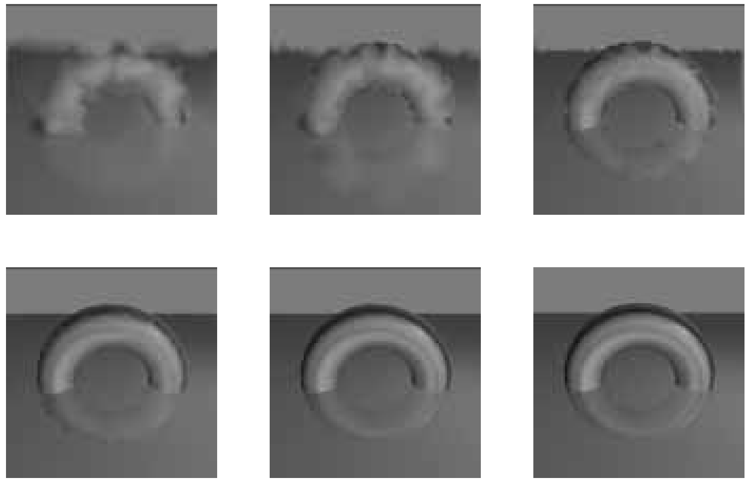


Figure 9: Gouraud reconstruction with progressive number of samples: 100, 240, 800, 2,000 and 8,200. The last frame is a normal raster image with 90,000 samples.

of progressively displayed images, which will be also incremental.

Another alternative is to fill each triangle of the dual Delaunay triangulation with a constant color. Since each Delaunay triangle has three samples at the corners, they must be averaged to yield a suitable color for the triangle. Although this method is also discontinuous, it yields a more blurred reconstruction, which seems adequately smoother. Since there are much more Delaunay triangles than Voronoi cells for a given set of samples, although the color information of the samples is the same, it is more distributed in the Delaunay triangulation. Figure 6 shows the same 2,000 samples reconstructed with constant color Delaunay triangles.

Figure 8: 15,000 samples reconstructed with Gouraud shaded Delaunay triangles.

For improved results, a continuous reconstruction should be used. The simplest one is a linear reconstruction, which yields a C^0 continuous result.

A C^0 image reconstruction can be implemented by linearly interpolating the color values inside each Delaunay triangle. Gouraud shading performs exactly this type of interpolation, and is available in standard 3D acceleration hardware. This means that linearly interpolated triangles can be painted almost as fast as flat shaded triangles. Figure 7 shows again the same 2,000 samples, reconstructed with Gouraud shaded Delaunay triangles.

Although higher order reconstructions are definitely better, linear filtering provides a quick, low-error approximation that works well for progressive quality refinement (see figure 9). At very low resolutions, artifacts due to the linear interpolation are clearly noticeable, but they tend to disappear at finer levels. This is especially true for synthetic images, which can be well represented with such approximations (Velho and Alvarenga, 1990).

7 Conclusion and Future Work

We have presented a technique for the representation of multiresolution images based on adaptive sampling, and methods for their reconstruction that can be performed quickly with hardware assistance.

The main applications of this technique are where interactive feedback is needed, such as in interactive rendering and transmission of images. We have implemented it as a quick, good quality preview mode for a ray tracer, and we are working on a WWW browser extension that allows the progressive decoding of such images.

We are currently working on other forms of adaptive sampling heuristics, which take advantage of knowing how the reconstruction will be performed. This includes properties of linear interpolation, which can be combined with estimates of the image derivatives to yield representations with less error (Rippa, 1992), and attempts to align the discontinuities of the reconstruction, i.e., the polygon edges, with those of the continuous image.

Acknowledgements

The authors would like to thank Luiz Henrique de Figueiredo and Amitabh Varshney for useful discussions that resulted in this work. Also, thanks to the reviewers for the detailed comments. This research was done in the Visualization laboratory at SUNY at Stony Brook and in the VisGraf laboratory at IMPA, and was partially supported by CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico).

References

Barnsley, M. and Hurd, L. (1993). *Fractal Image Compression*. A K Peters.

Bergman, L., Fuchs, H., Grant, E., and Spach, S. (1986). Image rendering by adaptive refinement. In *Computer Graphics (SIGGRAPH '86 Proceedings)*, volume 20, pages 29–37.

Cendes, Z. and Wong, S. (1987). C^1 quadratic interpolation over arbitrary point sets. *IEEE CG&A*, 7(10):8–16.

de Figueiredo, L. H. (1995). Adaptive sampling of parametric curves. In Paeth, A., editor, *Graphics Gems V*, pages 173–178. AP Professional, San Diego, CA.

Floriani, L. D. (1989). A pyramidal data structure for triangle-based surface description. *IEEE CG&A*, pages 67–78.

Froment, J. and Mallat, S. (1992). Second generation compact image coding with wavelets. In Chui, C. K., editor, *Wavelets: A Tutorial in Theory and Applications*, pages 655–678. Academic Press, San Diego, CA.

Glassner, A. (1992). Anti-aliasing in triangular pixels. In Kirk, D., editor, *Graphics Gems III*, pages 369–373. AP Professional, San Diego, CA.

Gomes, J., Costa, B., Darsa, L., and Velho, L. (1996). Graphical objects. *The Visual Computer*, 12(6).

Gomes, J. and Velho, L. (1995). *Computação Gráfica: Imagem*. Serie Computação e Matemática, IMPA-SBM.

Guibas, L. and Stolfi, J. (1985). Primitives for the manipulation of general subdivisions and the computation of voronoi diagrams. *ACM Transactions on Graphics*, 4(2):74–123.

Haeberli, P. (1990). Paint by numbers: Abstract image representations. In *Computer Graphics (SIGGRAPH '90 Proceedings)*, volume 24, pages 207–214.

Heckbert, P. and Garland, M. (1994). Multiresolution modeling for fast rendering. In *Proceedings of Graphics Interface '94*, pages 43–50.

Lischinski, D. (1994). Incremental delaunay triangulation. In Heckbert, P. S., editor, *Graphics Gems IV*, pages 47–59. AP Professional, San Diego, CA.

Pennebaker, W. and Mitchell, J. (1993). *JPEG: Still Image Data Compression Standard*. The Colour Resource, San Francisco, CA.

Preparata, F. P. and Shamos, M. I. (1985). *Computational Geometry: an Introduction*. Springer-Verlag, New York, NY.

Radin, G. (1994). The pinwheel tilings of the plane. *Annals of Mathematics*, 139:661–702.

Rippa, S. (1992). Long and thin triangles can be good for linear interpolation. *SIAM J. Numer. Anal.*, 29(1):257–270.

Samet, H. (1984). The quadtree and related hierarchical data structures. *ACM Comp. Surv.*, 16:187–260.

Sibson, R. (1978). Locally equiangular triangulations. *The Computer Journal*, 21(3):243–245.

Velho, L. and Alvarenga, C. (1990). Image compression by first-order approximation. In *Proceedings IMAGE'COM*.

Williams, L. (1983). Pyramidal parametrics. In *Computer Graphics (SIGGRAPH '83 Proceedings)*, volume 17, pages 1–11.