Nonuniformly sampled computer-generated holograms

Ravikanth Pappu
Massachusetts Institute of Technology
Media Laboratory
Spatial Imaging Group
20 Ames Street
Cambridge, Massachusetts 02139
E-mail: pappu@media.mit.edu

Abstract. An novel algorithm for computing holographic fringe patterns based on nonuniform sampling is described. We show both analytically and via simulation that nonuniform sampling reduces the number of samples required to represent the hologram, in some cases by as much as 30 to 40%. This method of computing compact holograms is information-theoretically lossless. Methods for decompressing the computed fringe pattern are also presented. The algorithm is implemented on the Massachusetts Institute of Technology (MIT) Mark II Holographic Video System and results demonstrating its performance are provided. To the best of the author's knowledge, this is the first time that nonuniform sampling has been used to compress holographic fringe patterns for display purposes. The possibility of using other sophisticated nonuniform sampling approaches emerges from this research.

Subject terms: electronic holography; computer-generated holograms; nonuniform sampling; lossless compression.

Paper ELH-06 received Nov. 30, 1995; revised manuscript received Feb. 14, 1996; accepted for publication Feb. 15, 1996.

1 Introduction

The idea of computing holographic fringe patterns and using them in optical setups is not new—it is almost as old as off-axis holography. This paper is devoted to developing a novel method of computing holographic fringe patterns for use as the input to an electroholographic display. We use the Massachusetts Institute of Technology (MIT) Holographic Video System as an example in this paper. The basic optical layout is shown in Fig. 1. The system receives two inputs—laser light and a computer-generated hologram (CGH) and produces an image in a predefined volume. The mechanism of image formation is diffraction in an acousto-optic modulator (AOM). See Refs. 1 through 4 for an extensive treatment of the theoretical and engineering details of the system.

The technique for computing CGHs presented in this paper represents a departure from conventional methods of thinking about CGHs. The primary difference between all previous methods and the method presented here lies in the way the hologram is sampled. Our method relies on a nonuniform sampling technique to represent different regions of the hologram plane. A search of the literature has revealed no previous work in nonuniform sampling approaches of the type discussed in this paper as applied to computational holograms, although a paper by Vanderlugt[5] discusses a related technique of representing holographic fringe patterns.

The next section presents a review of previous work in computer-generated holography. The nonuniform sampling algorithm is developed in Sec. 3. A simple proof that the algorithm does indeed reduce the number of samples necessary is also presented in this section. A visual motivation for the algorithm is presented by means of scanning probe micrographs. Section 4 deals with issues involved in implementing the algorithm. Results are presented in Sec. 5, and Sec. 6 discusses avenues for future work. The final section summarizes the material presented in this paper.

2 Previous Work

A concise review of the important techniques used to generate holographic fringe patterns in the past is presented in this section, which serves two purposes. First, it demonstrates the historical progression of ideas related to representation of CGHs. Second, it shows that all but one of the methods rely on sampling the holographic fringe pattern uniformly. The one exception is Ref. 5, wherein Vanderlugt proves that the total number of samples required to represent a Fresnel (near-field) hologram is exactly the same as the number of samples needed for an equivalent Fourier (far-field) hologram provided the sample size is varied from place to place.

The problem of generating holographic fringe patterns can be stated very simply—how can the complex-valued fringe pattern be generated efficiently? To produce moving holographic images, the pattern has to be written onto an erasable and rewritable material that can be refreshed rapidly. All but one of the methods listed shortly rely on the physical modeling of interference of two waves—the object wave and the reference wave—to compute the hologram. A ''recording geometry'' is usually assumed and the complex amplitudes of the two waves are added in the hologram plane to produce the holographic fringe pattern. Then, a transparency must be produced with the holographic fringe pattern on it. The encoding of the fringe pattern on a transparency is also dependent on the output devices available to print the pattern.

An excellent historical review of the techniques used to compute holograms is presented in Ref. 6. We simply state

 0091-3286/96/$6.00

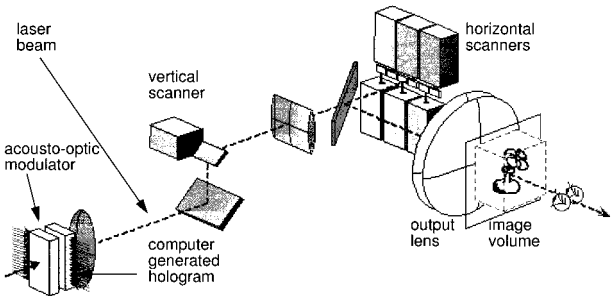


Fig. 1 MIT Holographic Video System Mark II—optical layout. Diagram courtesy of Pierre St. Hilaire.

the various methods used and review a few that have bearing on this work. The various methods, in roughly chronological order, are the binary detour-phase hologram,[7] generalized binary detour-phase hologram,[8] Lee's method (and Burckhardt's modification),[9,10] the kinoform,[11] and the referenceless on-axis complex hologram[12] (ROACH).

Several variations on the Fourier transform method of computing holograms have also been explored during the last two decades. Attempts have been made to compute the holographic fringe patterns of 3-D objects by slicing them in depth and computing and superposing the far-field hologram of each slice.[13] Iterative methods have been used to compute Fourier transform holograms.[14] The complexity of the objects used has kept pace with the increase in computing power available to the research community. However, Fourier transform holograms, by definition, are far-field holograms. To produce vivid 3-D images, near-field Fresnel holograms must be computed. We now turn our attention to efforts relating to the computation of near-field display holograms at the MIT Media Laboratory.

2.1 *Near-Field Holograms on a Mini-Supercomputer*

The first implementation of an algorithm to compute display holograms at the MIT Media Laboratory was performed on a Connection Machine-II mini-supercomputer.[15] The approach was fairly straightforward in that it modeled the propagation and interference of light and computed the fringe pattern according to the model. Horizontal-parallax only (HPO) holograms were computed, which meant that the hologram was composed of hololines—different horizontal lines of the CGH—each of which diffracted light only in the left-right direction. The holograms were displayed on the Mark I MIT Holographic Video System.

2.2 *Diffraction-Specific Computation*

This is an approach by Lucente[16] that does away with computing the interference pattern prescribed by the physical model of interference. Instead, an attempt is made to compute the diffracting structure that will cause a given image to form. Diffraction-specific fringe computation is essentially a recipe for deciding what to put into a computed fringe pattern so that it achieves a particular goal. This goal is usually defined in terms of bending light through a certain angle or focusing it to some area, in the simple case. The CGH is treated as an array of small pieces, called hogels, each of which is the superposition of a predetermined number of basis fringes. The contribution of each basis fringe in a hogel is specified by the corresponding element of a vector called a hogel vector. A vector-algebra representation of this process follows.

$$\underbrace{[w_1 \quad w_2 \quad \cdots \quad w_N]}_{\text{hogel vector}} \begin{bmatrix} \text{basis}_1 \\ \text{basis}_2 \\ \vdots \\ \text{basis}_N \end{bmatrix} = \underbrace{[\text{fringe}]}_{\text{hogel}}$$

The basis functions are calculated ahead of time by a process of simulated annealing. The elements of the hogel vector are determined using geometrical optics. The hogel can then be rapidly generated by performing a simple vector-matrix multiplication. This approach provides a significant improvement in the speed of computing because it does away with computing interference patterns. There is a one-time effort in computing basis fringes that are then stored in a lookup table. The physics of diffraction is embedded in the basis fringes while the amount of light energy that goes in each direction is determined by the elements of the hogel vector.

2.3 *Optimal Sampling of Fresnel Transforms*

Finally, we state the work that proposes a flavor of nonuniform sampling. In Ref. 5, Vanderlugt shows that it is possible to represent a Fresnel transform—in other words, a near-field hologram—with exactly the same number of samples required to represent a far-field hologram provided a specified nonuniform sampling method is used. His method involves the use of unequal sample sizes (and hence, unequal sample spacing) to represent Fresnel transforms. Vanderlugt's work was purely theoretical and was not implemented on any holographic display because there was only one real-time holographic display—the Mark I MIT Holographic Video System—in existence at the time of his work.

2.4 *Summary*

This review of the methods presented reveals a common feature: the hologram area is populated with samples—either real or complex-valued—which have a uniform spacing between them. In other words, they are all sampled at twice the maximum spatial frequency that occurs in the fringe pattern, following the prescription of the Shannon-Nyquist sampling theorem. This is entirely justified if and only if every point on the hologram records the maximum spatial frequency, say $f_{\max}$. It is the author's view that such a situation rarely, if ever, arises. There are several areas of the hologram that record maximum spatial frequencies that are significantly less than $f_{\max}$. In all the listed methods, these areas would still be sampled at twice $f_{\max}$, causing them to be oversampled. In this paper, we explore a technique of sampling the hologram area nonuniformly and present some of the trade-offs involved. In the next section, the proposed algorithm is developed and a proof that it does indeed reduce the total number of samples is presented.

3 Nonuniform Sampling Algorithm

In this section, we present the nonuniform sampling algorithm.[17] Consider a typical hologram recording geom-

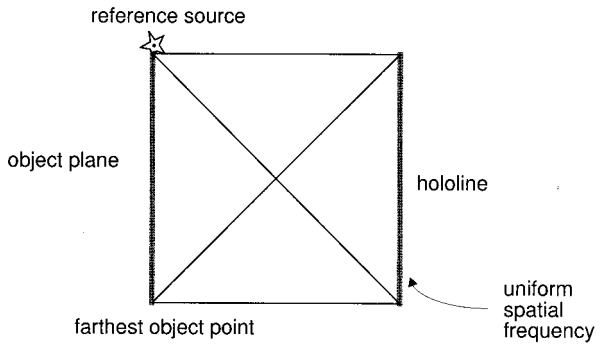


Fig. 2 Maximum spatial frequency is approximately constant across the hololine.

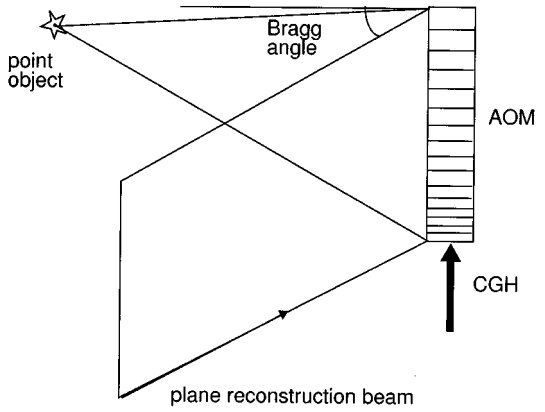


Fig. 3 In holovideo, the reconstruction beam is plane and the maximum spatial at all points on the hololine is not constant.

etry, as shown in Fig. 2. The reference beam source is located at the very edge of the object plane—a technique possible in computational holography because the intermodulation (or ''halo'') term can explicitly excluded from the computation. In the configuration shown, the maximum spatial frequency recorded on every point of the hologram is approximately the same. This is clear from the fact that the angle subtended by light from the farthest object point and reference source as measured at both ends of the hololine is the same, and this angle is approximately constant as the width of the plate is traversed. Constancy in spatial frequency is maintained under two conditions. First, the object must be one dimensional and light from every point on the object must be incident on the entire hololine. Second, the reference source must be a point source diverging from the edge of the object.

In the MIT Holographic Video System, neither of these conditions are met. The objects are always two dimensional and the reference beam has to be a plane wave because of the Bragg selectivity of the AOM. It is also not possible for the object points to illuminate the entire hololine because the AOM cannot record spatial frequencies beyond a certain cutoff frequency. Consequently, each object point can only illuminate a limited segment of the hololine. We conclude that the maximum spatial frequency recorded at all points on the hololine is not constant, but varies substantially from point to point. This is illustrated in Fig. 3. We now proceed to develop the nonuniform sampling algorithm.

What follows is applicable to one line of an HPO hologram where the object is three-dimensional and lies behind the hologram plane. The algorithm is

1. Segment the line into N equal parts.
2. Identify the maximum spatial frequency $f^i_{\max}$ in each part $(i=1,2,\ldots,N)$.
3. Compute a hologram sampled at $2\,f^i_{\max}$ in each segment.

The proof is as follows. Let the physical length of the hololine be P mm. The length of each segment is P/N mm. The total number of samples T_i in the ith segment is given by

$$T_i=(2Pf^i_{\max})/N \quad \text{samples.} \tag{1}$$

Because we determined the highest spatial frequencies in each segment, we can assume without loss of generality that the spatial frequencies can be sorted in decreasing order so that the following relationship is valid:

$$f^1_{\max}\geqslant f^2_{\max}\geqslant\cdots\geqslant f^N_{\max}. \tag{2}$$

The total number of samples $T_{\text{nonuniform}}$ in the hololine is given by

$$T_{\text{nonuniform}}=\sum_{i=1}^{N} T_i=\sum_{i=1}^{N}\left(\frac{2P}{N}\right)f^i_{\max}. \tag{3}$$

If the hologram had been sampled uniformly at $2\,f^1_{\max}$, the maximum spatial frequency on the hololine, then the total number of samples in the line would be

$$T_{\text{uniform}}=2Pf^1_{\max}=\sum_{i=1}^{N}\left(\frac{2P}{N}\right)f^1_{\max}. \tag{4}$$

Using Eq. (2), it is clear that

$$T_{\text{nonuniform}}\leqslant T_{\text{uniform}}. \tag{5}$$

This result demonstrates that the total number of samples required to represent the nonuniformly sampled hologram is always less than or equal to the total number of samples required to represent an equivalent uniformly sampled hologram.

3.1 *Physical Justification*

In the previous subsection, we showed that the number of samples in a nonuniformly sampled CGH is smaller than the number of samples in a uniformly sampled CGH for the same object. The proof was based on the assumption that the maximum spatial frequency recorded is not constant across the length of the hololine. To see that this is indeed often the case, the fringes on an optically recorded HPO hologram were observed using a scanning probe microscope. The hologram used was a 4-×5-in. bleached rainbow hologram.

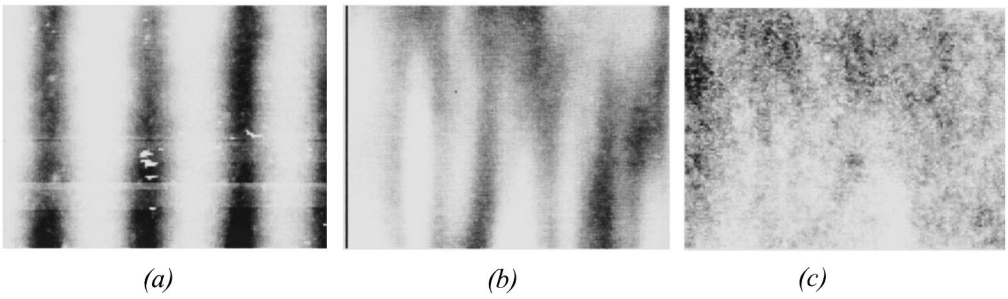


Fig. 4 (a) Low-spatial-frequency portion of the hologram, (b) the medium-spatial-frequency portion, and (c) the highest spatial frequency portion. These three scans came from different locations on a single horizontal line in an HPO hologram. The horizontal extent of all these scans is about 7 μm.

Bleaching was necessary to convert intensity variation into a surface relief pattern. Three 7-×7-μm scans were taken at three different locations along a horizontal line. These three scans are reproduced in Fig. 4(a), 4(b), and 4(c). From the scans, it is clear that the maximum spatial frequency at the three different locations on the hololine is not constant. Rather, it varies quite significantly from one location to another. These results justify the use of a nonuniform sampling strategy.

4 Issues in Nonuniform Sampling

In this section, we present and discuss the most important issues that arise when nonuniform sampling is used to compute holograms for display applications. Each of the issues is enumerated.

4.1 *Optimal Number of Segments in a Hololine*

A look at the algorithm presented in the previous section indicates that a decision has to be made regarding the number of segments in any given hololine. Recalling Eq. (3), we observe that the total number of samples apparently decreases with increasing N:

$$T_{\text{nonuniform}} = \sum_{i=1}^{N} T_i = \sum_{i=1}^{N} \left(\frac{2P}{N} \right) f^i_{\max} .$$

What, then, prevents us from making the number of segments very large? There are three reasons for keeping N reasonably small. First, the maximum spatial frequency $f^i_{\max}$ in each segment depends on the size of the segment and the locations of its endpoints. Therefore, $f^i_{\max}$ depends on N in a nontrivial way. The relationship between the number of segments and the total number of samples in a hololine is examined graphically in Fig. 5. Second, it must be kept in mind that the AOM requires that all the incoming data be sampled at a uniform rate. This is because the velocity of the acoustic wave in the AOM is constant regardless of sampling rate. To bring a low-rate segment back to a constant high rate, some form of interpolative filtering is necessary. A larger number of segments implies more time consumed in filtering. Third, the law of diminishing returns takes effect as the number of segments is made larger. The savings in the number of samples diminishes and reaches a steady value when the number of segments crosses a certain threshold. Any number of segments above this threshold causes more trouble than good. This is also illustrated in the following example.

Consider an object that is composed of a small number of points that are distributed randomly in the x-z plane. Figure 5 is a plot of the total number of samples as the number of segments is varied from 1 to 180. The graph clearly indicates the drop in the number of samples as the number of segments is increased. The number of samples drops by about 30% from 230,000 samples per hololine to 160,000. This result suggests that the reduction in the number of samples is significant when the object points are distributed nonuniformly throughout the x-z plane. Most objects of interest possess such a point distribution. Further, the total number of samples levels off when the number of segments becomes very large. Therefore, it does not make sense to choose a very large number of segments because the gains obtained by doing this diminishes as the number of segments increases.

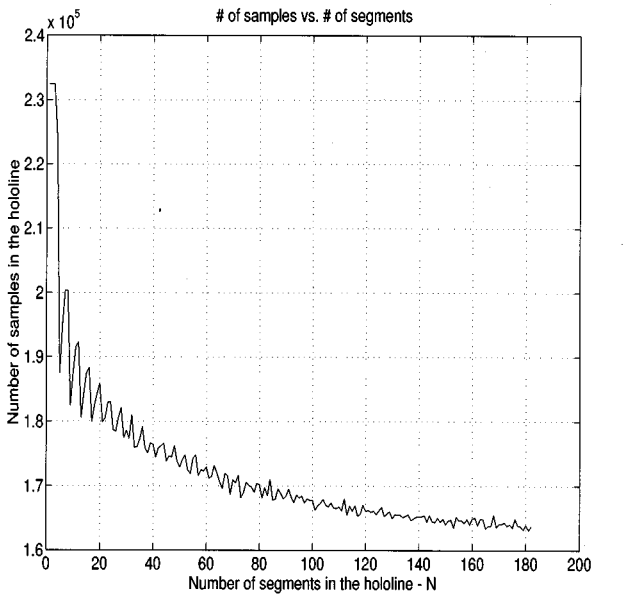


Fig. 5 Variation in the number of samples in a hololine with the number of segments. The total number of samples reaches a steady state.

4.2 *Interpolation*

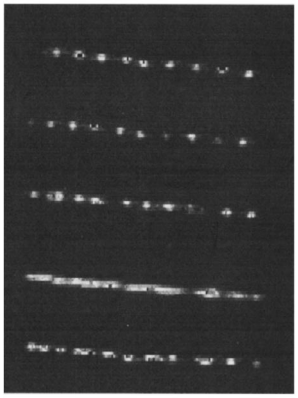
The output of the algorithm is a hololine that has N segments, each possibly sampled at a different rate. This hololine is fed into an AOM, which requires that its input signal be sampled at a single uniform rate. This is because the acoustic wave propagates down the crystal at a uniform velocity of 617 m/s regardless of the rate at which the signal was sampled. It is, therefore, necessary to convert the entire hololine to a uniform rate before sending it to the AOM. Conversion to a uniform rate can be achieved by interpolation, of which there are several kinds. Interpolation is simply the creation of some number of samples ''in-between'' the samples that one already has. No new information is created. Some of the kinds of interpolation are low-pass interpolation, linear interpolation, and replication. In low-pass and linear interpolation, the data to be interpolated are filtered by a specially designed digital filter to produce interpolated output.[18] Replication is a much simpler form of this where the number of extra samples required is produced by replicating existing sample values. For example, if a segment has 657 samples and it is required to have 1024 samples, then 367 samples randomly chosen in the data and replicated to produce a segment that contains 1024 samples. Replication is much faster than filtering if the operations are performed in software, but it also causes the shifting of samples from their original location—an artifact that does not accompany any other method of interpolation. Another method of interpolation is a technique borrowed from computer graphics known as the digital differential analyzer. This algorithm uses a simple incremental algorithm to rapidly compute intermediate values given the endpoints of a line.[19]

4.3 *Object Considerations*

The number of points in the object, their density in the x-z plane, and their depth from the hologram plane are all important factors in determining the total number of samples in any hololine. In this section, we look at a few limiting cases to get a feel for the behavior of the nonuniform sampling algorithm.

The worst case is when the object is a single line of points at the same depth with a very high density (number of points per unit length). Every segment will then receive a maximum spatial frequency contribution from some point in the line, which means that every segment will have to be sampled at a high rate. It makes sense in such cases to choose $N=1$ so as to simplify the computation. It is indeed fortunate that not many of the objects of usual interest possess such a structure. On the other hand, objects with random point locations and depths, such as the one that generated Fig. 5, make for extremely varied sampling rates leading to compact holograms. In this case, N should be chosen as large as possible to ensure greater savings. Of course, such objects also are not of much interest for display purposes either. Most of the objects that we will be interested in tread the middle ground between the two cases mentioned. Therefore, we expect the savings in the number of samples to be somewhere between the two cases.

To summarize, the total number of samples decreases but the effort to interpolate increases as the number of segments increases. The structure of the object also determines what kind of sampling rates are necessary in each segment so it is important to look at the object and make a decision on the number of segments. It is only necessary to look at the gross structure of the object because the spatial frequency structure of the hologram depends solely on geometrical considerations. The exact structure is not important in this calculation. Once the sampling rates are determined, they can be stored for use again. In this sense, it is a one-time effort.



Segments	Samples
1	262144
64	229774
128	223280
256	218288
512	215736

Fig. 6 Images of a single line of points produced by the algorithm. The first line consists of a single segment and is equivalent to a fully computed hologram. All subsequent lines are produced with varying number of segments (and hence samples) as indicated in the figure. All of these lines were produced without any interpolation, demonstrating the high tolerance of the AOM for varying sampling rates.

5 Results

The nonuniform sampling algorithm was implemented on an SGI Onyx workstation. Figure 6 shows the images of a single line of points that are spaced 1 mm apart in the x direction and 0.5 mm apart in the z direction. Because the points are uniformly spaced and span the entire object space, this is almost a worst-case scenario as far as nonuniform sampling is concerned. There is always at least one object point that contributes the maximum possible spatial frequency in a particular segment. The hologram corresponding to the topmost image had a single segment and is equivalent to a fully computed hologram. The subsequent images had the number of segments indicated in the columns alongside the figure. Two points must be noted. First, no interpolation was performed on the computed hologram. The hologram was displayed as computed, after being padded with zeros to make it appear like a 256-kbyte hololine (as required[20] by the framebuffer used in the Mark II). In Fig. 6, the fully computed hologram has 262,144 samples and the hologram with 512 nonuniformly sampled segments has 215,736 samples—a reduction of about 18%. This is less than the 30% obtained for the random object of Fig. 6, but appreciable nevertheless. Note also that the results look significantly better on the MIT Holographic Video System than they do on paper. The high tolerance of the AOM to changes in sampling rate is also evident in Fig. 6. This feature indicates the possibility of doing less interpolative filtering than usually required. For example, if two consecutive segments have maximum spatial frequencies that differ by no more than 10%, it is possible to interpolate both of them with the same filter.

6 Future Directions

In this section, we present two avenues for further study in the area of nonuniformly sampled CGHs.

6.1 *Nonuniform Segment Widths*

The algorithm described before used a certain number of segments all of which were the same physical width, but sampled at different rates. There is no reason why all the segments should be the same width. They could easily have been of different widths without increasing the complexity of computation or interpolation. To illustrate the implications of variable segment widths, let us reconsider Eq. (3). We replace the segment width P/N with w_i, the width of an individual segment that may or may not be the same as w_j, for $i \neq j$. We then have

$$T_{\text{nonuniform}} = \sum_{i=1}^{N} T_i = \sum_{i=1}^{N} 2 w_i f_{\max}^i, \tag{6}$$

with the constraint that

$$\sum_{i=1}^{N} w_i = P,$$

where P is the physical width of the hologram. This is an optimization problem. One possible approach is to decide the segment width adaptively based on the local spatial frequency gradient.

6.2 *Bandpass Sampling of CGHs*

One assumption that was made in the formulation of the nonuniform sampling algorithm was that every segment has spatial frequency contributions all the way from zero to $f_{\max}^i$. Simulation studies have shown that this is not always true and that there are several segments where the spatial frequency range is quite small. Such a segment can be called a bandpass segment. There are well-known techniques in communication theory that allow for sampling bandpass functions at a smaller rate of twice the actual bandwidth instead of at twice the maximum spatial frequency. One possible implementation of this technique is called heterodyned sampling. The bandpass signal is shifted in the frequency domain by multiplying by a sinusoid and then low-pass filtering to retain only the baseband. The baseband signal is then sampled at twice the bandwidth, as prescribed by Nyquist. Some postprocessing is required to recover the actual samples from the baseband samples. The postprocessing involves representing the signal in the conventional analytic form to recover the bandpass signal, a well-documented technique.[21,22] The big gain lies in the significantly lower bandwidth communication between the host computer and the framebuffer made possible by using bandpass sampling.

How is this technique of advantage to computational holographers? Recall the arguments made in Sec. 2. One of the quantities that required minimization was the total number of samples in the CGH. Both nonuniform sampling and bandpass sampling achieve this goal—bandpass sampling is more effective. The number of samples that are sent to the framebuffer are reduced by using either method. If the framebuffer had no computational capability, then the entire exercise would be futile because there would be no means of recovering a uniformly sampled stream of CGH data before it was input into the AOM. However, with an intelligent framebuffer such as the MIT Cheops system, it is possible to perform a great deal of signal processing on the data through the use of one or more specialized ''daughter cards''—cards that are designed to implement a specific type of operation in hardware. Both nonuniform and bandpass sampling require the use of such hardware, e.g., the filter processor and the remap processor.[23] Note once again that any advantages in computation and communication are independent of encoding methods that are used downstream in the communication chain.

Let us now consider how a nonuniform sampling scheme could be implemented if Cheops did not exist or, for some reason, was not available to us. The options available would be to use the radio-frequency (rf) portion of the holographic display to do some signal processing or to use optical means to decompress the data. An rf solution would involve changing the carrier frequency dynamically to match the sampling rate of each segment. This solution can be practically implemented. An optical solution would almost certainly involve more moving parts or a pulsed laser or both. Since one of the ultimate goals is to eliminate all moving parts from the display and make it inexpensive, both optical options do not appear to be very practical at the present time.

7 Summary

We showed that a nonuniform sampling algorithm can be used to reduce the number of samples in a CGH by 30 to 40% depending on the object under consideration. This is the first time, to the best of the author's knowledge, that nonuniformly sampled computed holograms have been used for computing display holograms. The issues relating to the nonuniform sampling algorithm were examined and methods for decompressing the CGH were provided. The principal advantages of the nonuniform sampling algorithm are, first, the savings that accrue due to the nonuniform sampling algorithm are independent of any further savings that may be possible due to encoding schemes and, second, the nonuniform sampling algorithm is lossless from an information theory standpoint.

It is clear that a great deal of work is still possible in simply reducing the number of samples in CGHs. This paper has explored the simplest possible nonuniform sampling method and its implications for computing display holograms. The possibility of using other sophisticated nonuniform sampling techniques emerges from this research.

Acknowledgments

The author thanks Professor Stephen A. Benton, Professor V. Michael Bove Jr., and the members of the Spatial Imaging Group at the MIT Media Laboratory for constructive criticism and helpful discussions during this research. This research was supported by the Television of Tomorrow Research Consortium at the MIT Media Laboratory.

References

1. J. S. Kollin, S. A. Benton, and M. L. Jepsen, ''Real-time display of 3-D computed holograms by scanning the image of an acousto-optic modulator, in *Holographic Optics II: Principles and Applications*, G. M. Morris, Ed., *Proc. SPIE* **1136**, 178–185 (1989).
2. P. St.-Hilaire, ''Real time holographic display: improvements using higher bandwidth electronics and a novel optical configuration,'' SM Thesis, Media Arts and Sciences Section, Massachusetts Institute of Technology (1990).
3. P. St.-Hilaire, S. A. Benton, M. Lucente, J. D. Sutter, and W. J. Plesniak, ''Advances in Holographic Video,'' in *Practical Holography VII: Imaging and Materials*, S. A. Benton, Ed., *Proc. SPIE* **1914**, 188–196 (1993).
4. P. St.-Hilaire, ''Scalable optical architectures for electronic holography,'' PhD Thesis, MIT Program in Media Arts and Sciences, Massachusetts Institute of Technology (1994).
5. A. Vanderlugt, ''Optimum sampling of Fresnel transforms,'' *Appl. Opt.* **29**(3), 3352–3361 (1990).
6. G. Tricoles ''Computer generated holograms: an historical review,'' *Appl. Opt.* **26**, 4351–4360 (1987).
7. B. R. Brown and A. W. Lohmann, ''Computer generated binary holograms,'' *IBM J. Res. Develop.* **13**, 160–167 (1969).
8. R. E. Haskell and B. C. Culver, ''New coding technique for computer-generated holograms,'' *Appl. Opt.* **11**, 2712–2714 (1972).
9. W. H. Lee, ''Sampled Fourier transform hologram generated by computer,'' *Appl. Opt.* **9**, 639–643 (1970).
10. C. B. Burckhardt, ''A simplification of Lee's method of generating holograms by computer,'' *Appl. Opt.* **9**, 1949 (1970).
11. L. B. Lesem, P. M. Hirsch, and J. A. Jordan, Jr., ''The kinoform: a new wavefront reconstruction device,'' *IBM J. Res. Develop.* **13**, 150–155 (1969).
12. D. C. Chu, J. R. Fienup, and J. W. Goodman, ''Multi-emulsion, on-axis, computer generated hologram,'' *Appl. Opt.* **12**, 1386–1388 (1973).
13. J. P. Waters, ''Three-dimensional Fourier transform method for synthesizing binary holograms,'' *J. Opt. Soc. Am.* **58**, 1284–1288 (1968).
14. J. R. Fienup, ''Iterative method applied to image reconstruction and to computer-generated holograms,'' *Opt. Eng.* **19**(3), 297–305 (1980).
15. J. S. Underkoffler, ''Development of parallel processing algorithms for real-time computed holography,'' SB Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology (1988).
16. M. Lucente, ''Diffraction-specific fringe computation for Electroholography,'' PhD Thesis, Electrical Engineering and Computer Science Department, Massachusetts Institute of Technology (1994).
17. R. S. Pappu, ''Minimum information holograms,'' SM Thesis, Massachusetts Institute of Technology (Sep. 1995). (This thesis is on the World Wide Web at http://pappu.www.media.mit.edu/people/pappu/)
18. A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ (1989).
19. J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice*, Addison-Wesley, Reading, MA (1990).
20. V. M. Bove, Jr., and J. Watlington, ''Cheops: a data-flow processor for real-time video processing,'' MIT Media Laboratory Technical Memo (1993).
21. S. Haykin, *Communication Systems*, Wiley Eastern Limited, New Delhi (1989).
22. R. J. Marks II, *Introduction to Shannon Sampling and Interpolation Theory*, Springer-Verlag, New York (1991).
23. V. M. Bove, Jr., and J. A. Watlington, ''Cheops: a reconfigurable data-flow system for video processing,'' *IEEE Trans. Circ. Syst. Video Technol.* **5**(2), 140–149 (1995).

Ravikanth Pappu is a PhD candidate in the Program in Media Arts and Sciences and a research assistant in the Spatial Imaging Group at the MIT Media Laboratory. His research interests are in computational holography, signal processing, and information theory. He is currently working on algorithms for efficient computation of computer-generated holograms. Pappu holds a BE in electronics and communication engineering from Osmania University, Hyderabad, India. He received an MSEE from Villanova University in 1993 and a MS in media arts and sciences from MIT in 1995. He is a member of Sigma Xi.