Chapter 5

Hogel-Vector Encoding

Diffraction-specific fringe computation provides a foundation for the development of two holographic encoding schemes. In this chapter and the next, hogel-vector encoding and "fringelet encoding" are described and demonstrated. Both are capable of achieving compression ratios of 16 and more, and both dramatically reduce the total time required to generate holographic fringes. These encoding schemes are the fulfillment of the most elusive goal of diffraction-specific computation.

Diffraction-specific fringe computation is developed around the ideas of sampling. A holographic pattern is represented by a sampling of its spectrum as a function of both space and spatial frequency. The information content of such a sampled fringe pattern is equal to the product of the number of hogels times the number of components in each hogel vector. Therefore, the sample spacing in the two sampled dimensions (w_h in space and Δ_f in spatial frequency) determine the information content of the sampled fringe representation. To reduce the number of samples, the spectrum of each hogel is sampled in larger frequency steps. This reduction in the total number of samples comprising the hogel-vector array provides a savings of bandwidth at the stages of computation before the hogel-vector array is converted to the final fringe pattern. The final fringe pattern must have a minimum number of samples, as dictated by the physics of diffraction. However, if the hogel-vector description of the fringe pattern contains fewer samples, then the array of hogel vectors can be thought of as an encoded form of the fringe pattern, one that is compressed in terms of information bandwidth.

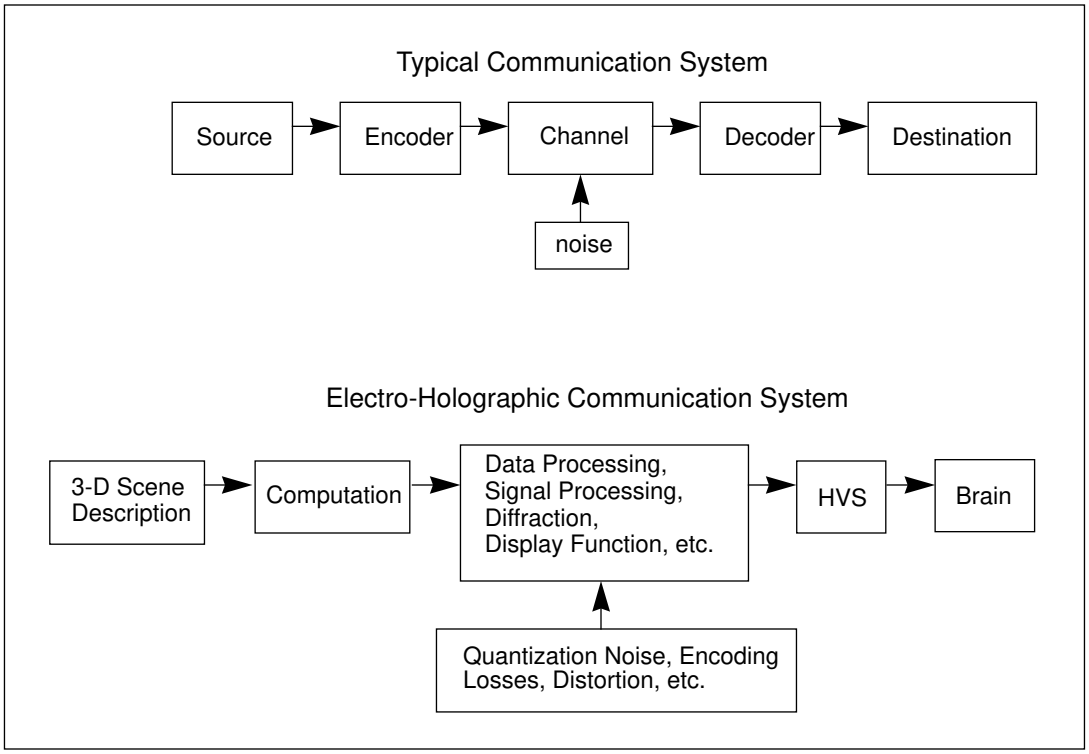
It is important to compress the huge information content of a fringe pattern so that it can be easily displayed, transmitted, or stored. In particular, because the Cheops framebuffer system (used to drive the second generation MIT holovideo display) is designed to read in data over a relatively low-bandwidth SCSI link, compression is

central to avoiding this bottleneck. To understand the nature of a holographic encoding scheme, fringe computation is next discussed in terms of communication systems. Such a discussion helps to frame the issues of information content in the realm of electro-holographic imaging.

5.1 The Electro-Holographic Communication System

This section deals with the concepts of information coding in a communication system. The communication system to be discussed - the electro-holographic display of 3-D images - is one that converts 3-D digital data (the 3-D scene description) into holographic images. The information is converted from one format to another along the way: hogel vectors, hogels, diffracted light, etc. The amount of information used to represent each of these formats (the required channel capacity or bandwidth) changes, though the amount of useful information never exceeds the amount contained in the original 3-D data source. One of the primary purposes for the development of diffraction-specific computation is to minimize the bandwidth required to represent the holographic information. Only for the purpose of actually diffracting light must the encoded fringe be converted to the less efficient, higher bandwidth fringe.

A typical communication system is diagramed in the succeeding figure. Data begins at the source, and the goal is to reproduce the data at the destination as accurately as possible[71]. Noise is introduced in the channel. This model, adapted to apply to holovideo computation and display, becomes a holographic communication system.



In the general case of holovideo, the source is a description of the 3-D object scene to be displayed. The encoder converts this information into fringes. The display system acts as the channel, converting the fringe information into a holographic image. The channel includes the intermediate storage of the fringes in a framebuffer, and any signal processing that occurs in the analog domain on the way to the display. The channel must also include the propagation of the diffracted light to the viewing zone. The decoder is essentially the stimulation of the human visual system (HVS). The destination is the brain of the viewer. When discussing holographic encoding schemes, the encoder is the computational process that generates the encoded fringe representation, and the channel includes the decoding process, which often contributes additional noise to the overall system. An effective encoding scheme reduces the required bandwidth (capacity) of the channel. Bandwidth compression allows for more rapid trans-

mission (display) of holographic information, and allows for faster computation at the source.

5.1.1 Information Symbols

In an information content analysis, it is necessary to define a set of information-bearing symbols. Discrete samples of continuous signals are commonly used as the information-bearing symbols[71]. The final fringe pattern that is used to diffract light in the display system is measured in samples. In the current MIT holovideo display, the number of 8-bit samples is 36 M per fringe pattern. It is possible to compute a large number of different 36-MB fringe patterns to generate the same holographic image. For example, because the acuity of the HVS is limited, every point or element in an image can be moved by an imperceptible amount before computing the fringe pattern using traditional interference-based methods. This can result in thousands of different fringe patterns that produce essentially the same image. Clearly, the 36 Msamples are not being put to full use in an information bearing sense: these samples are dedicated to producing microscopic imperceptible differences in images. If an encoded format is to represent this fringe pattern with a reduced number of symbols, then most of these 36 M samples must be culled from the fringe. This observation, however, gives no clue as to how bandwidth reduction can be performed. This is one of the reasons that until now no research has been done to develop a fringe description that allows for the unique and therefore most information-efficient specification system.

Diffraction-specific computation uses a sampled spectrum to encode holographic information. The encoding formats developed from diffraction-specific computation are formulated by specifying the diffractive duty of each hogel only to a degree that matches the requirements of a typical human visual system.

5.1.2 Information Entropy

The concept of *entropy* is used as a measure of efficiency (versus redundancy) in an encoding scheme. The entropy of the occurrence of a symbol is essentially a measure

of its uncertainty and is a function of its probability p of occurring[71]. Simply stated, the entropy of a symbol is the useful information conveyed by the occurrence of a particular symbol. Each symbol has an entropy of

$$\text{entropy} = -p\log_2 p. \tag{14}$$

The higher the entropy, the more information "valuable" a symbol is. (For practical systems, $0 \leq p \leq 0.5$.) An important axiom in information theory is that when multiple symbols (different fringes) exist containing the same information (the same hogel vector component), the entropy in a symbol transmission underutilizes the channel capacity[71]. To most efficiently use bandwidth (the number of samples), each symbol must convey unique information.

Consider hogel vectors. There are a large number of fringe patterns that diffract light from a particular hogel to a particular part of the imaged scene. The actual usable information in any one of these possible fringes is the same, yet only a single hogel vector component is required to convey the information that light is to diffract is this manner. The components of hogel vectors are a more efficient set of coding symbol, and have an entropy that is roughly ten times higher than the individual samples in a traditionally computed fringe pattern.

Recall that hogel vectors, in representing the diffractive duties of a hogel, are related directly to the spectrum of a hogel. Consider the number of different spatial frequencies in a hogel. For N_h samples, this number is N_h. Although the spectrum has N_h samples for the magnitude and N_h for the phase, the real-valued fringe (physically, a real-valued intensity) has a spectrum with even conjugate symmetry (i.e., S(f)=S*(-f)). Therefore, the N_h samples describing spectral phase carry no useful information from the point of view of the holographic system. The useful information content of a set of spatial frequencies depends on the number of spatial frequencies that give rise to distinguishably different viewer stimuli. For example, a hogel containing N_h=1K discrete spatial frequencies can diffract light in 1K different directions. However, the viewer is capable of resolving only a fraction of these. In the development of the diffraction-spe-

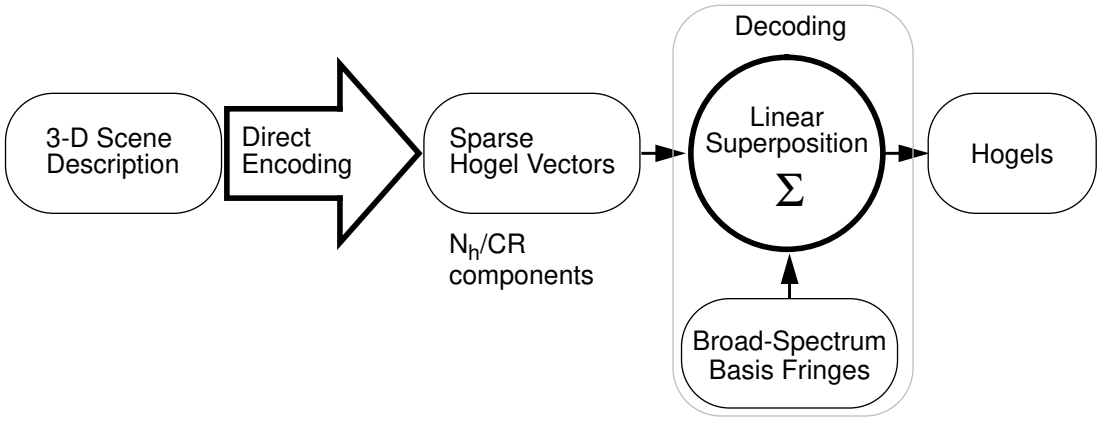
cific method of computation (see Section Chapter 4) it is remarked that the compression ratio achievable in HPO holographic fringes should be ten or more without destroying image quality. Therefore, to reduce bandwidth, efficient holographic encoding schemes must discretize the spectrum in larger steps.

The focus of the remainder of this dissertation is on the two encoding schemes developed on top of diffraction-specific computation. The first - called "hogel-vector encoding" - is based on undersampling the spectrum of hogels as represented by hogel vectors. The second - called "fringelet encoding" - is also based on undersampling the spectrum, but uses an encoding format that is a step closer to an actual fringe representation but using only a fraction of the number of samples.

5.2 Description of Hogel-Vector Encoding

The first type of holographic encoding is called "hogel-vector encoding" because it essentially undersamples hogel vectors to reduce information content. The array of hogel vectors is treated as the encoded fringe format. To achieve bandwidth compression, the hogel spectrum is discretized in larger steps. For larger spectral sampling, fewer samples are required, increasing the compression ratio (CR). The decoding step in hogel-vector decompression is the process of superposing basis fringes weighted by the appropriate hogel-vector components to produce a decoded hogel (fringe). But because hogel-vector encoding undersamples the spectrum, it must use a special set of basis fringes. For a higher CR each basis fringe must represent a larger portion of the hogel spectrum. Each basis fringe must be tailored to cover a proportionally larger region of the spectrum. The decoded hogels each contain information about the entire spectrum. The difference between these hogels and the ones generated without compression (i.e., minimum spectral sampling) lies in the specificity and uncertainty in the final spectrum.

Hogel-Vector Encoding and Decoding



In hogel-vector encoding, a more coarsely sampled spectrum represented by a hogel vector with fewer discrete components contains a less specific description of the desired diffraction. Since a single component of a hogel vector includes all contributions within its enlarged region of the spectrum, information is lost: it is uncertain from whence in the spectral range a unit of energy in a particular vector component arose. However, from the point of view of the human visual system, the fully sampled hogel-vector array contains redundant information about the hogel spectra. Therefore, hogel-vector encoding eliminates redundancy in holographic fringes and injects a tolerable amount of ambiguity.

Stated in terms of the holographic communication system, the entropy contained in each information-bearing symbol (hogel-vector component) is increased. For example, a hogel vector that is not undersampled contains N_h components. Generally, each is equally like to occur, making each symbol have a probability

$$p = \frac{1}{N_h} \tag{15}$$

of occurring and an entropy of

$$\text{entropy} = -\frac{1}{N_h}\log_2(\frac{1}{N_h}) \quad (16)$$

For N_h=512, the entropy per symbol is (1/512)x9 = 0.0176. If the spectrum is undersampled by a factor of CR=8, each symbol has a probability and entropy of

$$p = \frac{CR}{N_h} \quad (17)$$

$$\text{entropy} = -\frac{CR}{N_h}\log_2(\frac{CR}{N_h}). \quad (18)$$

For N_h=512, the entropy per symbol in this compressed case is (8/512)x6 = 0.0938. The channel of the holographic communication system is used over 5 times more efficiently in terms of usable information conveyed per symbol.

Hogel-vector encoding is somewhat similar to certain 2-D image encoding schemes that divide the 2-D image into blocks. In particular, hogel-vector encoding is similar to discrete cosine transform (DCT) encoding[67]. The important difference is that hogel-vector decoding does not use constant frequency sinusoids as basis functions. The basis fringes used to perform the hogel-vector decoding must utilize the whole spectrum available to the hogel. If the decoded fringe has gaps in its spectrum (as in the following illustration) the dropouts severely degrade the image, producing a picket fence of drop-outs across the viewing zone, giving the image the appearance of being behind bars. Gaps in the spectrum do not fulfill the second step in sampling theory, which requires that an appropriate low-passing be performed on the samples to recover the continuous signal. In hogel-vector encoding, the decoding step performs the proper low-pass filtering of the sampled spectrum by convolving each component with the spectrum (rectangular for purposes of illustration) of the corresponding basis fringe. A properly decoded fringe has a continuous spectrum (as in the following illustration), and the desired spectrum is reproduced with only the added ambiguity as a possible artifact.

.

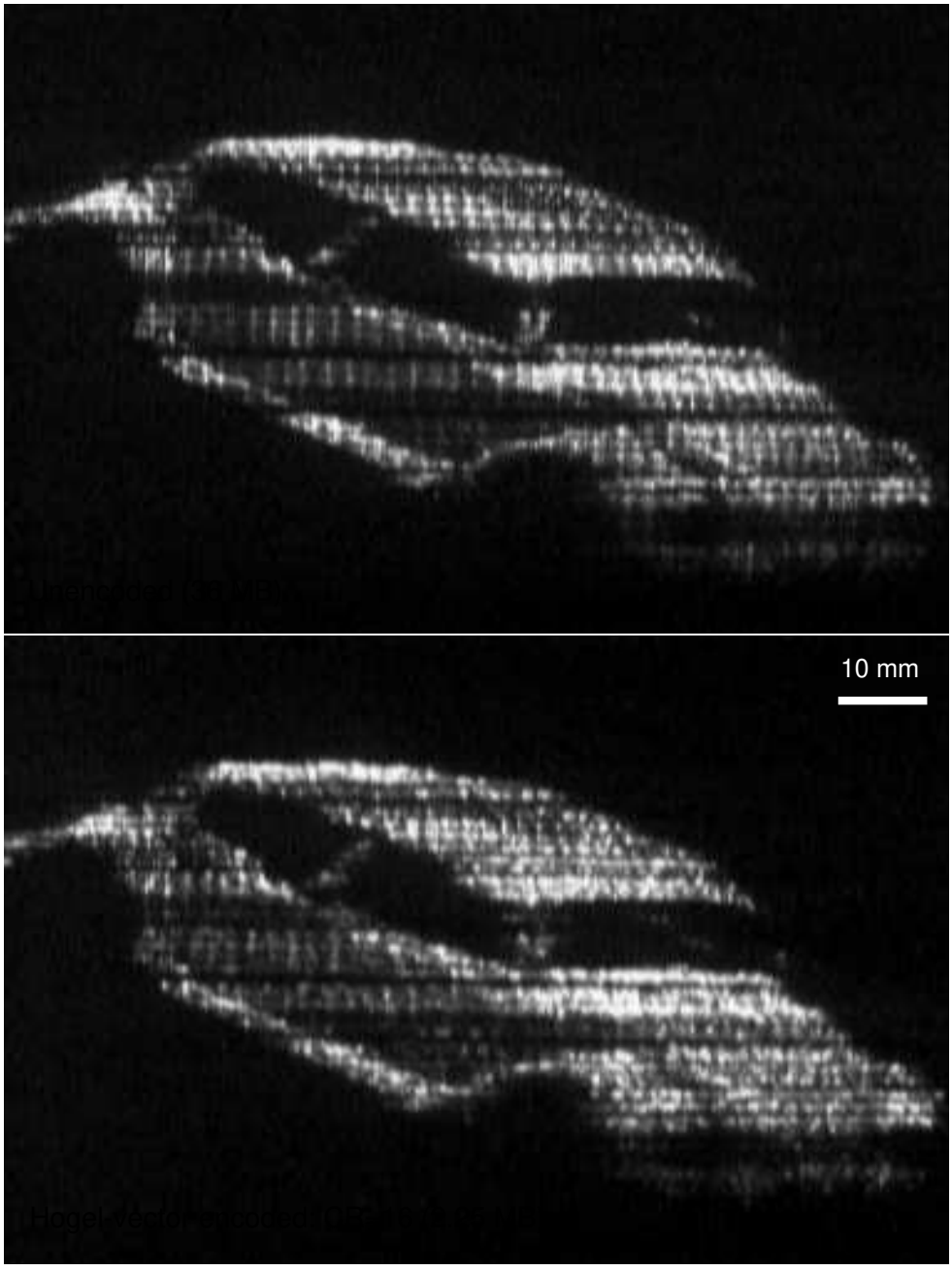
5.3 Image Generation

The computation steps in hogel-vector encoding are the same as the hogel-vector generation described for (unencoded) diffraction-specific computation. The two-step computation was implemented on the Onyx/Cheops computation platform. The hogel-vector array was generated on the Onyx workstation. The hogel-vector array was downloaded to the Cheops P2 card, where it was then decoded using the Splotch Engine. The decoded 36-MB fringe pattern was subsequently loaded into the VRAM of the Cheops output cards. (Due to concern for reliability, for some experiments the hogel-vector decoding was performed on the Onyx, and the decoded fringes were downloaded directly to the Cheops output cards. The computed fringes were identical, byte-for-byte, to those computed using the Splotch engine.) For a CR of 1 (no encoding), the hogel-vector array comprises 36 MB. For larger values of compression ratio, this number is proportionally smaller: a CR=16 gives a 2.25-MB hogel-vector array.

Images generated using hogel-vector encoding showed an increase in imaged point spread as a function of compression ratio (CR). In many cases, this point spread was not perceivable. Hogel-vector encoding also added a noticeable speckle-like appearance to the image. The figure on page 86 shows results using hogel-vector encoding. The 3-D holographic image of a VW Beetle* car was generated from a polygon data-

*. Historical note: This VW Beetle database was originally measured by hand by Ivan Sutherland et al.[11]

Image of a VW Beetle



Top: A digitally photographed picture of a 3-D image computed using diffraction-specific computation.

Bottom: The same image, computed using hogel-vector encoding, CR=16, N_h=1024. The discrete image points that form the image are blurred, and a slightly speckle-like appearance is added to the image.

base comprising 1079 polygons. The image was converted into over 10,000 discrete elements in the image volume using a simple lighting model.

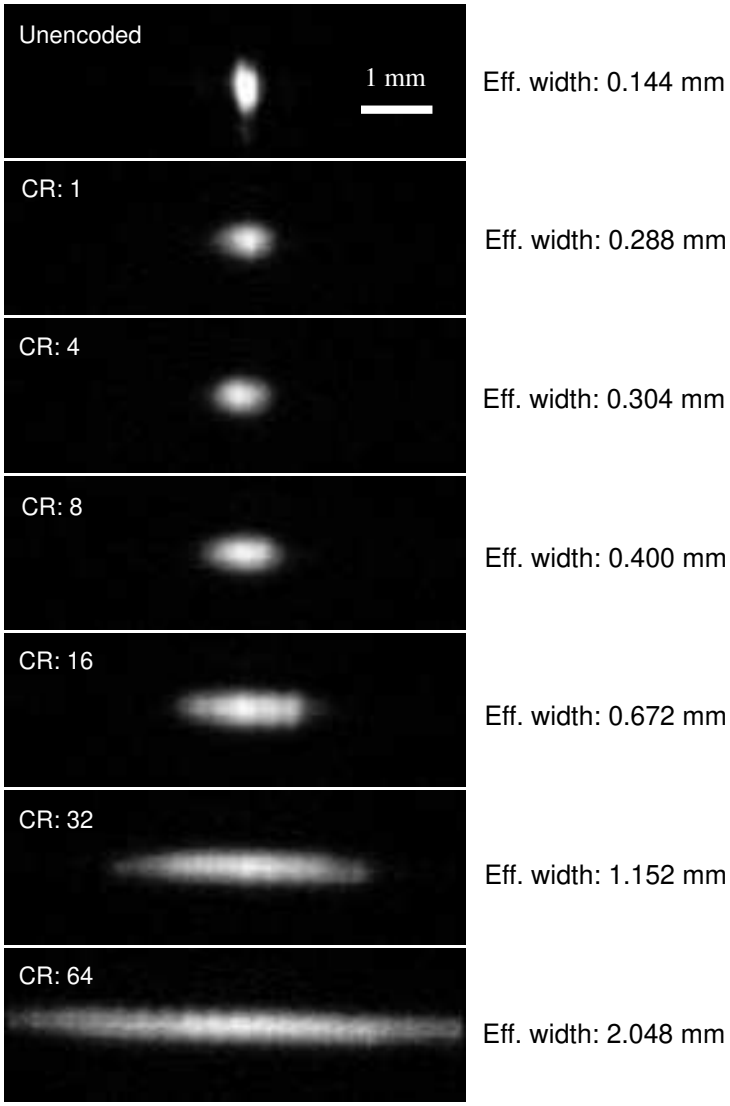
To analyze the effect of hogel-vector encoding on point spread, a series of experiments were performed using a single imaged point. To make the blurring effect as pronounced as possible, the point was imaged to z=80 mm, the maximum usable depth of the MIT display. The point was also imaged to other depths, such as z=40 mm, z=20 mm, etc. In each case, the imaged point was generated with a fringe pattern (a single hololine) computed using hogel-vector encoding. Compression ratios ranged from CR=1 to CR=64, in powers of two. Typical hogel widths used and their corresponding number of samples per hogel, are listed in the following table. (Fringe sampling pitch was 1.7×10^3 samples/mm.)

w_h	N_h
0.150 mm	256
0.300 mm	512
0.600 mm	1024
1.200 mm	2048

To capture the imaged spot's cross-section, the small 768x494 CCD array was placed in the beam path within the image volume at the location of an imaged point. The detected cross-section was digitized and stored. A measure of effective point spread was calculated for each using the half-energy convention. Many of these cross-sections and point-spread data are gathered on the succeeding pages.

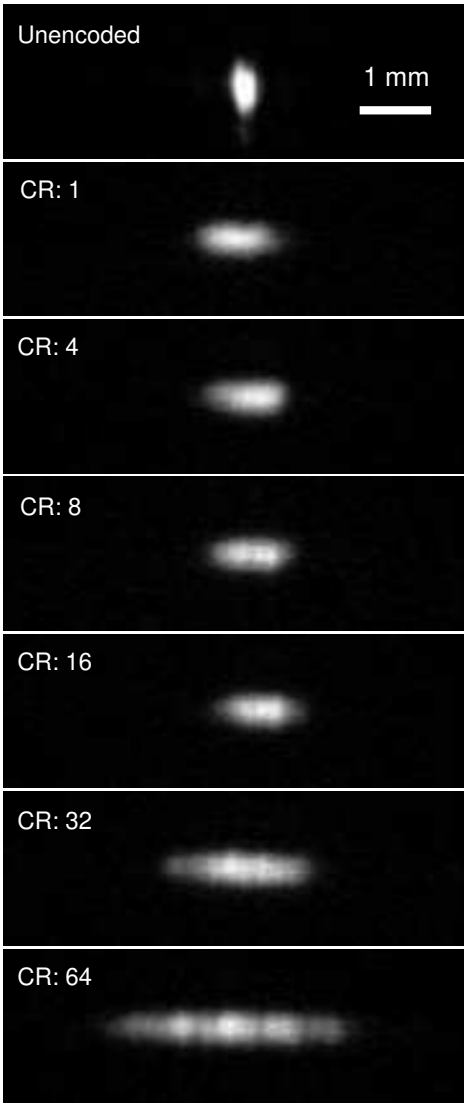
Image resolution is limited by the blur added using hogel-vector encoding. Point spread is a function of image depth and of encoding parameters, as shown in the illustrations on pages 88-90. The succeeding subsection discusses the point spread caused by hogel-vector encoding.

Hogel-Vector Encoding:
Point Imaged at z=80 mm, w_h=0.300 mm (N_h=512)

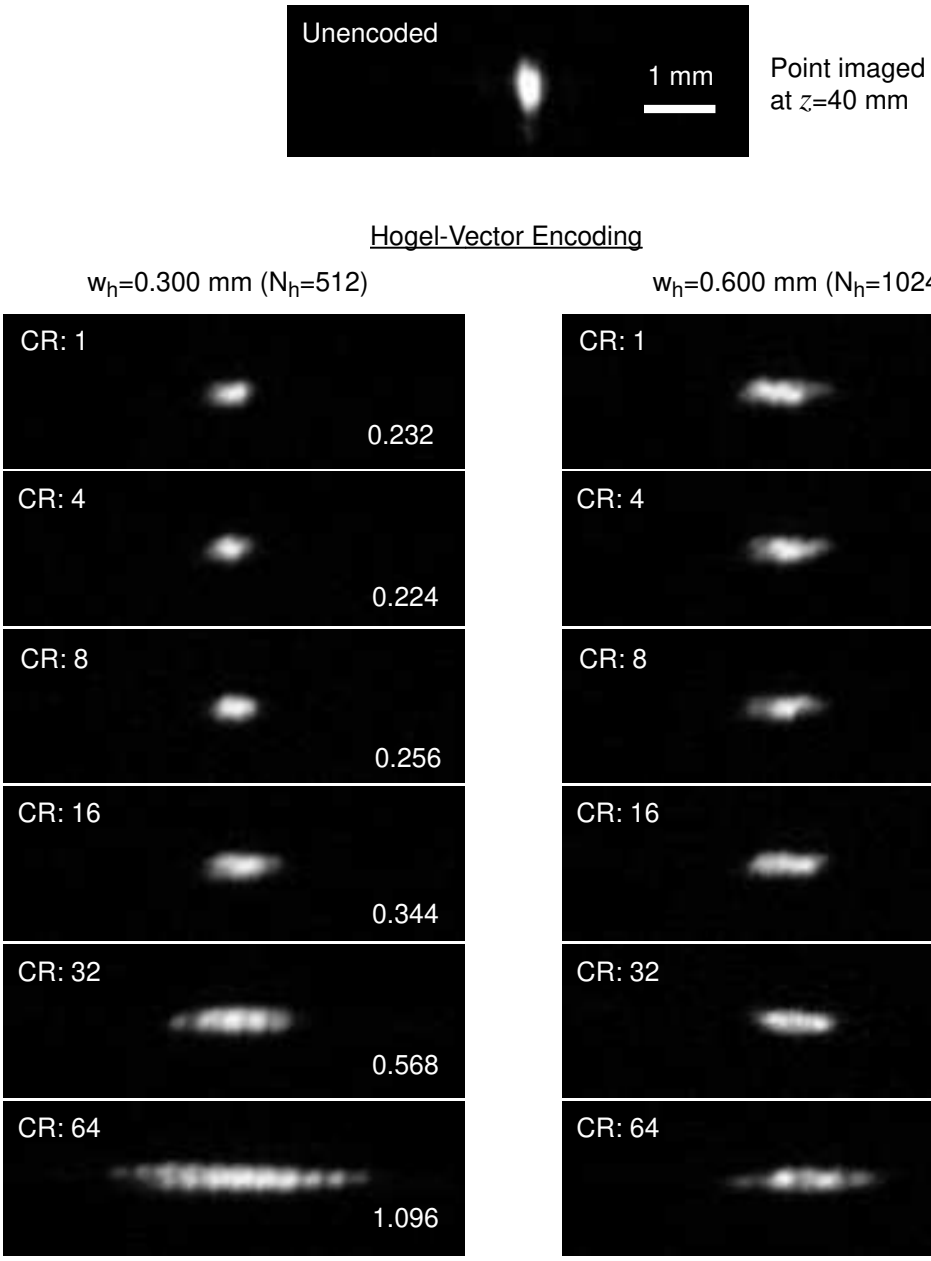


This figure shows a series of cross-sections of a point imaged at z=80 mm for a range of compression ratios. The point blurs (horizontally) as CR increases. For CR of 8 or lower, the point is still relatively sharp. For compression of CR=16, the point begins to blur to a width that is easily seen by a human viewer. However, in many cases this blur is an acceptable trade-off for increased computation speed. Speed of computation is directly proportional to CR. The CR=16 point, though blurred to 0.672 mm (less than 5 times the unencoded point width) required 1/16 the computation time. Hogel width for all of these images was w_h=0.300 mm, or N_h=512 samples.

Hogel-Vector Encoding:
Point imaged at z= 80 mm, w_h=0.600 mm (N_h=1024)



This figure shows a series of cross-sections of a point imaged at z=80 mm. This is similar to the previous figure, except that the hogel width in this case was w_h=0.600 mm, or N_h=1024 samples (twice the previous figure). This figure illustrates that setting w_h=0.600 blurs the point more for low compression ratios. However, for CR=16 and higher the point was less blurred. Clearly, for a CR=16, the choice of w_h=0.600 mm is better than a hogel width half this size. Although point spread of roughly 0.5 mm limits image resolution to less than the acuity of the HVS, recall that this is the worst-case image. Points nearer to the holoplane (i.e., |z|<80 mm) have smaller point spread.



This figure shows a series of point images focused at z=40 mm. To further illustrate the effect of choosing the correct hogel width, the point is shown for hogel widths of 0.300 mm and 0.600 mm over the range of compression ratios. The measured effective widths (in mm) are listed in the lower right corner of each cross-section. Notice that the point spread is generally less at z=40 mm than for points at z=80 mm. Although the two values of w_h are more similar than in the case of the z=80 mm point, the choice of w_h=0.600 mm gives less point spread at high values of CR.

5.4 Discussion of Point Spread

The blur added when using hogel-vector encoding is the result of several processes:

- spectral sampling blur due to more coarsely sampling hogel spectra;
- aperture effects;
- aberrations in the display;
- quantization and other noise.

The total increase in point spread is equivalent to a reduction in image resolution.

Contribution to blur from spectral sampling is a function of the spectral sampling width,

$$\Delta_f = pBW\frac{CR}{N_h} \quad , \tag{19}$$

and is approximated by the following expression:

$$blur_{\text{spectr.sampl.}} = z\lambda pBW\frac{CR}{N_h} \tag{20}$$

where BW is the total spectral bandwidth (in cycles/samples), z is the distance (in mm) of the point from the hologram plane and p is the fringe sampling pitch in samples/mm. The expression N_h/CR the number of components in a hogel vector, and the expression CR/N_h is the fraction of the spectrum represented by each hogel vector component. As CR increases or as N_h decreases, spectral sampling increases, and each component carries information about a wider region of the hogel spectrum. Light is diffracted in a larger range of angles, limiting the achievable image resolution. Here, then, is the trade-off between bandwidth and image resolution. System bandwidth is reduced proportionally as the number of symbols per hogel vector decreases, but the trade-off is the decrease in image resolution.

Aperture effects are simply the results of the finite extent of a hogel (i.e., spatial sampling). For a beam of light diffracted from an aperture of width w_h, the width of the beam at a depth of z (mm) is

$$blur_{aperture} = \left(w_h^2 + \left(\frac{z\lambda}{\pi w_h}\right)^2 \right)^{1/2} . \quad (21)$$

Near the holoplane, the aperture effect is approximately constant, making the minimum spot size equal to the hogel width w_h. For larger values of z/w_h, spreading due to diffraction by the aperture becomes significant, adding to blur for deep points.

The spectral sampling blur and aperture blur add geometrically with other sources of blur. Blur caused by the display was measured for various z locations in the image volume. Combining, the model for point spread (minimum spot size) becomes

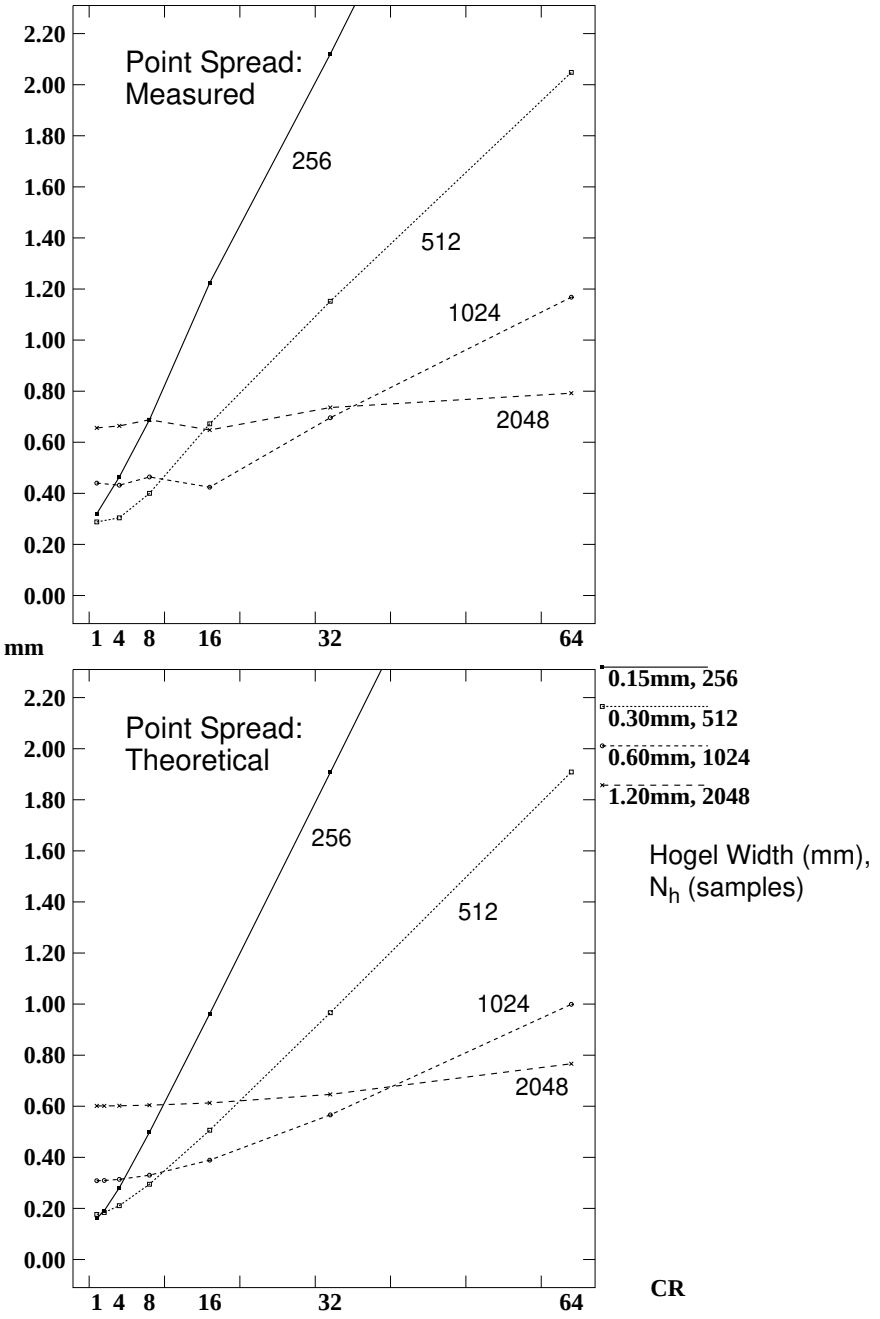
$$blur_{total} = \left(w_h^2 + \left(\frac{z\lambda}{\pi w_h}\right)^2 + \left(z\lambda pBW\frac{CR}{N_h}\right)^2 + blur_{display}^2 \right)^{1/2} . \quad (22)$$

Additional contributions to point spread are for now neglected. Nevertheless, the following subsection demonstrates that this point-spread model matches the experimentally measured image point spread.

5.4.1 Comparison of Theory and Experiment

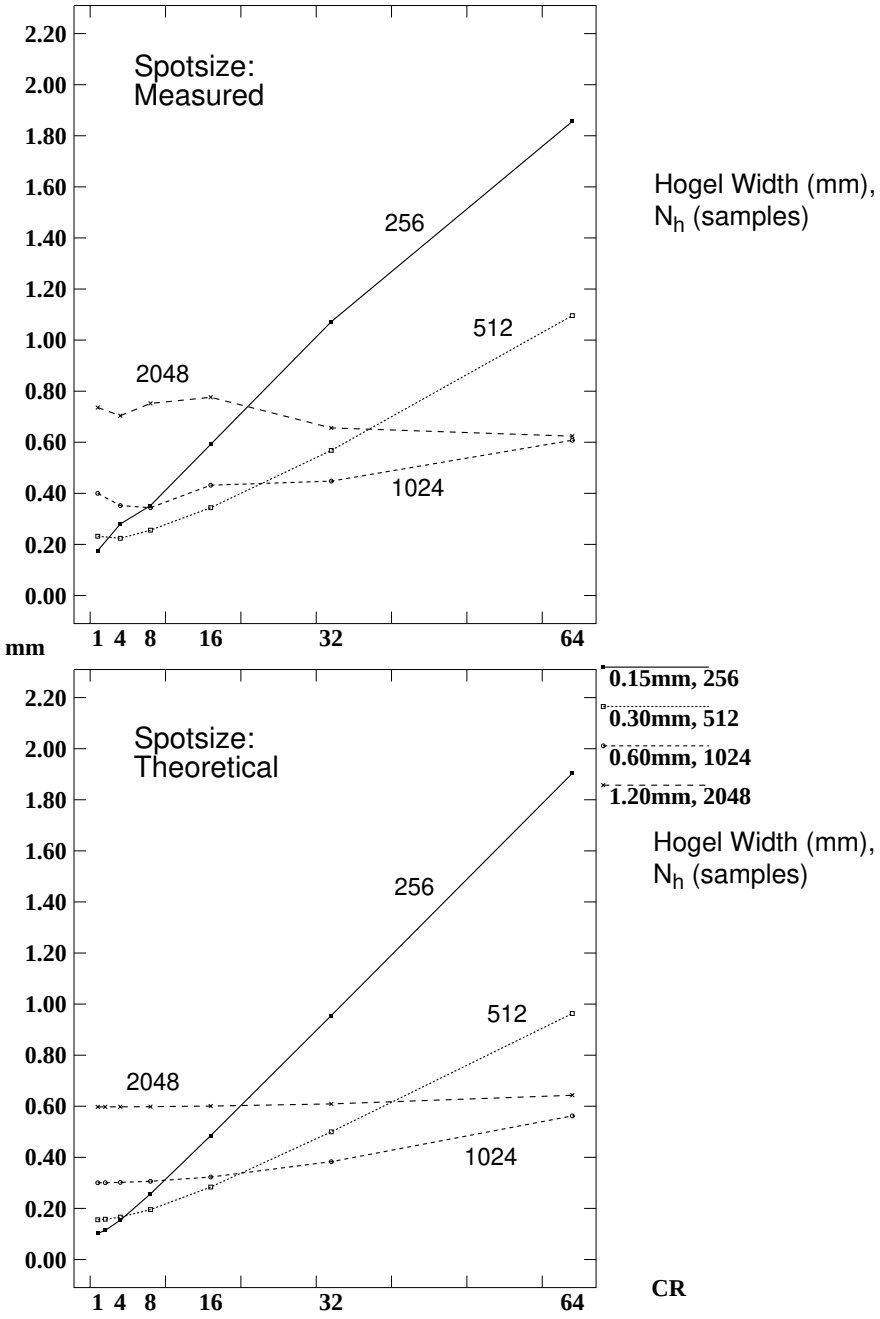
The figures on pages 93-94 compare the point-spread model (Equation 22) to the values measured from points displayed on the MIT display.

Measured and Theoretical Point Spread vs. Compression Ratio: Point at z=80 mm



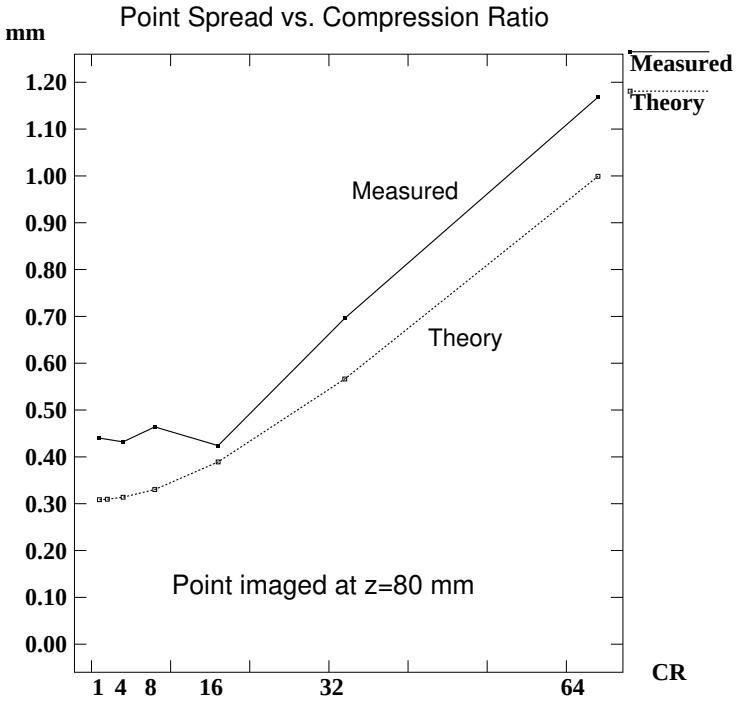
The top graph is a plot of the measured point width versus compression ratio for a point imaged at z=80 mm. The bottom graph is a plot of the values derived from the theoretical model for point spread. Each plotted line is a different hogel width.

Measured and Theoretical Point Spread vs. Compression Ratio: Point at z=40 mm



The top graph is a plot of the measured point width versus compression ratio for a point imaged at z=40 mm. The bottom graph is a plot of the values derived from the theoretical model for point spread. Each plotted line is a different hogel width.

Qualitatively, the point spread model fits very well to the measured data. In absolute terms, there appears to be an additional source of blur for some values of CR. To illustrate this deviation more clearly, the following graph compares the measured and theoretical point spread values for a point imaged at z=80 mm and a hogel width of w_h=0.600 mm (N_h=1024).



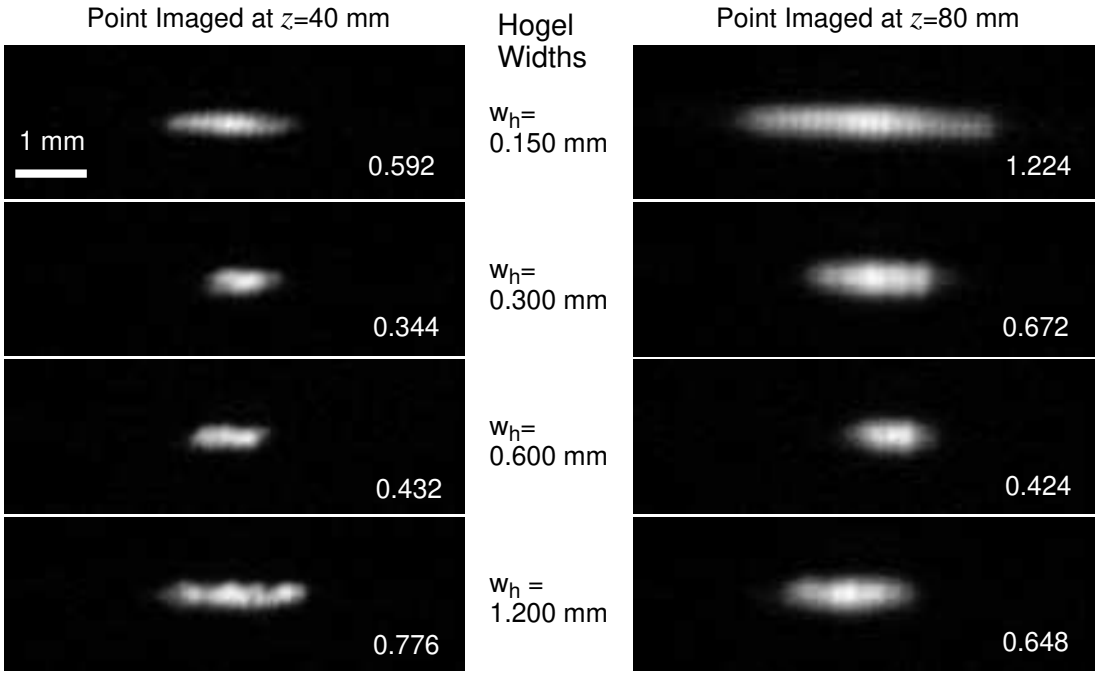
For small values of CR, this additional blur is most likely the result of quantization noise. These fringes resulted from the superposition of a large number of basis fringes. For N_h=1024 and CR=1, the number of basis functions to be accumulated is 1024. For CR=4, this number is still 256. Since the final fringe pattern was quantized to 8 bits (256 levels), the superposition of over 128 basis fringes means that each is represented by only 2 levels. For an essentially binary waveform, the signal-to-noise ratio (SNR) is only 20:1. These experiments were performed for worst-case conditions. In a typical

image, however, roughly 1/3 or the hogel vector components are zero or nearly zero, making better use of the 8-bit dynamic range of the display system.

Another likely source of the additional blur is an additional aperture effect caused by the AOM of the holovideo display. As a hogel traverses the aperture of the AOM in the form of an acoustic wave, it is clipped at the beginning an end of its path. This clipping adds to the aperture blur, and is more prominent for larger hogel widths. AOM clipping is most likely the cause of the additional blur found in hogel widths of N_h=1024 and N_h=2048 in the measured point-spread data graphed on pages 93-94.

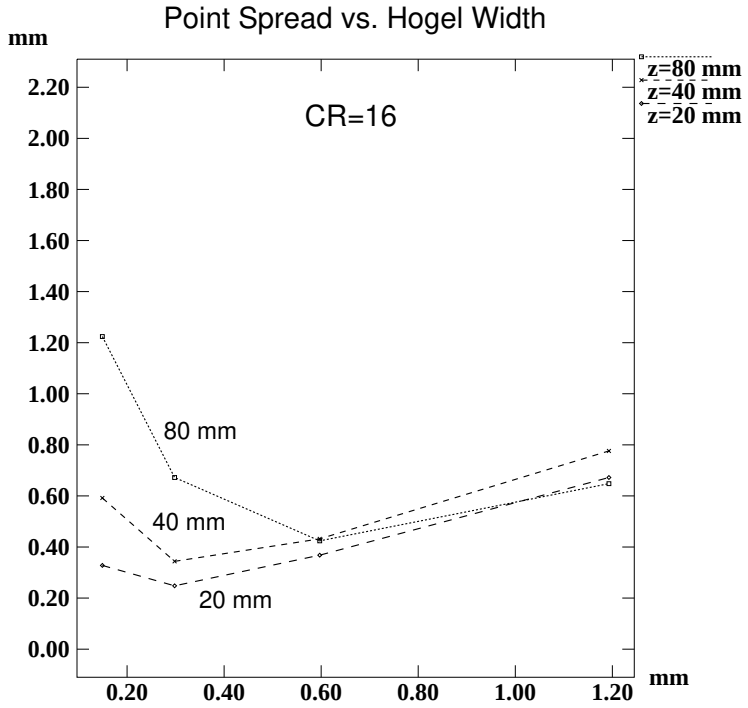
5.4.2 Empirical Selection of System Parameters

The optimal parameters can be selected for a given application. If CR=16 is required, then the model of point spread can be used to select the proper hogel width. A small hogel width causes significant blur for large z values due to spectral blur. A large hogel width keeps spectral blur at bay, but minimum spot size is limited to the hogel width. The following figure illustrates the variation in spot size as a function of hogel width. For deep points, a hogel width of 0.600 mm provides a reasonable spot size.



Compression Ratio: 16

The following graph is a subset of the measured effective point width for a point imaged at depths of z=20 mm, z=40 mm, and z=80 mm. The hogel width that ensures reasonable spot size for any depth is w_h=0.600 mm, i.e., N_h=1024. One additional feature is that for the parameters of this display, w_h=0.600 mm has a roughly constant spot size for a wide range of depths Note in this graph that the worst case of the 80-mm deep point can be unacceptably blurred. Depth is the primary struggle when compressing bandwidth. It is relatively easy to achieve high values of CR and good image fidelity when all points are near the holoplane. Though this may seem like an obvious statement, it is good to see that all of this work to compress holographic bandwidth (as opposed to 2-D image bandwidth) is actually necessary.



5.4.3 Analytical Selection of System Parameters

The model for point spread (Equation 22) is used to derive an analytical expression that relates the various parameters of the holovideo communication system. The resulting expression is used to analytically select certain parameters – such as hogel width and compression ratio – given other parameters – such as image resolution and maximum image depth.

Consider Equation 22. The proper selection of hogel width w_h ensures that total point spread is minimized given the other parameters. Let δ be the maximum allowable image point spread, and let Z be the maximum image depth. Minimizing point spread as a function of w_h yields:

$$\delta^2 = Z\lambda\left[\sqrt{c} + \frac{1}{\pi^2\sqrt{c}} + \frac{BW \cdot CR^2}{\sqrt{c}}\right] \quad (23)$$

$$c \equiv \frac{1}{\pi^2} + (BW \cdot CR)^2 \quad (24)$$

The expression simplifies under the assumption that the compression ratio (CR) is somewhat greater than one:

$$BW \cdot CR \gg \frac{1}{\pi} \quad (25)$$

$$\Rightarrow \quad \delta \approx \frac{Z}{N}(2\sqrt{2}\sin\frac{\Theta}{2}) \quad (26)$$

given that the hogel width has been chosen as

$$w_h = \delta\sqrt{BW} \quad . \quad (27)$$

(This assumption is equivalent to setting equal the contributions to point spread from the aperture and spectral sampling effects. Also, for this analysis, other sources of blur are treated as negligible.) Note that the parameters of pitch (p) and wavelength (λ) have been substituted by the expression containing the size of the viewing zone (Θ). The parameter $N=N_h/CR$ is simply a measure of bandwidth in symbols per hogel.

Equation 26 is an analytical expression relating the most important parameters of a holovideo system, namely the image resolution, the image depth, the bandwidth (N) and the size of the viewing zone (Θ). To design a holovideo system, an image resolution is first chosen, setting the parameter δ. Next, the hogel width is fixed by Equation 27. Finally, minimum bandwidth N and maximum depth Z are adjusted for optimal performance using Equation 26. This analysis shows quite clearly the trade-off between image depth and required bandwidth. These concepts are explored further in Section 7.3 “Engineering Trade-Off: Bandwidth, Depth, Resolution”on page 134.

5.5 Speed

Using hogel-vector encoding, total computation time consists of the initial direct-encoding step to produce the hogel-vector array on the Onyx, the hogel-vector decoding step done with the Cheops Splotch Engine, and the time to transfer the encoded fringes (the hogel-vector array) to the Cheops system. The first step - generation of the hogel-vector array - was very fast. For simple wireframe objects, typical times were less than 1.0 s for a CR of 16. The downloading of the hogel-vector array over the SCSI link is slow, even with a 2.25-MB hogel-vector array (CR=16), the data transfer required approximately 2.0 s. This time should reduce to a negligible amount with the future introduction of a high-speed data link.

The decoding step required the greatest amount of computing time. Each fringe sample in the resulting hogel requires N_h/CR MACs. For compression of 16 times (i.e., CR=16), the decoding step requires that for each of 144x512 hogels, 32 basis fringes of length N_h=512 bytes be multiplied and accumulated to produce a 36-MB fringe. This is approximately 1.2 GMACS, i.e., over 1 billion multiplies and 1 billion adds. When implemented on a single Splotch Engine, hogel-vector decoding time was 20 s. (These timings were worst case, measured using a fully non-zero hogel-vector array. Typical test images were closer to 9 s owing to their more sparse hogel vectors.) Compared to the (unencoded) diffraction-specific computation speed in the previous chapter, hogel-vector encoding was a factor of 16.0 times faster. This was due to the reduction by a factor of 1/CR in the number of time-consuming MAC calculations required for each fringe sample. Since the number of MACs decreases with higher compression ratios, the speed of hogel-vector decoding increases linearly with CR. Faster speeds can be achieved by sacrificing image quality for a reduction in information content.

As previously noted (Section 4.7), the Splotch Engine should be able to achieve a factor of two speed increase once it has been reconfigured to handle hogel-vector decoding. Also, because the Cheops P2 board can contain three Splotch Engines, decoding

time can in the future be reduced by two thirds. This potentially brings total computation time down to about 5 s total, including all transfer times, using hogel-vector encoding implemented on the Onyx/Cheops/Splotch platform. For comparison, a Connection Machine Model 2 with 16 Kprocessors required 21.9 s for the computation task in the previous paragraph (not including the transfer time from the CM2 to Cheops). In other words, for hogel-vector decoding, a single Cheops P2/Splotch combination provides computing power equivalent of a CM2 - all on one convenient and portable board.

The following listing summarizes the timings for hogel-vector encoding and decoding. The first two numbers indicate times for direct-encoding and for decoding. These times sum to total computing time (excluding transfer time).

- Onyx → Cheops Splotch Engine: 1 s + 20 s = 21 s
- CM2: 1 s + 22 s = 23 s
- SCSI transfer time: add 2 s

5.6 Conclusion

Hogel vectors are a natural choice for a fringe encoding format. Recall the three important features characterize an information-efficient diffraction-specific hogel description: (1) the computation of a hogel can proceed using nothing more than a hogel vector; (2) no two hogels computed from different hogel vectors give rise to the same viewer stimulus; (3) no two different hogel vectors give rise to the same viewer stimulus. Hogel vectors increase the amount of entropy per symbol.

Redundancy is inherent in physically useful fringes. The fringe pattern, encoded as the hogel-vector array, contains reduced redundancy and increased symbol entropy. Thus, the bandwidth is reduced until the final computation step, decoding.

Speed is still a problem. Although speed has increased by over an order of magnitude, the fastest typical time of 9 s is still beyond the reach of interactive computing. The decoding step accounts for most of the total computing time. In the next chapter, a second holographic encoding scheme - “fringelet encoding” - is developed specifically to increase the simplicity and speed of decoding.