Compression of digital holograms for three-dimensional object recognition

Thomas J. Naughton[a,b], Yann Frauel[b], Bahram Javidi[b], Enrique Tajahuerce[c]

[a]Department of Computer Science, National University of Ireland, Maynooth, Ireland
[b]Department of Electrical and Computer Engineering, University of Connecticut, U-157, Storrs, Connecticut 06269-2157, USA
[c]Departament de Ciències Experimentals, Universitat Jaume I, E-12080 Castelló, Spain

ABSTRACT

In this paper we present the results of applying data compression to a three-dimensional object recognition technique based on phase-shift digital holography. Industry-standard lossless data compression algorithms were first applied. Next, lossy techniques based on subsampling, discrete cosine transformation, and discrete Fourier transformation were examined. We used normalized cross-correlation in the object plane as our performance metric. For each hologram tested, we found that as many as 90% of the cosine and Fourier components could be removed, without significant loss in correlation performance.

Keywords: three-dimensional image processing, computer holography, data compression, pattern recognition, discrete cosine transform.

1. INTRODUCTION

Correlation is a widely used metric for the comparison of images and in the detection of objects (or regions of interest) in scenes. Usually, two-dimensional images are compared.[1,2] Recently, three-dimensional (3D) object recognition techniques have appeared[3–7] including those based on digital holography.[8–11] In our system, we create a digital hologram using phase-shifting interferometry.[12,13] This hologram encodes multiple views of the object from a small range of angles. Different views of the object can be constructed by extracting the appropriate window of pixels from the hologram and applying a numerical propagation technique.[13,8] These views could be combined as a composite filter[10] or in a filter bank. In this paper, we investigate the most effective way of compressing these holograms for transmission (or storage), prior to any correlation operation, and without knowledge of which views will eventually be required. With uncompressed sizes of almost 65 Mbytes per 4 Mpixel hologram, when stored in standard Matlab (The Mathworks) double precision format, this becomes a valuable exercise. In the absence of any compression strategy, a 100 Mbit/s network connection would require over 5 s to transmit each hologram. This is too slow for realtime object recognition.

We apply lossless and lossy data compression techniques to the digital holograms. When applying lossy techniques, we do not examine the error in the decompressed hologram directly, but instead examine the error in the resultant object reconstruction. As one of our main applications for digital holography is 3D pattern recognition[8–10] we choose normalized cross-correlation as our metric for reconstruction integrity. In Sect. 2, we describe phase-shift digital holography and present our experimental setup. In Sect. 3, we apply standard lossless data compression techniques. The simplest form of lossy compression, that of resampling or image resizing, is examined in Sect. 4. In Sect. 5, we use a discrete cosine transform (DCT) technique to selectively remove cosine components from 8×8 blocks of the hologram. This technique is repeated with the discrete Fourier transform (DFT), before concluding.

2. PHASE-SHIFT DIGITAL HOLOGRAPHY

We record digital holograms with an optical system based on a Mach-Zehnder interferometer (see Fig. 1). A linearly polarized Argon ion (514.5 nm) laser beam is divided into object and reference beams, both of which are expanded and spatially filtered. The first beam illuminates a reference object placed at a distance $d = 350$ mm from a 10-bit

Further author information: (Send correspondence to B.J.) T.N.: tomn@cs.may.ie Y.F.: frauel@engr.uconn.edu B.J.: bahram@engr.uconn.edu

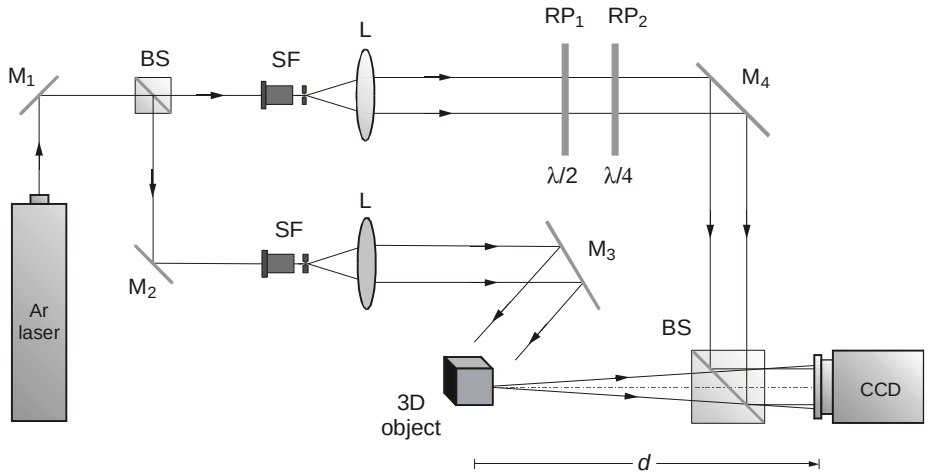

Figure 1. Experimental setup for phase-shift digital holography: M, mirror; BS, beam splitter; SF, spatial filter; L, lens; RP, retardation plate.

2028×2044-pixel Kodak Megaplus CCD camera. We refer to the complex amplitude distribution in the plane of the object as $U_0(x, y)$. The reference beam passes through half-wave plate RP_1 and quarter-wave plate RP_2. The linearly polarized beam can be phase-modulated by rotating the two retardation plates. Through permutation of the fast and slow axes of the plates we can achieve phase shifts of 0, $\frac{\pi}{2}$, π, and $\frac{3\pi}{2}$. The reference beam combines with the light diffracted from the object and forms an interference pattern in the plane of the camera. At each of the four phase shifts we record an interferogram. We use these four real-valued images to compute the camera-plane complex field $H_0(x, y)$ by phase-shift interferometry.[12] We call this computed field a digital hologram. Our holograms have dimensions 2028×2044 pixels and are originally in floating point representation, with 8 bytes of amplitude information and 8 bytes of phase angle information for each pixel.

A digital hologram $H_0(x, y)$ contains sufficient amplitude and phase information to reconstruct the complex field $U(x, y, z)$ in a plane in the object beam at any distance z from the camera.[13,8] This can be calculated using the Fresnel-Kirchhoff formula

$$U(x, y, z) = H_0(x, y) \star h(x, y, z) \ , \tag{1}$$

where

$$h(x, y, z) = -\frac{\mathrm{i}}{\lambda z} \exp\left(\mathrm{i}\frac{2\pi}{\lambda} z\right) \exp\left(\mathrm{i}\pi \frac{(x^2 + y^2)}{\lambda z}\right) \tag{2}$$

is the point spread function for free space, λ is the wavelength of the illumination, and $\star$ denotes a convolution operation. At $z = d$, and ignoring errors in digital propagation due to discrete space (pixelation) and rounding, the digital reconstruction $U(x, y, z)$ approximates $U_0(x, y)$. Furthermore, as with conventional holography, a windowed subset of the hologram can be used to reconstruct a particular view of the object as shown in Fig. 2. As the window explores the hologram a different angle of view of the object can be reconstructed. The range of viewing angles is determined by the ratio of the window size to the full CCD sensor dimensions. In our system, a 1024×1024-pixel window has a maximum lateral shift of 9 mm over the face of the CCD sensor.[10] With an object positioned $d = 350$ mm from the camera, viewing angles in the range $\pm 0.74°$ are permitted.

The problem we address can be stated as follows. A digital hologram H_0 of some object U_0 is to be compressed and transmitted from sender to receiver (see the illustration in Fig. 3). At the receiver, the hologram is decompressed as H_0' and an object U_0' reconstructed by numerical propagation. We wish to determine the most effective way of compressing H_0 such that U_0' correlates strongly with U_0.

3. LOSSLESS DATA COMPRESSION

Lossless data compression techniques are used in situations where the data must be faithfully decompressed, such as in text compression. If we use lossless techniques we are assured that U_0' will be identical to U_0, apart from rounding and pixelation errors. The set of 3D objects used in the compression experiments is shown in Fig. 4. The

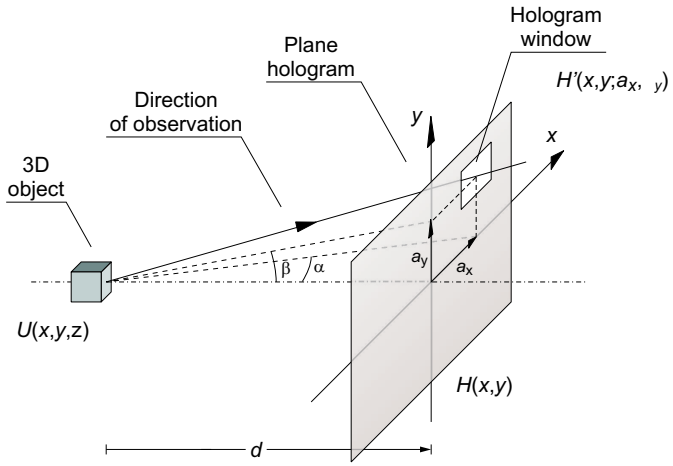

Figure 2. Relationship between the hologram window and angle of view.

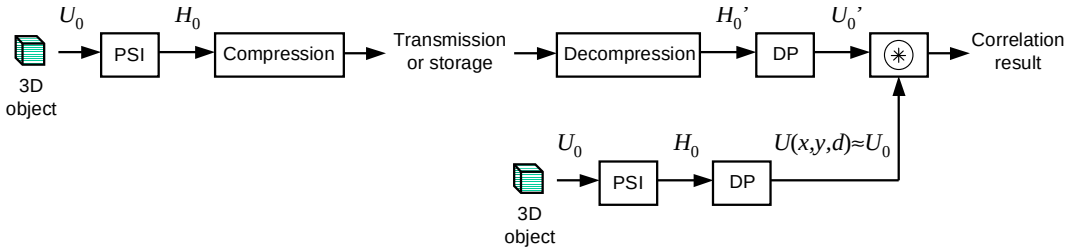

Figure 3. Illustration of the problem statement: PSI, image capture and phase-shift interferometry stage; DP, digital propagation (reconstruction) stage; ⊛, normalized cross-correlation operation. Digital hologram H_0 must be compressed and transmitted such that decompressed and reconstructed U_0' correlates strongly with the approximation of the original complex amplitude distribution $U(x, y, d)$.

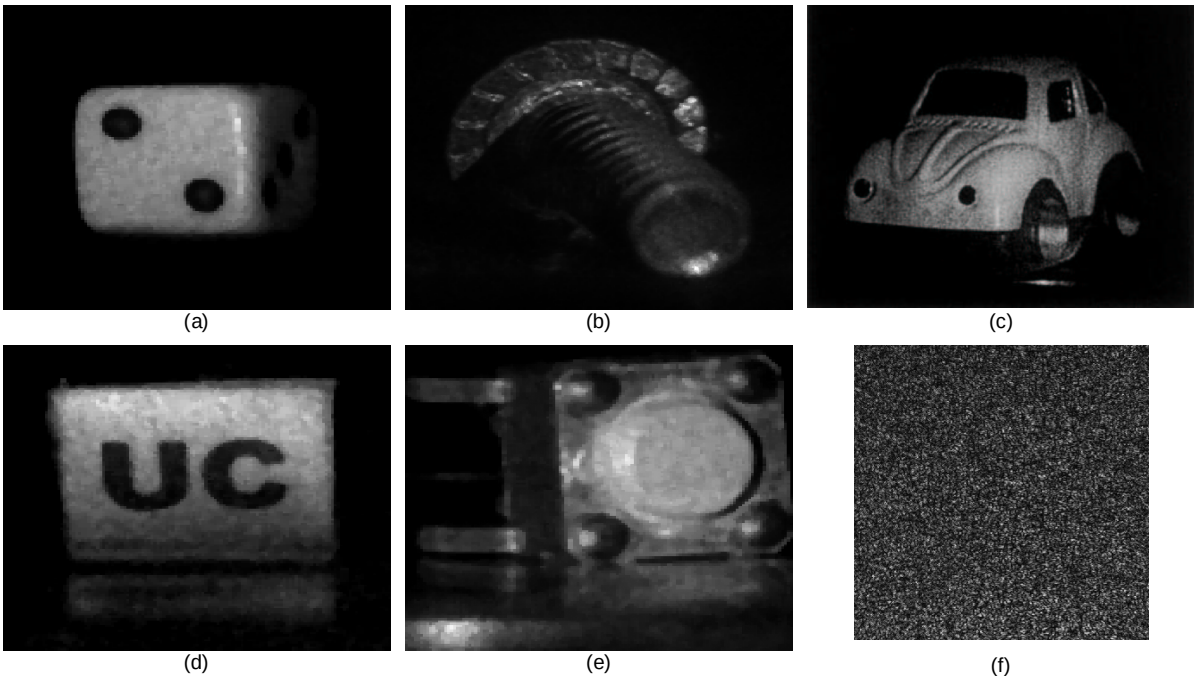

Figure 4. The set of holograms used in these experiments: (a) through (e) are the amplitudes of the reconstructed wavefronts for holograms no. 1 through no. 5, respectively. Image (f) shows the amplitudes of an example 512 × 512 subset of digital hologram no. 1.

Hologram no.	Size (kB)	LZ77 (kB)	LZW (kB)	Huffman (kB)	LZ77 compression rate	LZW compression rate	Huffman compression rate
1	64769	52038	64769	62236	1.24	1.00	1.04
2	64769	62353	64769	62298	1.04	1.00	1.04
3	64642	32718	54766	61784	1.98	1.18	1.05
4	64769	54923	64769	62262	1.18	1.00	1.04
5	64769	53608	64769	62267	1.21	1.00	1.04
Averages:					1.33	1.04	1.04

Table 1. Compression with LZ77, LZW, and Huffman coding, when the hologram is treated as a binary data stream with alternating amplitude and phase angle components, achieves compression rates in the range $[1.0, 1.98]$.

digital holograms were treated as binary data streams. Three of the most common industry-standard compression techniques were chosen: Huffman coding,[14] Lempel-Ziv coding[15] (LZ77), and Lempel-Ziv-Welch coding[16,17] (LZW).

Huffman coding,[14] an entropy based technique, is one of the oldest and most widely used compression methods. It replaces each symbol in the input by a codeword, assigning shorter codewords to more frequent symbols. Huffman coding is often used for text compression and is used as the final stage in baseline JPEG (Joint Photographic Experts Group) compression.[18,19] It has a standard implementation in the form of the UNIX pack utility. Huffman coding is an example of a variable length code, replacing input blocks of fixed size by variable length strings of bits.

The LZ77 algorithm[15] takes advantage of repeated substrings in the input data. In contrast to Huffman coding, a variable length string of input symbols is replaced by a fixed-size codeword (a reference to the previous occurrence of that string). Variants of LZ77 apply additional compression to the output of the substitution stage. A statistical compressor such as Huffman is often chosen to take advantage of the unequal probabilities of output codewords. This is the combination used in gzip, LHa, Zip and others, and in the PNG (Portable Network Graphics) specification. For our experiments we used GNU gzip.

The LZ77 algorithm has a shortcoming in that the way it references repeated substrings makes it biased towards local redundancy. In LZ78 coding,[16] a dictionary (or lookup table) of previously encountered substrings is maintained to make more efficient use of the reference codewords. This substantially reduces the number of comparisons required during each encoding step. The LZW algorithm[17] is a popular refinement of LZ78, permitting variable sized codewords and the deletion of old strings from the dictionary [both of which are implemented with the standard UNIX compress utility and the GIF (Graphics Interchange Format) specification]. We used an implementation of compress in our experiments.

The results of using these lossless algorithms are shown in Table 1. Treating the holograms (each a sequence of pairs of amplitude and phase values) as binary data streams achieves compression rates in the range $[1.0, 1.98]$, where the compression rate r is calculated from

$$r = \frac{\text{uncompressed size}}{\text{compressed size}} \ , \tag{3}$$

and where a rate of 1.0 was used when no compression was achieved, or in cases where the 'compressed' hologram was actually larger in size. By the term compression rate we mean the number of bits of uncompressed data that are effectively communicated with a single bit of compressed data. As shown in Table 1, on average, each bit of compressed data represented 1.33 bits of uncompressed holographic data for LZ77 encoding, and 1.04 bits for LZW and Huffman coding. Given their size, it is worrying how poorly the holograms compress using these techniques. Their performance is possibly due to the noisy influence of speckle in the hologram data [see Fig. 4(f)]. In general, the fact that Huffman is outperformed by the Lempel-Ziv algorithms is no surprise. However, that simpler and locally-biased LZ77 consistently outperforms LZW could tell us something about the nature of the holographic data and about the local characteristics of its speckle content.

3.1. Intermediate coding

For some compression algorithms, the way data is represented can have a great impact on how well it will be compressed. We investigate three additional representations (or intermediate codings) for the digital holographic

Hologram no.	Size (kB)	LZ77 (kB)	LZW (kB)	Huffman (kB)	LZ77 compr. rate	LZW compr. rate	Huffman compr. rate
1	64769	21380+25608	32385+32385	30285+31163	1.38	1.00	1.05
2	64769	30521+31308	32385+32385	30476+31296	1.05	1.00	1.05
3	64642	11967+16580	17244+25969	30155+30889	2.26	1.50	1.06
4	64769	23076+27108	32385+32385	30293+31173	1.29	1.00	1.05
5	64769	21825+26560	32385+32385	30252+31215	1.34	1.00	1.05
Averages:					1.46	1.10	1.05

Table 2. Compression with LZ77, LZW, and Huffman coding, when the hologram is treated as two separate binary data streams: the first containing amplitude components and the second containing phase angle components. This technique achieves compression rates in the range $[1.0, 2.26]$.

Hologram no.	Size (kB)	LZ77 (kB)	LZW (kB)	Huffman (kB)	LZ77 compression rate	LZW compression rate	Huffman compression rate
1	64769	18442	17819	27038	3.51	3.63	2.40
2	64769	60885	64769	62008	1.06	1.00	1.04
3	64642	15024	16071	54513	4.30	4.02	1.19
4	64769	18580	17902	26867	3.49	3.62	2.41
5	64769	18207	17610	26548	3.56	3.68	2.44
Averages:					3.18	3.19	1.90

Table 3. Compression with LZ77, LZW, and Huffman coding, when the hologram is treated as a binary data stream with alternating real and imaginary components, achieves compression rates in the range $[1.0, 4.3]$.

Hologram no.	Size (kB)	LZ77 (kB)	LZW (kB)	Huffman (kB)	LZ77 compr. rate	LZW compr. rate	Huffman compr. rate
1	64769	9356+9127	8931+8791	13899+13079	3.50	3.65	2.40
2	64769	29890+29752	32385+32385	30987+31009	1.09	1.00	1.04
3	64642	7254+8180	7747+8318	27154+27282	4.19	4.02	1.19
4	64769	9465+9125	9047+8805	13815+13027	3.48	3.63	2.41
5	64769	9227+9061	8839+8718	13439+13082	3.54	3.69	2.44
Averages:					3.16	3.20	1.90

Table 4. Compression with LZ77, LZW, and Huffman coding, when the hologram is treated as two separate binary data streams: the first containing real components and the second containing imaginary components. This technique achieves compression rates in the range $[1.0, 4.19]$.

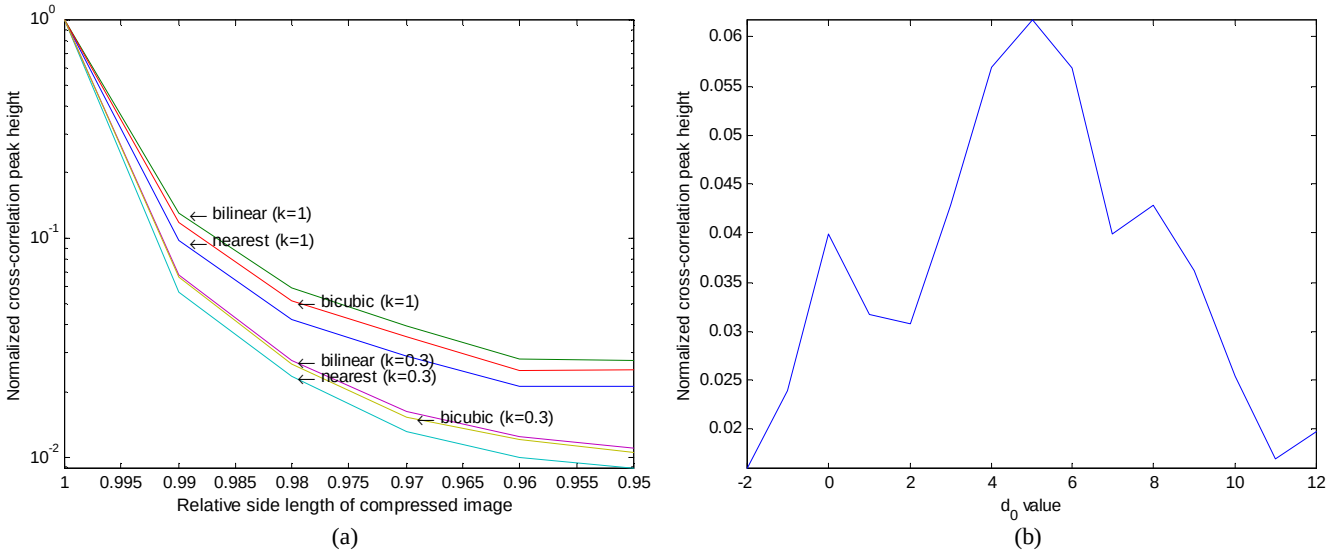

Figure 5. Resampling of the digital hologram: (a) a plot of hologram side length (relative to the original side length) against linear and nonlinear correlation performance for three interpolation strategies; (b) searching the z-axis for an appropriate d_0 offset for a hologram resize of 0.97 and bilinear interpolation.

data. First, the amplitude and phase values are collated into two separate data streams to exploit the possible redundancy between neighboring amplitude values. Table 2 shows slight but consistent improvements in compression rate for each hologram and for each compression algorithm, increasing the range to $[1.0, 2.26]$ and the LZ77 average to 1.46. For the other representations, the holographic data was transformed from (amplitude,phase) space to the equivalent, neglecting rounding errors, (real,imaginary) space. Treating the hologram as a binary data stream of pairs of real and imaginary values achieves lossless compression rates in the range $[1.0, 4.3]$, an improvement of almost a factor of 2 (see Table 3). Separating the real and imaginary components into two separate data streams gives comparable results, as shown in Table 4. From Table 4, it is also evident that both the real and imaginary streams are equally difficult to compress.

Since Huffman's algorithm exploits the frequency of occurrence of binary words rather than their relative positioning, it is no surprise that the changes in representation proposed here did not have an impact on the Huffman compression results. For algorithms that exploit the repeated occurrence of substrings (LZ77 and LZW), compression rates can be doubled by initially coding the digital hologram as a binary stream of real-imaginary values, rather than the amplitude-phase values produced by default by phase-shift interferometry. If lossless compression of digital holograms is required, then these results indicate that the hologram should be stored as a binary data stream of separate real and imaginary matrices, and compressed with LZW for best average performance (with an expected average compression rate of 3.2).

3.2. Lossless image compression

Standard lossless image compression algorithms can be applied to the digital holograms. This would require some intermediate processing to convert each complex-valued hologram into two positive real-valued images. The performance of GIF and TIFF-LZW (Tagged Image File Format) compression can be gauged by looking at the results for the LZW algorithm above, where an average compression rate of 3.2 can be achieved. The performance of PNG can be gauged by looking at the results for the LZ77 algorithm, where a similar average compression rate was recorded. If a higher rate of compression is required then a lossy form of compression will have to be applied. Lossy systems are investigated next.

4. COMPRESSION BY RESAMPLING

The criteria for grading the performance of lossy compressors will not be the same as in conventional image compression. The errors introduced into the digital hologram as a result of lossy compression are not of direct concern; it is errors in the reconstructed object, loss of viewing angle, and so on, that are of most interest. As

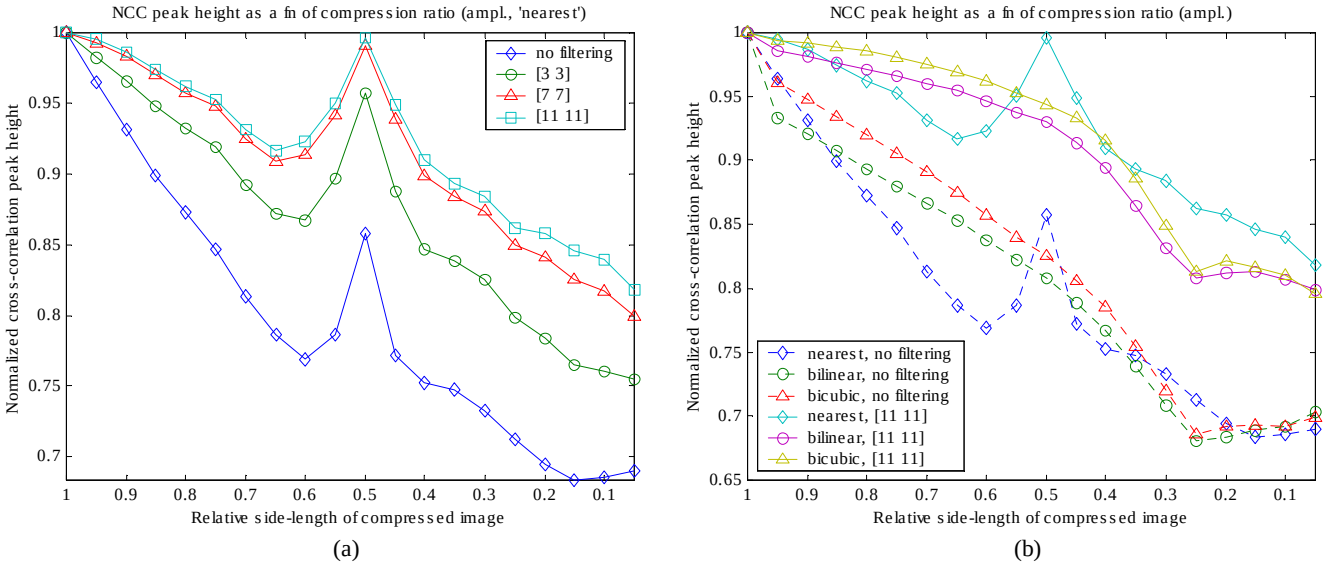

Figure 6. Resizing hologram no. 1 with nearest neighbor interpolation and then using only the amplitude information in the reconstructed object plane. Plots show normalized cross-correlation peak height: (a) the effect of resizing with median filtering neighborhoods of 3×3, 7×7, and 11×11 pixels, and with no median filtering. The large artifacts occur at exactly a resizing of 0.5 (when 4 pixels are compressed to 1); (b) the effect of resizing with three different interpolation strategies. A hybrid bicubic-nearest neighbor strategy would seem to achieve best performance for all levels of median filtering.

mentioned before, since pattern recognition is our motivation, the degree of correlation was chosen as the metric to evaluate the integrity of reconstructed images.

The simplest and most common form of lossy compression is that of resampling. In general, in optical information processing, a single image might be resampled many times: sampling by the spatial frequency of the CCD sensor, sampling of the CCIR signal by the interface card, and sampling by the pixel resolution of a monitor screen or SLM. Digital holograms are very sensitive to resampling/resizing. In Fig. 5(a), hologram no. 1 is resized using 3 different interpolation strategies prior to reconstruction of U_0'. The reconstructed object U_0' is then correlated with U_0. The figure shows normalized cross-correlation peak height on a $\log_{10}$ scale plotted against hologram side length (relative to the original side length). The plot shows peak heights for both linear and nonlinear correlation (where $k = 0.3$ is the k-th law nonlinearity[20]). Resizing the hologram to 0.97 of its previous side length causes the normalized cross-correlation peak height to fall dramatically to the order of 10^{-2} of its former height.

Resizing the hologram will cause the object to be formed at a distance different to d from the hologram plane. To compensate for this, z in Eq. (2) becomes $z = d + d_0$, where d, as before, is the original object distance from the camera and d_0 is an offset. Figure 5(b) shows the result of a search for an appropriate d_0 value. In this plot, a resizing to 0.97 of the original image side length requires a d_0 offset of $+5.0$ for maximum correlation. Even then, the correlation is poor (0.06 when normalized). This sensitivity is due in some part to speckle. The unique speckle pattern constructed in the object plane by the hologram causes a large correlation normalization factor. With the slightest modification to the digital hologram a completely different speckle pattern is reconstructed, and the value for normalized cross-correlation drops considerably. In order to reduce the effect of speckle we discard the phase information in the reconstructed object wavefront and apply a median filtering operation. Our justification for keeping only the object plane amplitude information is based on this information's dependency on both the amplitude and phase of the hologram plane: if the amplitude information in the object plane has been reconstructed correctly, this indicates that sufficient amounts of both amplitude and phase information were preserved during compression.

The following plots examine the effect of resizing a digital hologram when only the amplitude information of the reconstructed wavefront is analyzed. To avoid the need to search for an appropriate d_0 offset every time, holograms are returned to full size after the loss due to compression is introduced. (We assume that this resizing operation does not introduce additional error.) The reconstructed object amplitudes were evaluated in terms of normalized cross-correlation with $k = 1$. Figure 6(a) contains plots of normalized cross-correlation peak heights for hologram no. 1.

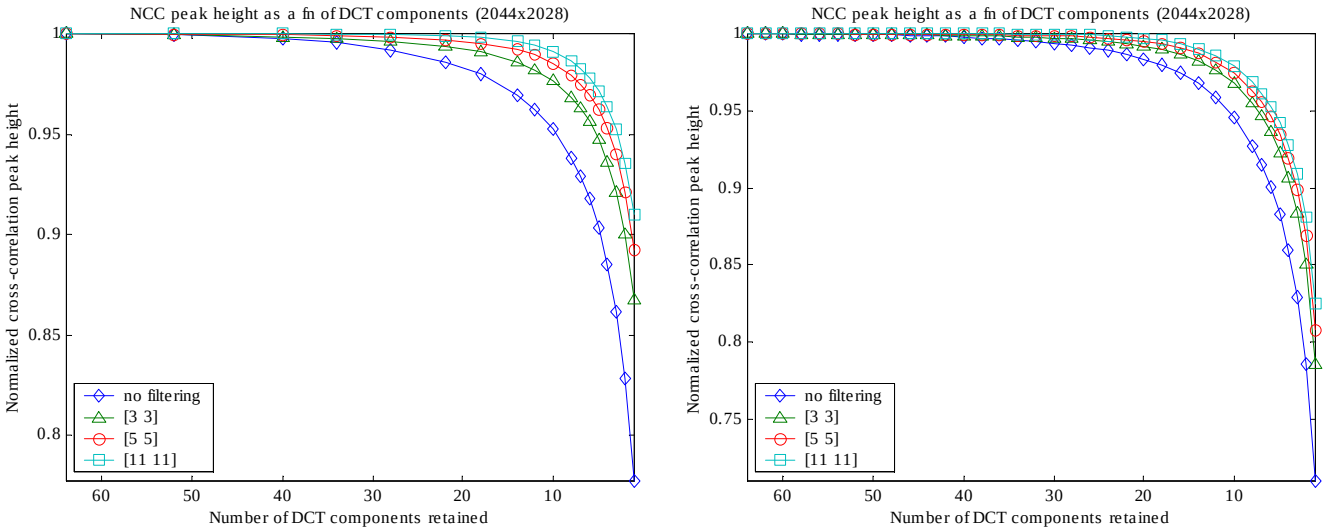

Figure 7. Normalized cross-correlation peak height as a function of DCT components retained, for various median filter neighborhoods: (a) for hologram no. 1, and (b) for hologram no. 2.

For each curve, the hologram is resampled with nearest neighbor interpolation before reconstruction, the phase information at the object plane removed, and the amplitude median filtered using one of three neighborhoods. For Fig. 6(b), the procedure was repeated, this time using only the median filter neighborhoods 1×1 (no filtering) and 11×11, and including results for bilinear and bicubic interpolation during resampling. These plots show that when only the amplitude of the reconstructed object is taken into account, the hologram is more tolerant to resampling. As the level of median filtering is increased (as the neighborhood is increased) this tolerance increases. Specific applications will dictate when too much filtering obscures salient details in the reconstructed object. Resampling, if it is acceptable for a particular application, can therefore improve the compression rate. Furthermore, since resampling simply reduces the number of pixels and (possibly) changes their values rather than compresses the underlying data format, the lossless techniques of Sect. 3 can be applied as a final step.

Taking an example from Fig. 6(b), if a normalized cross-correlation of at least 0.98 is required (and 11×11-pixel median filtering is acceptable) for a particular application, then bicubic interpolation can resize to a side length of 0.75. This reduces the number of pixels by a factor of 1.78, which combined with the average performance of the lossless techniques gives a compression rate of 5.7. The technique of resampling achieves its best performance at 0.5 resizing, with 11×11-pixel filtering and nearest neighbor interpolation, where a compression rate of 12.8 can be averaged.

5. DCT- AND DFT-BASED COMPRESSION

The DCT operation[21] is at the heart of the JPEG algorithm.[18,19] JPEG performs most of its compression through quantization in the cosine domain. The JPEG standard is defined only for real images, but we adapt it to holograms, taking the DCT in turn of each block of 8×8 pixels. This does not reduce the amount of information in the image but it does tend to concentrate the majority of the hologram information into a few DCT components in each block. By quantizing (or setting to zero, in our case) particular coefficients in each block we reduce the length of its bit description and allow an entropy coder (such as Huffman or a run-length technique) to compress effectively the DCT components. In our quantization technique, we set a fixed number of the smallest DCT components in each block to zero. This involves sorting the values in each 8×8 DCT and setting the $(64 - n)$ smallest components to zero, where n is a positive integer in the range $[1, 64]$ denoting the number of DCT components to be retained in each block. Sorting of the DCT components was performed by amplitude first, and then (if necessary) by phase angle. Upon decompression, the hologram was used to construct object U_0' for correlation with U_0. Linear correlation ($k = 1$) was employed. Once again, the phase of the object wavefront was discarded and the amplitude median filtered prior to correlation to lessen the effects of speckle.

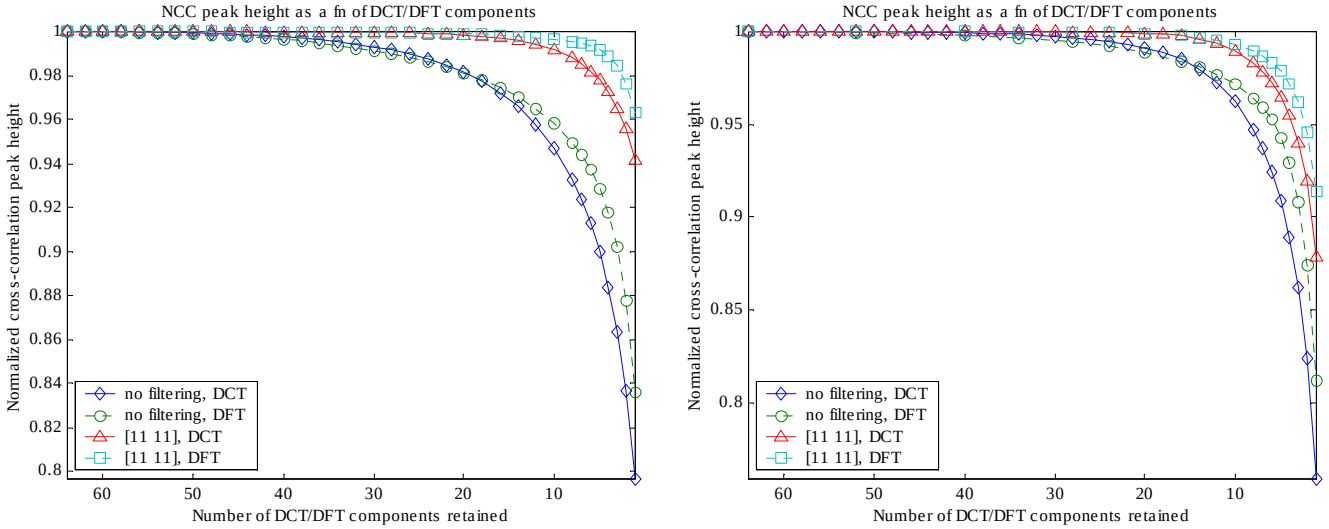

Figure 8. Plots of normalized cross-correlation peak height against number of DCT and DFT components retained in each 8×8 block, with both 11×11-pixel neighborhood median filtering and no median filtering: (a) hologram no. 1, and (b) hologram no. 2.

Figure 7 shows the normalized cross-correlation peak heights (for holograms no. 1 and no. 2) for various values of n, and for different median filtering neighborhoods. The results are consistent over both holograms: as many as 90% of the DCT components can be removed without significant loss in correlation performance. For hologram no. 1, as many as 46 DCT components can be removed from each 8×8 (when no median filtering is applied) for a normalized correlation of 0.98, and as many as 57 removed with an 11×11 median filter for the same performance. This results in expected average compression rates of 3.5 and 9.0, respectively. The remaining nonzero-valued DCT components are each stored with 8 bytes, and so the lossless techniques of Sect. 3 can be used to compress them even further.

The DCT is derived from the DFT, and is convenient for use with images since a real-valued input results in a real-valued DCT. However, since we are working with complex-valued signals, it would seem more appropriate to use the DFT to generate the spatial frequency representations. The experiments were repeated with the DFT. It was found that at the highest levels of compression, and for the same correlation performance, the DFT consistently allowed more frequency components to be removed than the DCT. This can be seen in Fig. 8, where the DFT and the DCT are compared for two levels of median filtering, for each of two holograms.

6. CONCLUSION

We have investigated various techniques for the compression of digital holograms created by phase-shifting interferometry. With industry-standard lossless data compression techniques an average lossless compression rate of 3.2 can be expected. This rate was achieved by an intermediate coding of separated real and imaginary components. Lossy resampling techniques, combined with phase-removal and median filtering to lessen the effects of speckle, were examined in terms of normalized correlation peak height. For resampling, a hybrid bicubic-nearest neighbor interpolation strategy would seem preferable, obtaining in one special case a compression rate of 12.8. Currently, a technique based on the removal of DCT components achieves approximate compression rates of up to 9.0 depending on the degree of median filtering allowed. It is anticipated that this can be improved by applying a lossless compression technique or quantization to the remaining DCT components. In addition, we showed that at the highest compression rates, correlation performance can be further improved by use of the DFT instead of the DCT.

ACKNOWLEDGMENTS

TN acknowledges support from Enterprise Ireland grant no. IC/2001/046, the Department of Electrical and Computer Engineering, UConn, and the Department of Computer Science, NUIM.

REFERENCES

1. A. VanderLugt, "Signal detection by complex spatial filtering," *IEEE Trans.* **IT-10**, pp. 139–145, 1964.
2. J. W. Goodman, *Introduction to Fourier Optics*, McGraw-Hill, San Francisco, 1968.
3. A. Pu, R. F. Denkewalter, and D. Psaltis, "Real-time vehicle navigation using a holographic memory," *Opt. Eng.* **36**, pp. 2737–2746, 1997.
4. R. Bamler and J. Hofer-Alfeis, "Three- and four-dimensional filter operations by coherent optics," *Opt. Acta* **29**, pp. 747–757, 1982.
5. J. Rosen, "Three-dimensional joint transform correlator," *Appl. Opt.* **37**, pp. 7538–7544, 1998.
6. J. J. Esteve-Taboada, D. Mas, and J. García, "Three-dimensional object recognition by Fourier transform profilometry," *Appl. Opt.* **38**, pp. 4760–4765, 1999.
7. J. Guerrero-Bermúdez, J. Meneses, and O. Gualdrón, "Object recognition using three-dimensional correlation of range images," *Opt. Eng.* **39**, pp. 2828–2831, 2000.
8. B. Javidi and E. Tajahuerce, "Three-dimensional object recognition by use of digital holography," *Opt. Lett.* **25**, pp. 610–612, 2000.
9. E. Tajahuerce, O. Matoba, and B. Javidi, "Shift-invariant three-dimensional object recognition by means of digital holography," *Appl. Opt.* (accepted March 2001).
10. Y. Frauel, E. Tajahuerce, M.-A. Castro, and B. Javidi, "Distortion-tolerant 3D object recognition using digital holography," *Appl. Opt.* (accepted May 2001).
11. Y. Frauel and B. Javidi, "Neural network for three-dimensional object recognition based on digital holography," *Opt. Lett.* (accepted July 2001).
12. J. H. Bruning, D. R. Herriott, J. E. Gallagher, D. P. Rosenfeld, A. D. White, and D. J. Brangaccio, "Digital wavefront measuring interferometer for testing optical surfaces and lenses," *Appl. Opt.* **13**, pp. 2693–2703, 1974.
13. I. Yamaguchi and T. Zhang, "Phase-shifting digital holography," *Opt. Lett.* **22**, pp. 1268–1270, 1997.
14. D. A. Huffman, "A method for the construction of minimum redundancy codes," *Proc. IRE* **40**, pp. 1098–1101, 1952.
15. J. Ziv and A. Lempel, "A universal algorithm for sequential data compression," *IEEE Trans.* **IT-23**, pp. 337–343, 1977.
16. J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," *IEEE Trans.* **IT-24**, pp. 530–536, 1978.
17. T. A. Welch, "A technique for high performance data compression," *IEEE Computer* **17**, pp. 8–19, 1984.
18. G. K. Wallace, "The JPEG still picture compression standard," *Commun. ACM* **34**, pp. 30–44, 1991.
19. A. Léger, T. Omachi, and G. K. Wallace, "JPEG still picture compression algorithm," *Opt. Eng.* **30**, pp. 947–954, 1991.
20. B. Javidi, "Nonlinear joint power spectrum based optical correlation," *Appl. Opt.* **28**, pp. 2358–2367, 1989.
21. N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans.* **C-32**, pp. 90–93, 1974.