

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Digital Hologram Coding

Young-Ho Seo, Hyun-Jun Choi and Dong-Wook Kim

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/53904>

1. Introduction

In this section, the overall process to code a digital hologram video and the whole architecture of the service system for digital holograms are explained.

1.1. Overview of holographic signal processing

A digital hologram includes not only the light intensity but also the depth information of the 3D (dimensional) object. That is, both the whole amplitude and the phase information are necessary to reconstruct the original 3D object, which are included in the interference pattern between the reference light and the object light. Since a digital hologram signal has much different characteristics from general 2D signal, a particular data processing technique is required. In spite, both have the same goal to acquire image information from 3D world and to display it as realistic as possible to human visual system (HVS). Thus, if the techniques based on the 2D-image are modified and some proper techniques for 3D image are added, they can be efficient enough to be applied to digital holograms [1, 2].

Fig. 1 shows an overall simplified scheme to process digital holographic signals. A digital hologram video is captured from a moving 3D object by an optical system using CCD (Charge Coupled Device) or by generating with a computer calculation (Computer-Generated Hologram, CGH) [3]. The acquired information (Fresnel field or fringe pattern) shows a noise-like feature. The fringe pattern is converted into a different data type to be applied to a developed image processing tool in *Data Processing* step. The result is encoded with properly designed coding tools in *Video Coding* step, for which this section uses standard techniques of H.264/AVC [4], entropy coder, and a lossless coder [5, 6].

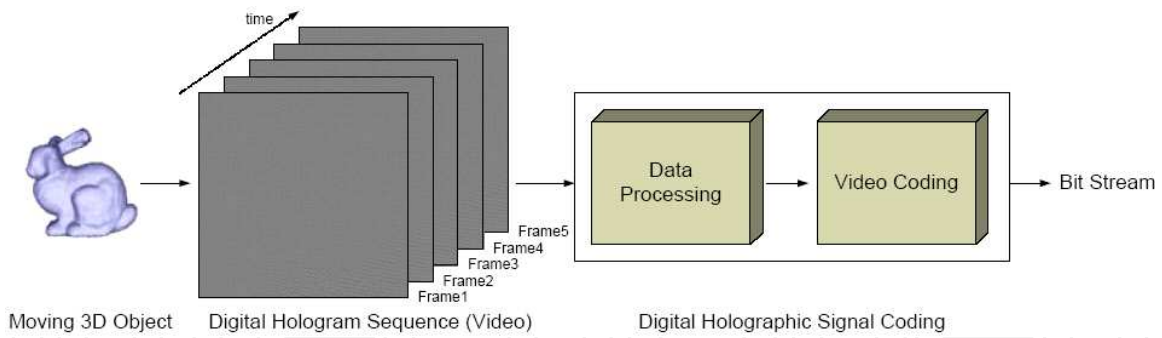


Figure 1. Overall data process scheme for digital hologram videos

1.2. Digital hologram service system

A digital hologram service system can be organized with 3D information acquisition, compression (or encoding), transmission, decompression (or decoding), and holographic display, as shown in Fig. 2. In the data acquisition, 3D information of a real object is acquired as the types of a digital hologram, a depth-and-texture image pair, or a graphic model. A digital hologram is generated by a scanning-based holographic system or components of optical system. A depth-and-texture image is extracted from an advanced imaging system such as the infrared sensor-based depth camera that captures 8-bit depth information for each gray-level pixel. The 3D model is formed by IBMR (Image Based Model Rendering) using multi-view information that is captured by several 2D cameras.

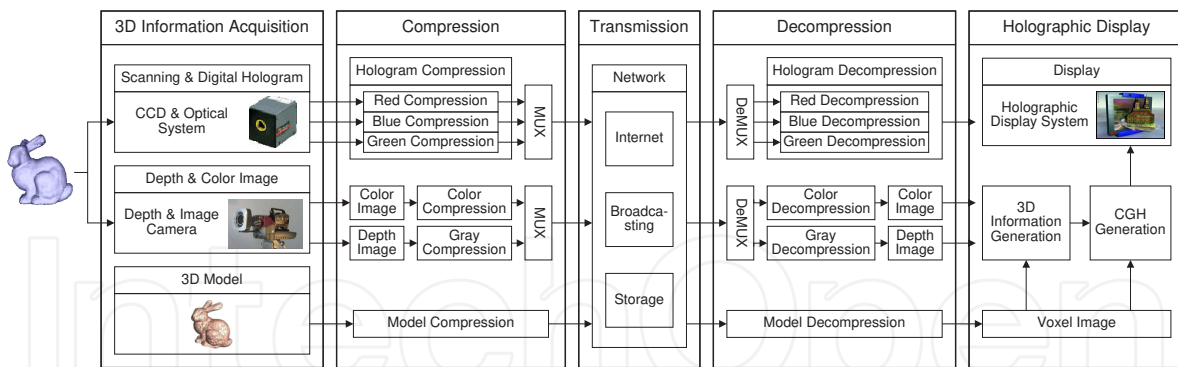


Figure 2. Digital hologram video service system

The acquired 3D information is encoded with various coding algorithms according to the characteristics of the 3D information in the data compression steps. Basically the holographic image is a color image. Some proposed a scheme to process a color hologram without separating the color components [5]. But in this book, each color component (R, G, or B) of a digital hologram is processed separately because each component has different frequency band and different importance level, which are very important factors in our scheme. The encoded data is transmitted through a network and the received data is decoded with the decoding algorithms that are dependent of the encoding process. The decoded data which

retains the digital holographic information such as fringe pattern can be easily displayed in a holographic display system. If the decoded data is a 3D model or a depth-and-texture image, the adequate 3D information to the corresponding display system must be extracted.

2. Characteristics of digital holograms

A digital hologram is a 2D data even though it retains 3D information such as intensity and phase of the interference pattern between object wave and reference wave. Also, a digital hologram is a kind of a transformed result (Fresnel transform, for example) from the spatial domain to a frequency domain. But we regard it as a 2D spatial data throughout this book. In this section, we analyze the characteristics of a digital hologram in a viewpoint of a 2D image to use the techniques for 2D image.

2.1. Sub-sampling

First, we examine the pixel, a basic element of a 2D digital hologram. As a methodology of examination, we took the sub-sampling method as follows: A digital hologram, CGH is divided into fixed-sized blocks, one out of two blocks in both direction is discarded, and the remaining blocks are re-assembled to form a reduced digital hologram whose size are one-fourth of the original one.

Fig. 3 shows the example results of the Rabbit image for various sizes of the sub-sampling blocks. As can see in the figures, a hologram shows totally different characteristics from a natural 2D image. The reconstructed image after sub-sampling one out of two pixels totally loses the original information, as Fig. 3(b). As the size of the sub-sampling segment increases as Fig. 3(c) and (d), the 3D information is getting perfect. Note that the size of the reconstructed image is proportional to the size of the digital hologram. It means that, differently from 2D image, a pixel in a digital hologram retains a piece of inter-dependent information for the whole image, not the localized one. Thus a certain amount of neighboring pixels can reconstruct the original information and the quality is getting better as the number of neighboring pixels increases [6].

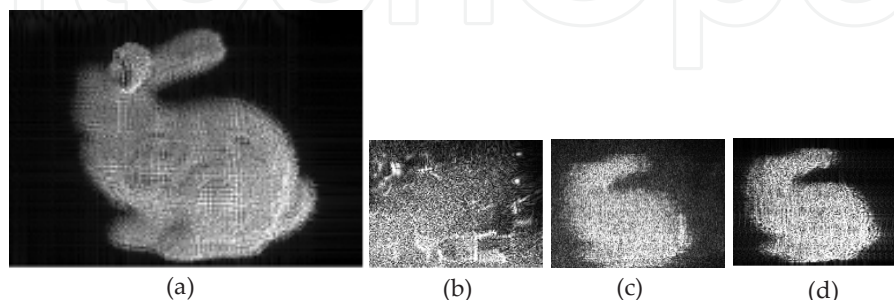


Figure 3. Example of reconstructed images after sub-sampling; (a) original 3D object, by the block size of (b) 1×1 pixel² (one pixel), (c) 32×32 pixel², (d) 128×128 pixel².

2.2. Localization

This inter-dependent property of each pixel can be clearer by taking a part of a digital hologram (cropping) and reconstructing the image. Fig. 4 shows several cropping examples, in which various sizes of co-centered parts of a fringe pattern are cropped out and each of which is converted to reconstruct the image. As in the figures, the reconstructed holographic image is getting larger and clearer as the cropped size increases. This property is similar to the previous result from sub-sampling, which shows entirely different characteristics from the case of 2D image: a local region of a natural 2D image retains only the specified information that is defined by the position of the region, while a local region of a hologram retains the information of the entire image [6].

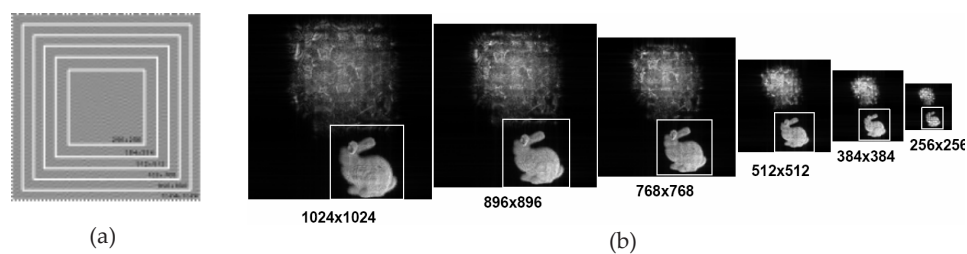


Figure 4. Examples of localization by cropping a digital hologram; (a) cropping scheme for digital hologram, (b) reconstructed images.

This inter-dependent property of a local region in a hologram can also be ascertained by dividing the whole fringe pattern into several segments and reconstructing the holographic image with each part or segment, as shown in Fig. 5, where segmentation into four 512×512 segments and sixteen 256×256 segments are depicted. In these figures, the dotted lines are the vertical and horizontal center lines for each locally reconstructed image to show the difference in the location and shape in each local image.

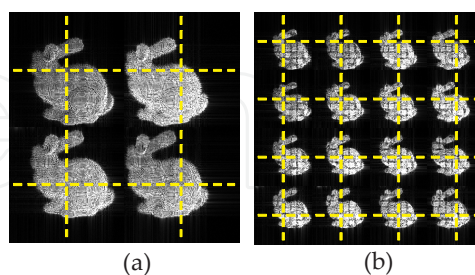


Figure 5. Reconstructed objects in local regions after segmentation with the size of; (a) 512×512 size, (b) 256×256 size.

This property provides a possibility that a digital hologram can be processed as a multi-view image. That is, each of the segmented local regions can be treated as an individual 2D image. Because each of the segments resulting from a fringe pattern has similar information, we can convert each segment into a proper type of data to find correlation among them and efficiently encode them by eliminating the redundancies.

2.3. Frequency characteristic

As shown in Fig. 4, a fringe pattern has noise-like feature and frequency property of it has different tendency from that of a 2D image. Fig. 6 shows the scheme to examine the energy distribution of results from DCT (Discrete Cosine Transform) [7] and DWT (Discrete Wavelet Transform) [8], the two representative frequency-transform methods for 2D natural images, where the whole fringe pattern is 2-dimensionally processed as one processing unit for DCT to retain a consistency in both transform. That is, the regions of frequency bands are matched to compare the energies for each band. In both transforms, the left top band retains the lowest-frequency coefficients and the frequency is getting higher as goes to right and bottom.

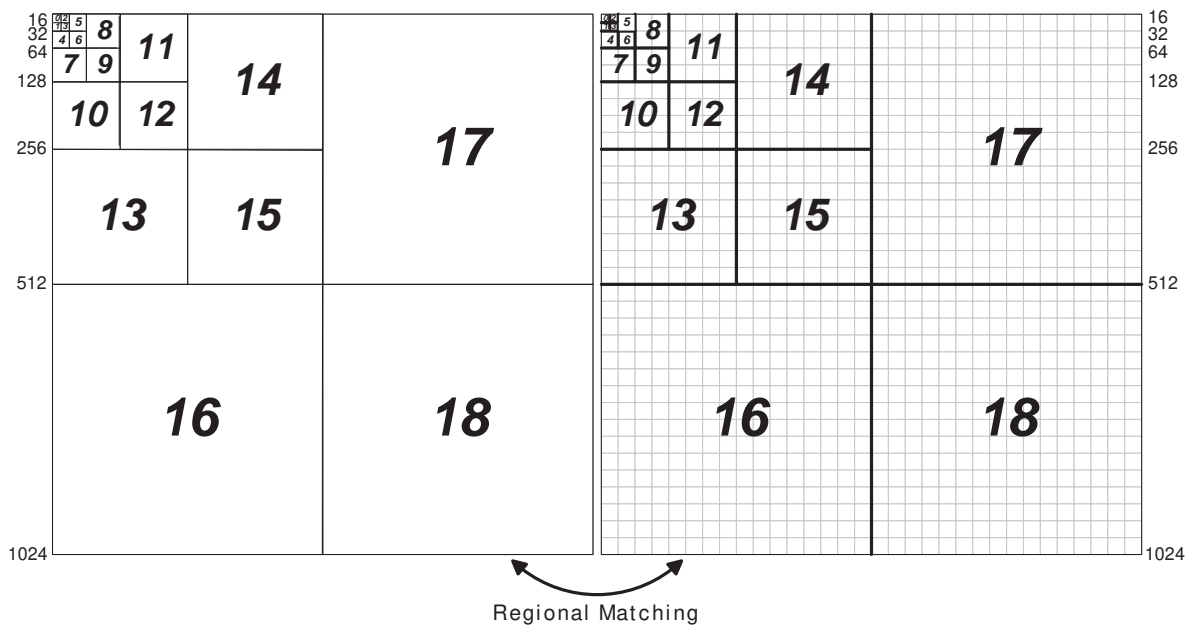


Figure 6. The scheme to examine the coefficient energy for DCT and DWT

The average energies of the coefficients in the frequency bands are depicted in Fig. 7, which was obtained by experimenting various test digital holograms. Note that the energy values of the lowest frequency band are the numbers on the graph, not the quantity the graph scale shows, because they are too large to show in a same graph with others. It is quite similar to a 2D image in that the lowest-frequency coefficient or region has very large energy. The difference in energy distribution for a digital hologram is quite different from that of a natural 2D image is in that some of the high-frequency bands have quite high energy distribution. It implicitly shows that to deal with the frequency-transformed coefficients with an image processing technique for natural 2D images without an additional processing or modification is not efficient for a digital hologram.

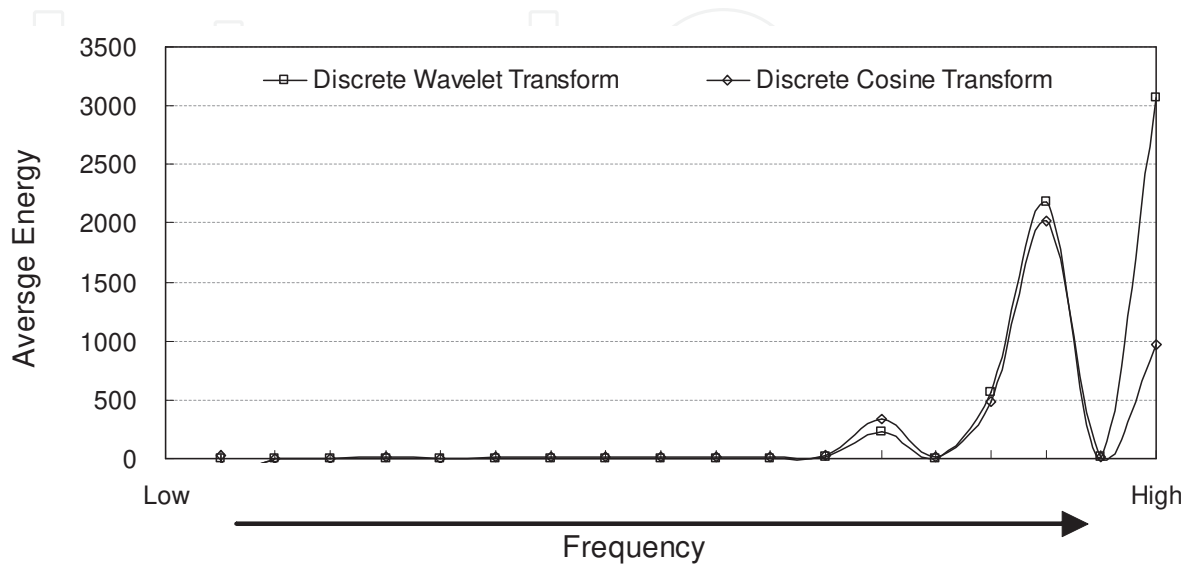


Figure 7. Average coefficient energies at the frequency bands for both DCT and DWT.

3. Lossless compression of a digital hologram

One of the classifications divides the source coding algorithms into lossless coding and lossy coding. During coding processing, a lossless coding method retains perfect information of the original data. But lossy coding method removes a part of it with a permissible limit according to application. In this section, digital hologram is compressed by the following lossless coding methods [9].

3.1. Run length encoding

RLE, or Run Length Encoding, is a very simple method for lossless compression. It encodes how many times each coefficient repeats (a pair of codes with each coefficient and the number of repetitions) [10]. Although it is simple and obviously very inefficient for general purpose compression, it can be very useful at times (it is used in JPEG compression [11], for instance).

3.2. Shannon-Fano coding

Shannon-Fano coding was invented by Claude Shannon (often regarded as the father of information theory) and Robert Fano in 1949 [12]. It is a very good compression method. But since David Huffman improved it later, the original Shannon-Fano coding method has almost never used.

The Shannon-Fano method replaces each symbol with an alternate binary representation, whose length is determined by the probability of the particular symbol such that more common symbol uses less bits. The Shannon-Fano algorithm produces a very compact representation of each symbol that is almost optimal (it approaches optimum when the number of different symbols approaches infinite). However, it does not deal with the ordering or repetition of symbols or sequence of symbols.

3.3. Huffman coding

The Huffman coding method was presented by David A. Huffman, a graduate student of Robert Fano, in 1952. Technically, it is very similar to the Shannon-Fano coder, but it has the nice property of being optimal in the sense that changing any binary code of any symbol will result in a less compact representation [13].

The only real difference between the Huffman coder and the Shannon-Fano coder is the way of building the binary coding tree: in the Shannon-Fano method, the binary tree is built by recursively splitting the histogram into equally weighted halves (i.e. top-down), while in the Huffman method, the tree is built by successively joining the two least weighting nodes until there is only a single node left - the root node (i.e. bottom-up).

Since Huffman coding has the same complexity as Shannon-Fano coding (this also holds for decoding) with always better compression (although only slightly), Huffman coding is almost always used instead of Shannon-Fano coding in reality.

3.4. Rice coding

For data consisting of large words (e.g. 16 or 32 bits) and mostly low data values, Rice coding can be very successful to achieve a good compression ratio [14]. This kind of data is typically audio or highly dynamic range images that have been pre-processed with some kind of prediction (such as delta to neighboring samples). Although Huffman coding should be optimal for this kind of data, it is not a very suitable method due to several reasons (for instance, a 32-bit word size would require a 16 GB histogram buffer to encode the Huffman tree). Therefore a more dynamic approach is more appropriate for data consisted of large words.

The basic idea behind Rice coding is to store as many words as possible with less bits than in the original representation just as with Huffman coding. In fact, one can think of the Rice code as a fixed Huffman code. The coding is very simple: Encode the value X with X '1' bits followed by a '0' bit.

3.5. Lempel-Ziv coding

There are many different variants of the Lempel-Ziv compression scheme. The Basic Compression Library has a fairly straight forward implementation of the LZ77 algorithm (Lempel-Ziv, 1977) [15]. The LZ coder can be used for general purpose compression, and performs exceptionally well for text. It can also be used in combination with the provided RLE and Huffman coders to gain some extra compression in most situations.

The idea behind the Lempel-Ziv compression algorithm is to take the RLE algorithm a few steps further by replacing sequences of bytes with references to previous occurrences of the same sequences. For simplicity, the algorithm can be thought of in terms of string matching. For instance, in written text certain strings tend to occur quite often, and can be represented by pointers to earlier occurrences of the string in the text. The idea is, of course, that pointers or references to strings are shorter than the strings themselves.

The coding results for some example images by above algorithms are shown in table 1.

	Bunny	Duck	Spring	Bramhs
	Ratio (%)	Ratio (%)	Ratio (%)	Ratio (%)
Huffman Coding	85.810	84.816	84.287	91.373
LZ77	99.935	99.862	99.917	100.000
RICE 8-bit	100.000	100.000	100.000	100.000
RLE	99.992	99.986	99.889	100.000
Shannon-Fano	86.332	85.395	84.899	91.717

Table 1. Lossless compression results

4. Lossy compression of a digital hologram

As shown in the previous section, it is identified that lossless coding methods don't give enough compression efficiency. To get more efficient compression for a digital hologram, a lossy coding method can be used. In this section, we try code a digital hologram into a very small amount of bitstream [5, 6].

4.1. Pre-processing

In general, a fringe generated from a 3-D object image has color components. To code it, the fringe pattern should be separated into R, G, B or Y, U, V components. Fig. 8 shows this process. Each of the separated components (RGB or YUV) is coded as an independent channel. Since the YUV format shows more degradation than the RGB format, we decided to use the RGB format. Because the G (green) component shows superior results to the others, we use it to explain the scheme from now on.

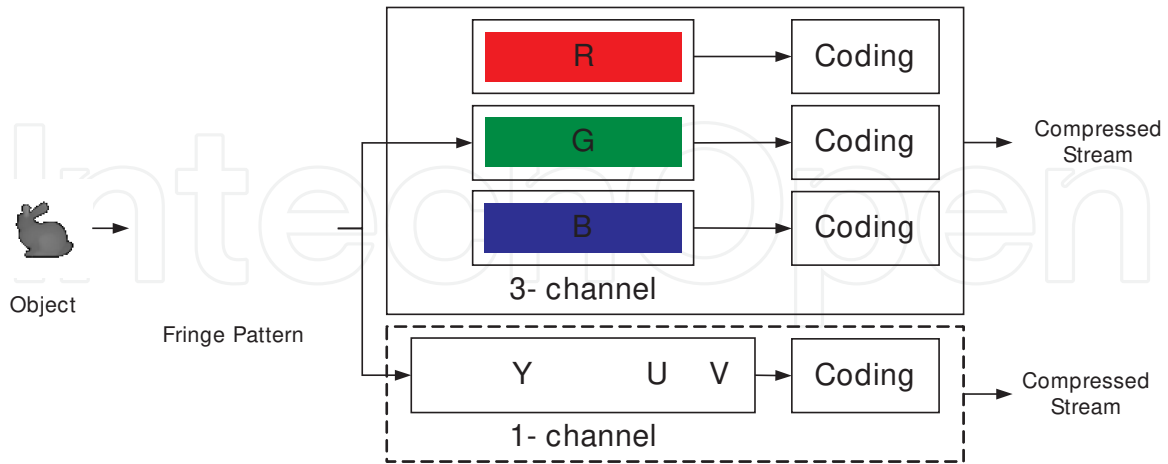


Figure 8. Pre-process to code a digital hologram.

4.2. Data formatting

Each separated component (fringe component) of a fringe pattern is segmented and transformed by 2-D DCT to be applied to a video compression tool.

4.2.1. Segmentation of a digital hologram

The first step in lossy coding for each color component is to divide it into several segments. The size of the segment can be determined arbitrarily but the feature by the different size of the segment is different, which is related to the explanation in section 2. This feature is discussed in next sub-section. Here, we examine various size of segment from 8×8 [pixel²] to 512×512 [pixel²]. The segment size is exactly the same block size of 2D DCT. If the horizontal and vertical size of a fringe component cannot be divided by the segments size, a fringe component is extended with "0" values (zero-padding technique). The segmentation and extension method of a fringe component for the block-based DCT is depicted in Fig. 9.

4.2.2. Frequency transform of each segment

As shown in Fig. 9, each separated segment $I_{m,n}(x,y)$ from a fringe pattern $I(x,y)$ is separately transformed using 2D DCT [7] of Eq. (1), which is for $M \times N$ block size, where C_k is $1/\sqrt{2}$ ($k=0$) or 1 (otherwise).

$$z_{m,n}(x,y) = \frac{1}{4} \sum_{u=0}^M \sum_{v=0}^N C_u C_v I_{m,n}(u,v) \cos[\pi(2x+1)u/16] \cos[\pi(2y+1)v/16] \quad (1)$$

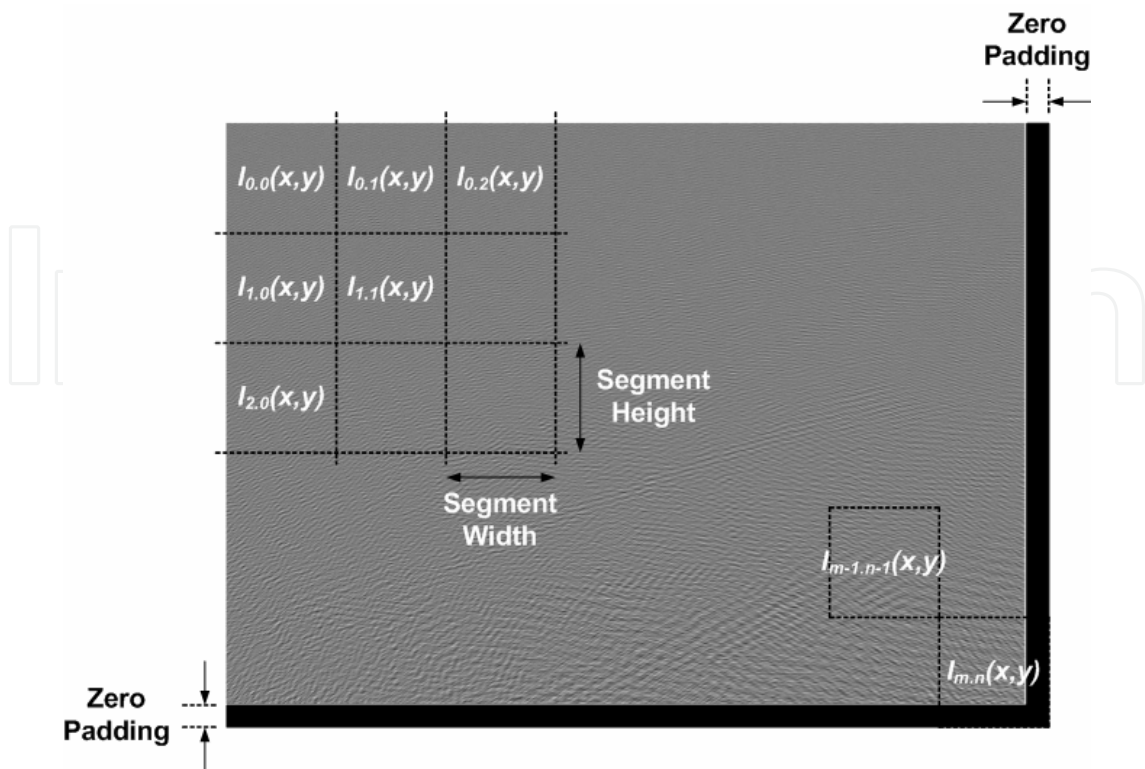


Figure 9. Fringe segmentation and extension for 2D DCT

Fig. 10 shows the results from 2D DCT for the divided segments that have the sizes from 512×512 [pixel²] to 64×64 [pixel²]. A fringe pattern itself is the result of a frequency transform, but we treat it as an ordinary 2D image. Thus, 2D DCT makes double transform for each segment, which results in the somewhat similar shape to the original object for each segment, as can be seen in the figures. In other words, we can treat a fringe segment as a video frame because the local characteristic of the divided fringe component is very similar to the temporal variation characteristics in 2D video. In addition, the differences among the fringe segments are minimal, because the patterns are similar to each other. Therefore, it can be treated as temporal redundancy and compressed efficiently by a coding system for a moving picture [5, 6, 16, 17].

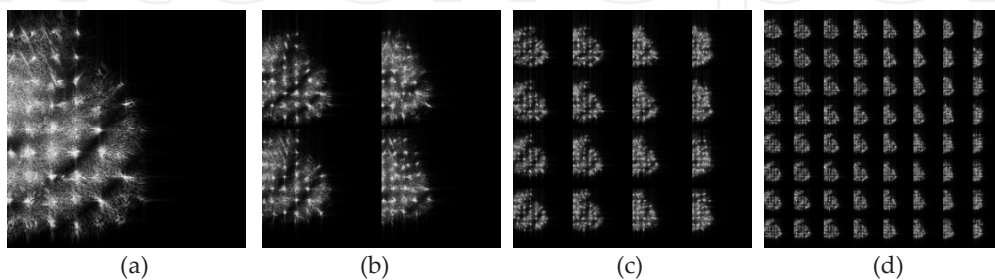


Figure 10. The results from 2D DCT of a fringe with the segment size of (a) 512×512 [pixel²], (b) 256×256 [pixel²], (c) 128×128 [pixel²], and (d) 64×64 [pixel²].

4.2.3. Feature of fringe by segmentation

Fig. 11 shows peak signal-to-noise ratios (PSNRs) of the reconstructed fringe and normalized correlation (NC) of a reconstructed object image after applying only 2D DCT. As shown in Fig. 11(a), PSNR increases as the block size increases, but NC does not show any specific relationship as in Fig. 11(b). From the experiment, the segment size of 64×64 [pixel²] shows best NC values of the reconstructed object images [5].

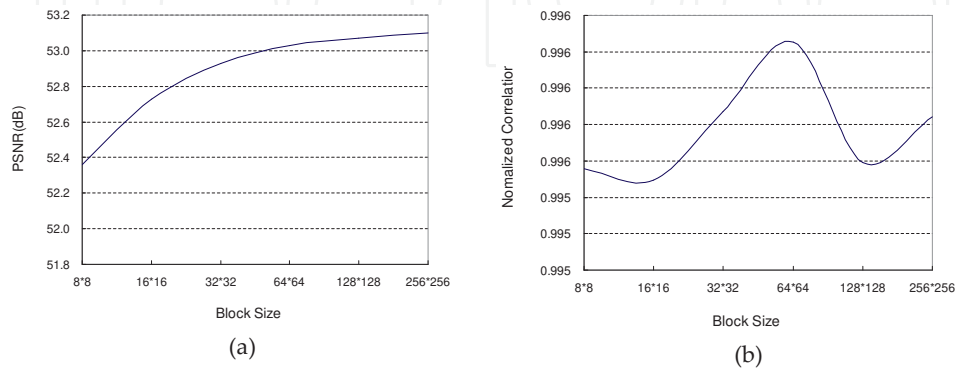


Figure 11. Simulation results for segmentation and DCT; (a) PSNR of the reconstructed fringe, (b) NC of the reconstructed object image.

4.2.4. Sequence formation

As mentioned above, the segmented and 2D DCTed fringe segments can be treated as the video frames with motion. To make a video sequence with those frames, we impose timing information to the segments by ordering them. We call this process as *scanning*. The concept of video streaming including hologram generation is shown in Fig. 12. The object that is projected at a segment or a part of the hologram film (the fringe pattern) has the information of the entire original object with different optic angles. It is similar to the multi-view stereoscopic display method. To use this property in compression coding, a video sequence is generated by segmentation, transformation, and scanning a fringe pattern.

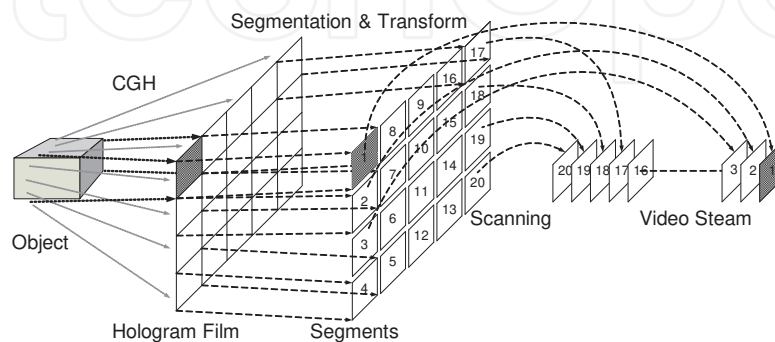


Figure 12. Conceptual description of the scanning

4.3. Segment-based coding

From this sub-section, the lossy coding techniques are explained. First, a scheme based on the segmentation itself is explained, in which a video sequence is formed on the basis of each fringe pattern frame. Many components of this scheme are used in the next coding schemes which will be explained in the next sections [5, 6, 17].

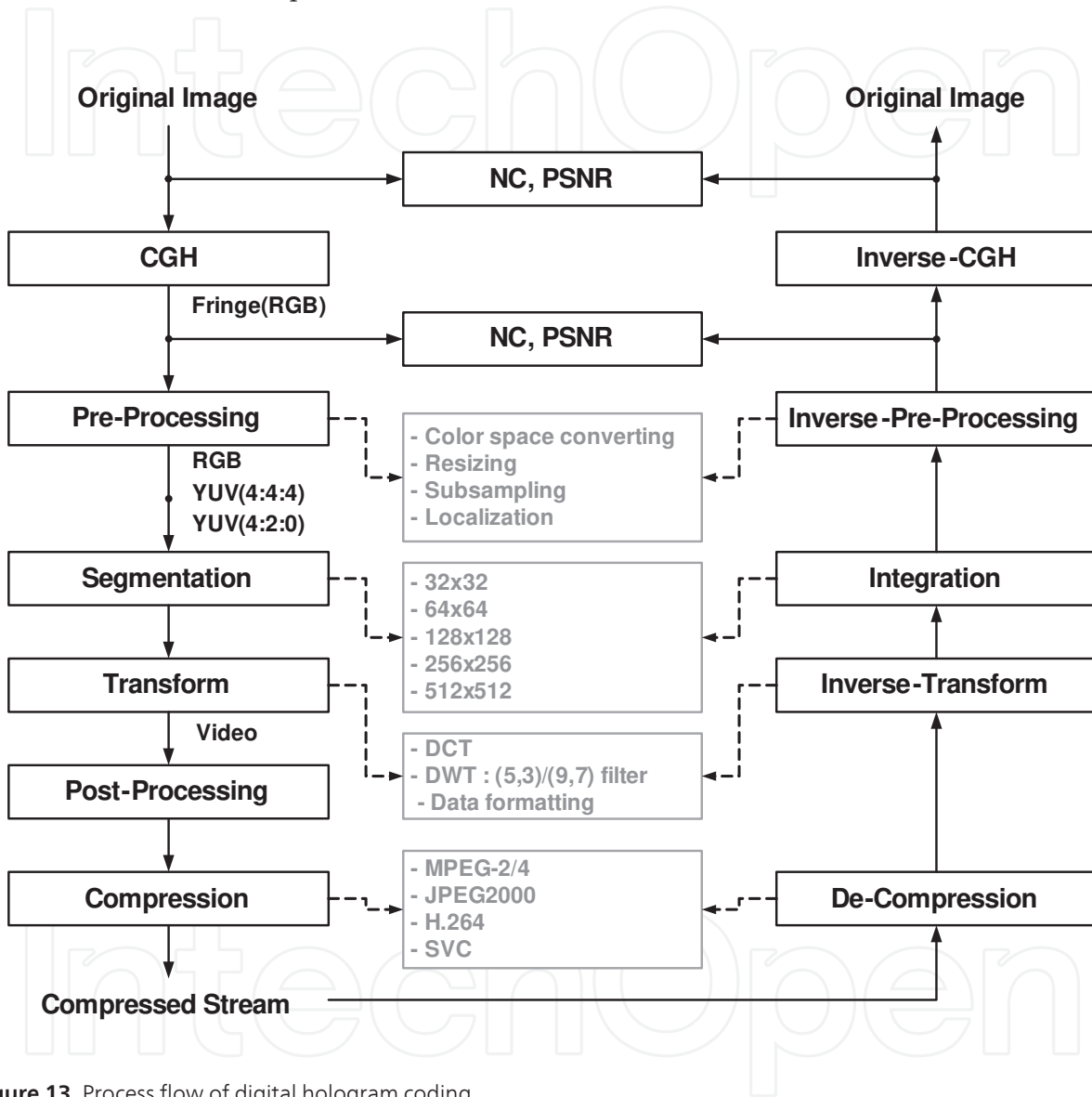


Figure 13. Process flow of digital hologram coding

The first coding scheme is a hybrid coding scheme for hologram compression which includes the proper treatment of the intermediate data. We use the international standard 2D video coding techniques for main compression coding. Thus, it includes the some data manipulation steps to convert the fringe data to fit those standards.

Fig. 13 shows the whole flow to code and decode a digital hologram. It consists of pre-processing, segmentation, transform, post-processing, and compression. During the process, the fringe pattern is divided into several segments (*segmentation*) for 2D DCT (*transform*), as explained

above. The DCT coefficients are rearranged into a video sequence by a data scanning. Therefore, a compression system for 2D moving picture is applied to compress the hologram data.

4.3.1. Scan method

If the segmented fringes are transformed independently by 2D DCT, they have similar property to 2D video frames. Therefore, to apply the 2D video coding tools, they have to be scanned to form a frame sequence in a temporal ordering. Fig. 14 shows the possible scan methods. After 2D DCT, the choice of scan method is related with temporal characteristic and efficiency. That is, each scanning method determines how the object moves, and how much correlation (or redundancy) the two consecutive frames have.

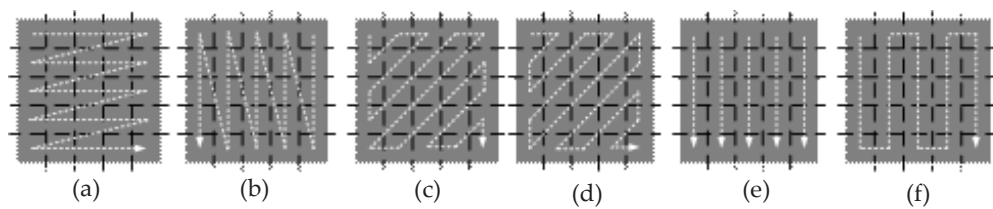


Figure 14. Scanning methods for the segmented fringe images

We analyzed the cost and performance for various scanning methods in Fig. 14 after applying the DS (Diamond Searching)-based ME (Motion Estimation) and MC (Motion Compensation). Two items were compared and the results are shown in Fig. 15 for various sizes of the segmented images. The first item is the number of search points (Fig. 15(a)) which represent the amount of calculation to find the best-matched point. The other is the error value that is the difference between the original image and the ME/MC result, for which the SAD (Sum of Absolute Differences) values are used (Fig. 15(b)).

Both the number of search points and the SAD value decrease as the size of segment increases. In both items, Method f shows the best performance although all the methods showed similar results for the number of search points.

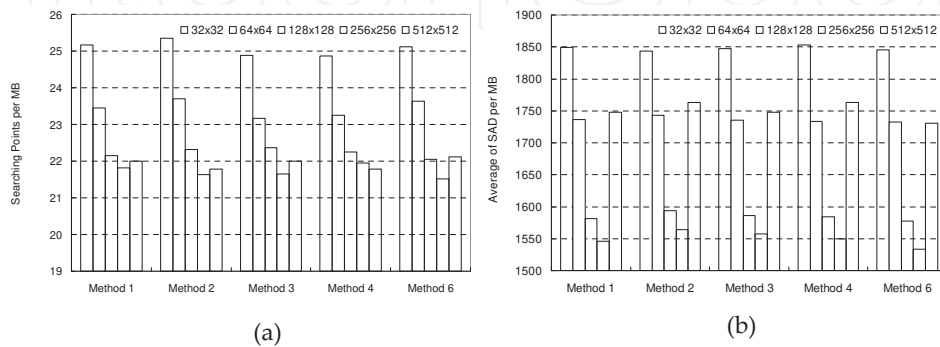


Figure 15. Comparison of (a) searching points in DS, (b) SAD in DS for the scanning methods in Fig. 14

4.3.2. Data renormalization

We classify the transformed data into three kinds according to the characteristics of the data and they are treated differently. The basic classification is shown in Fig. 16. First, the DC coefficients are separated from the other AC coefficients because the DC coefficients have very large energy portion of the DCT block. The DC and several AC coefficients may have the absolute value over 255 and usually they are discarded during quantization. These coefficients have very low possibility to occur (as shown in Table 2, they are less than 0.1%) but their values are so large that they might affect quite amount of image quality. Thus, we treat them especially as will be explained in the next subsection with the name of *exceptional coefficients* (ECfs), while the coefficient with absolute value lower than 256 is named as *normal coefficient* (NCf).

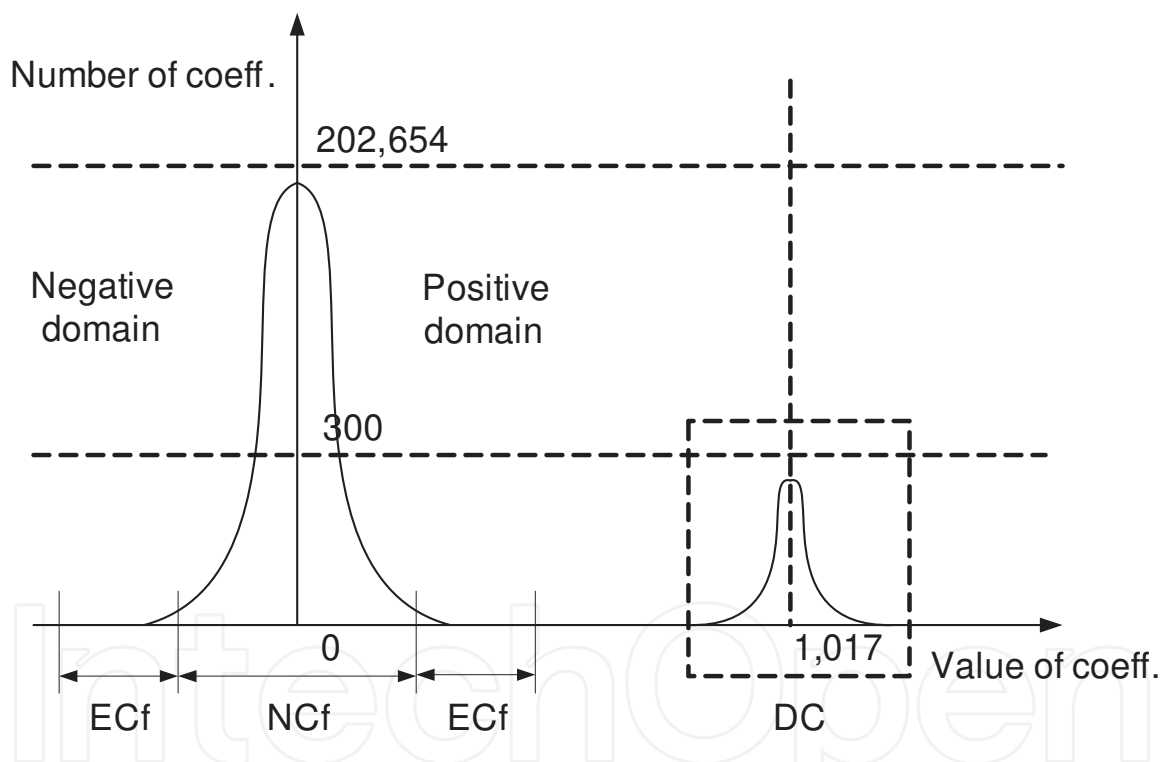


Figure 16. Classification of DCT coefficients

NC can also have a negative value. However negative values are not suitable as the input of general 2D compression tools such as MPEGs. Thus, we change the expression of each coefficient into a signed-magnitude format as shown in Fig. 17. That is, each coefficient has a sign bit and its magnitude. The sign bits are assembled as a bitplane named as *sign bitplane* (SB), in which each sign bit lies in the corresponding position to the original coefficient. As the result, all the coefficients have positive values and a separate SB is produced.

Segment Size	DC Coefficient	AC Coefficient		Sign Bitplane	
		NCfs	ECfs	Plus	Negative
8×8	16,384 (1.5625%)	1,032,192 (98.4375%)	0 (0%)	531,796 (50.7203%)	516,780 (49.2797%)
16×16	4,096 (0.3906%)	1,044,480 (99.6094%)	0 (0%)	526,922 (50.2533%)	521,654 (49.7467%)
32×32	1,024 (0.0976%)	1,047,526 (99.8999%)	26 (0.0024%)	524,825 (50.0521%)	523,751 (49.9479%)
64×64	256 (0.0244%)	104,378 (99.9543%)	223 (0.0213%)	524,926 (50.0652%)	523,650 (49.9348%)
128×128	64 (0.0061%)	1,047,611 (99.9079%)	883 (0.0842%)	524,264 (49.9878%)	524,312 (50.0122%)
256×256	16 (0.0015%)	1,047,677 (99.9143%)	901 (0.0859%)	526,878 (50.2509%)	521,698 (49.7491%)
512×512	4 (0.0004%)	1,047,866 (99.9322%)	706 (0.0673%)	523,328 (49.9085%)	525,248 (50.0926%)

Table 2. Coefficient distribution of the results from 2D DCT

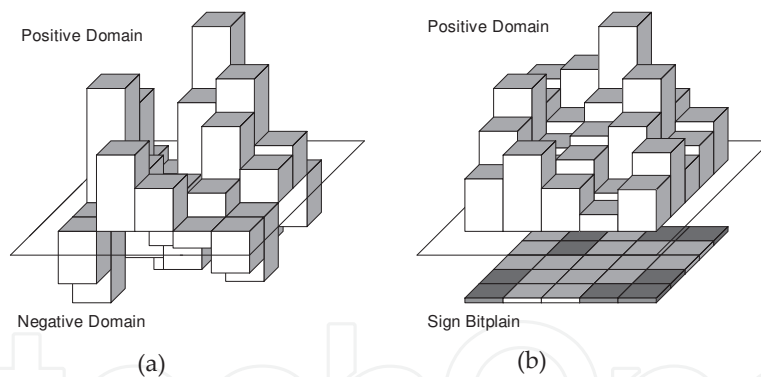


Figure 17. Rearrangement of DCT coefficients; (a) original coefficient plane, (b) signed-magnitude format.

4.3.3. Video coding standard-based compression

As shown in Fig. 18, three compression schemes are involved for compression coding. NCfs of AC coefficients are compressed with an MPEG encoder for 2D video. The MPEG Encoder corresponds to one of MPEG-2 [18], MPEG-4 [19], and H.264/AVC [4]. The ECfs are compressed by DPCM (Differential Pulse Code Modulation) method and the results are applied to an entropy encoder such as Huffman encoder or arithmetic encoder. The SB is very important to recover the image. So it is compressed by a conventional binary compression method such as ZIP [20] etc.

The results of the three compression scheme are assembled to a bit stream to send or store.

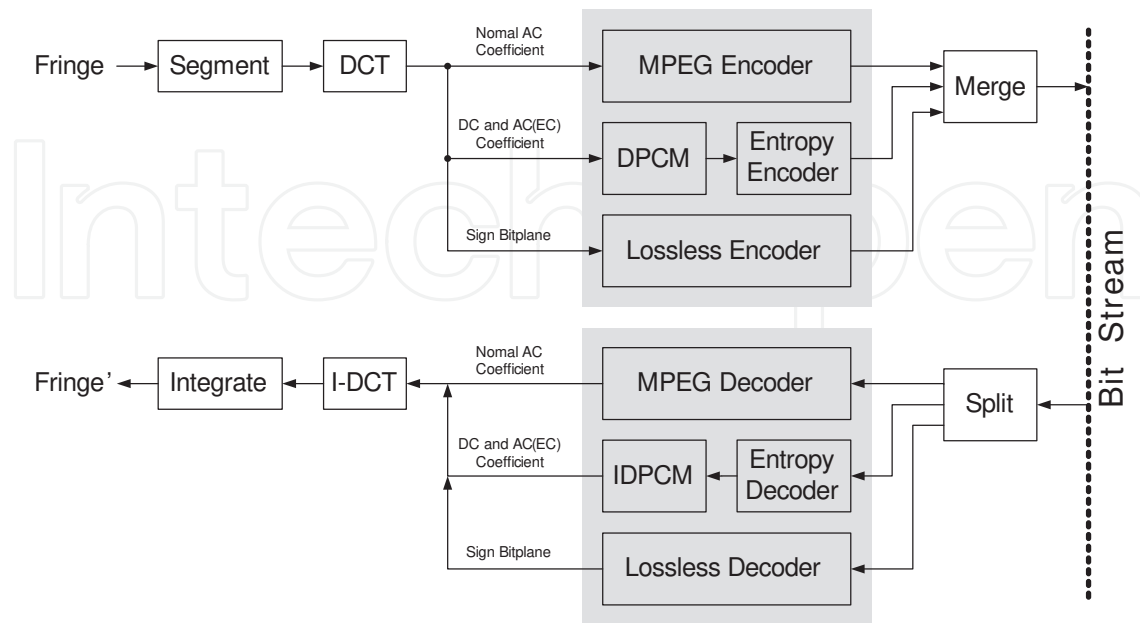


Figure 18. A hybrid coding algorithm for digital hologram compression

4.3.4. Coding characteristics

For the MPEG encoder in the hybrid coding scheme in Fig. 19, we applied MPEG-2, MPEG-4, and H.264/AVC with other tools fixed. The size of segment was from 16×16 [pixel²] to 512×512 [pixel²]. The experimental results from compression and reconstruction are shown in from Fig. 19 to Fig. 31, which are the lossy compression results without lossless compression. Since the reduced amount of data by lossless compression is negligible to lossy compression, it has little influence on the compression ratio.

Fig. 19 and 20 show the example results with Rabbit image after compression and reconstruction by MPEG-4 and H.264, respectively for various compression ratios. In visual examination, there is little performance difference between the results of MPEG-4 and H.264/AVC.

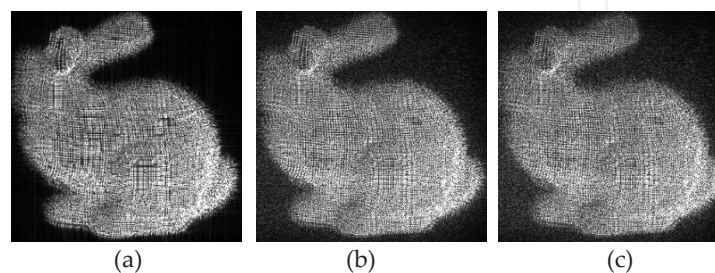


Figure 19. Reconstruction results of object image by MPEG-4 (64×64 segment): (a) original object image, (b) 30:1 compression, and (c) 50:1 compression.

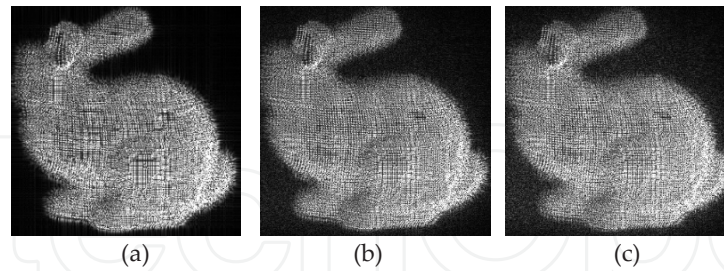


Figure 20. Reconstruction results of object image by H.264/AVC (64×64 segment): (a) original object image, (b) 30:1 compression, and (c) 50:1 compression.

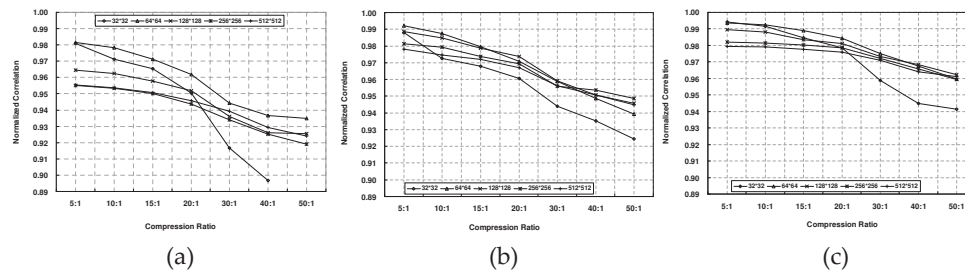


Figure 21. NC values of reconstructed images: (a) MPEG-2, (b) MPEG-4, and (c) AVC.

In Fig. 21, the average image qualities after reconstruction are graphically summarized for the various coding tools as the function of compression ratio. Here, we used the NC value as the measure for the image quality. As seen in Fig. 21, segmented results of the size of 64×64 pixels showed the best quality, as expected by the explanation above. Among the compression tools, H.264/AVC shows the best image quality at the same compression ratio. In the case of H.264/AVC, the NC value of the reconstructed image retained over 0.94, even at the compression ratio of 50:1, except for the segment size of 16×16. Consequently, the scheme shows the best quality when the segment size of 64×64 pixels and H.264/AVC tool are used. In the compression ratio with the same image quality, this scheme shows from four to eight times better performance than the previous schemes whose compression ratios were between 8:1 and 16:1 at 0.94 of NC value.

Fig. 22 and 23 are additional results for other 3-D object images, which are Spring and Duck. The experiment used H.264/AVC for a 64×64 segment. As shown in these figures, the reconstructed images have little difference from the original images in visual inspection, and the values of NC are more than 0.95. Also the reconstructed digital holograms are optically captured and they are shown in Fig. 24.

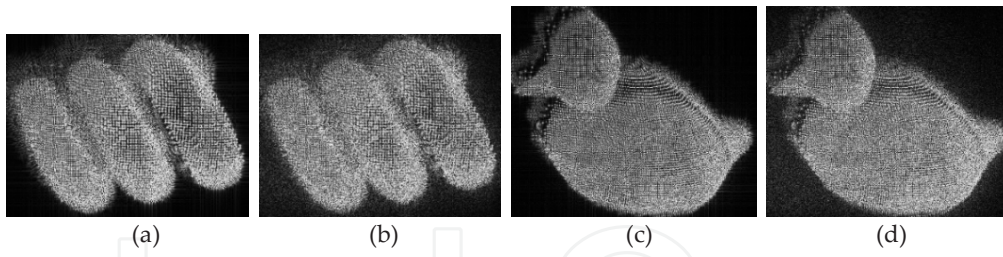


Figure 22. Reconstruction results by H.264/AVC (64x64 segment); (a) original spring image, (b) 50:1 compression, (c) original duck image, and (d) 50:1 compression.

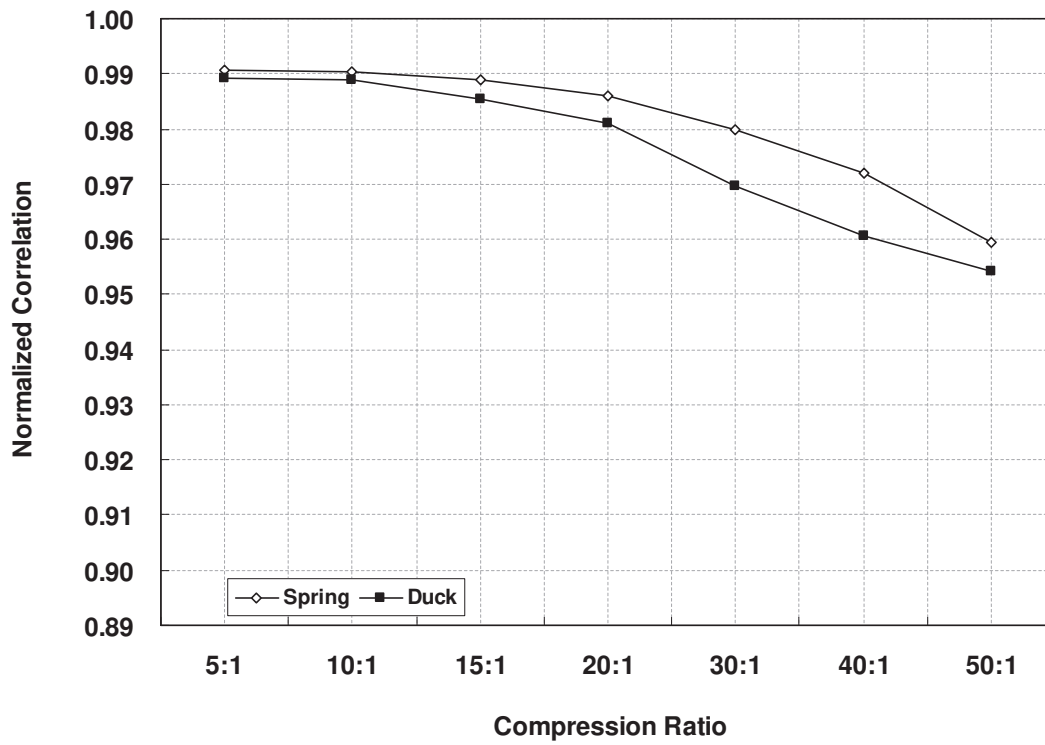


Figure 23. NC values of reconstructed results for the Spring and Duck images in Fig. 22.

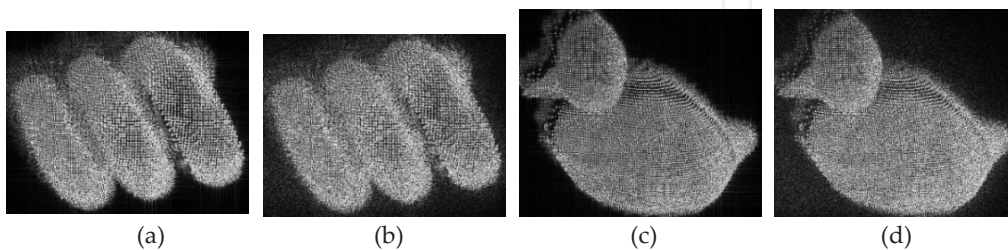


Figure 24. Optically-captured results; (a) original spring image, (b) 50:1 compression, (c) original duck image, and (d) 50:1 compression.

4.4. 3D Scanning-based coding

In this section, we introduce a coding technique based on the properties explained in the previous section. Fig. 25 shows the whole flow to encode a digital hologram, which is for digital hologram videos. It consists of capturing, segmentation, transform, scanning the 3D segments, and lossy/lossless compression. A fringe pattern is divided into several blocks that are defined as segments. After transformed by 2D DCT, the segments are rearranged by 3-dimensional scanning into a video sequence, in which a segment corresponds to a frame. The coefficients of each segment are classified and normalized to fit the standard 2D coding tools. Finally, the video sequence is compressed with tools for moving picture [6].

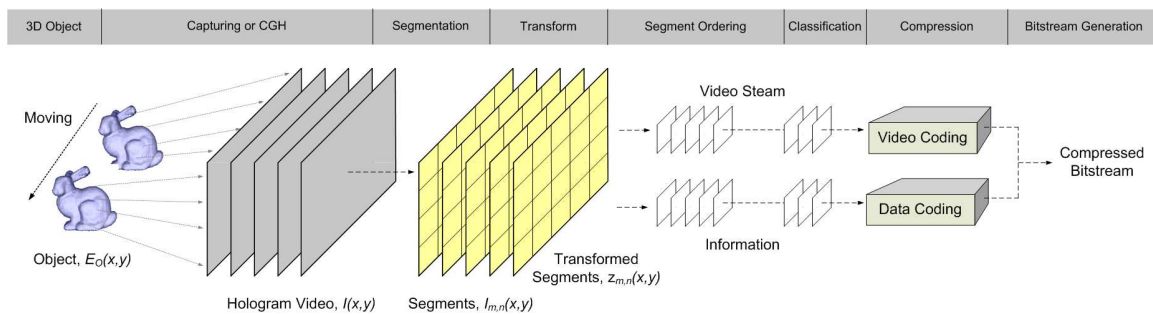


Figure 25. The 3D scanning-based coding procedure for digital hologram videos.

4.4.1. 3D Scanning method

A hologram video consists of several frames of digital holograms and a frame consists of several segments. After rearranged, each segment is treated as a frame to form a new video sequence. Three such sequencing methods with several fringe frames can be considered [6].

- Method 1: it scans all the segments frame by frame, by a predefined order in a frame, to form a sub-sequence (as shown in Fig. 26(a)). The scanning order in a frame is defined to maximize the inter-frame redundancy. This method focuses on the correlation between the segments in the same frame.
- Method 2: it scans segments in the same positions of frames in a GOH in turn (as shown in Fig. 26(b)). The scanning order for the positions is predefined. This method focuses on the correlation for the segments of the same positions in the fringe frames throughout the GOH.
- Method 3: it is a combined form of method 1 and method 2 (shown in Fig. 26(c)) Part of scanning sequence is taken frame by frame.

Method 1 is based on visual similarity between segments in a hologram frame. It is the extended version of coding technique for still object. The scanning order of segments consists of up-scanning and down-scanning in vertical direction to generate a video sub-sequence, which is similar method with the previous section. It connects the frames by the unit of the frame. Meanwhile, in Method 2, the inter-frame connection is drawn by the unit of segment

in the same position. It is based on the similarity between segments in successive frames of GOH. The positions are ordered up and down as Method 1. Method3 is to combine Method 1 with Method. It is based on the similarity between a bundle of segments in the same columns of frames in a GOH.

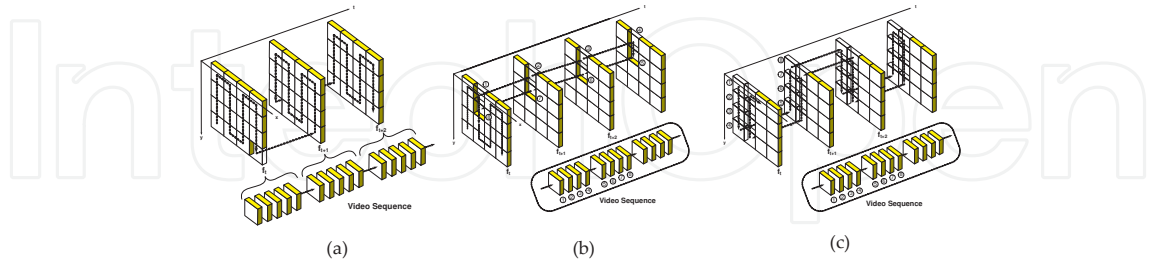


Figure 26. segment scanning methods; (a) Method 1, (b) Method 2, (c) Method 3.

4.4.2. Video coding standard-based compression

Fig. 27 shows a hybrid encoding architecture that depicts the detailed of the compression process in Fig 25. After classification and normalization, NCf, ECf, and SBP are inputted into the compression process. NCfs of AC coefficients are compressed by H.264/AVC, while ECfs are compressed by DPCM (differential pulse code modulation) and the results are applied to an entropy encoder such as Huffman encoder or arithmetic encoder. The SBP is very important to recover the image. So it is compressed by a conventional binary compression method. The results of the three compressed results are assembled to a bit stream to send or store [6].

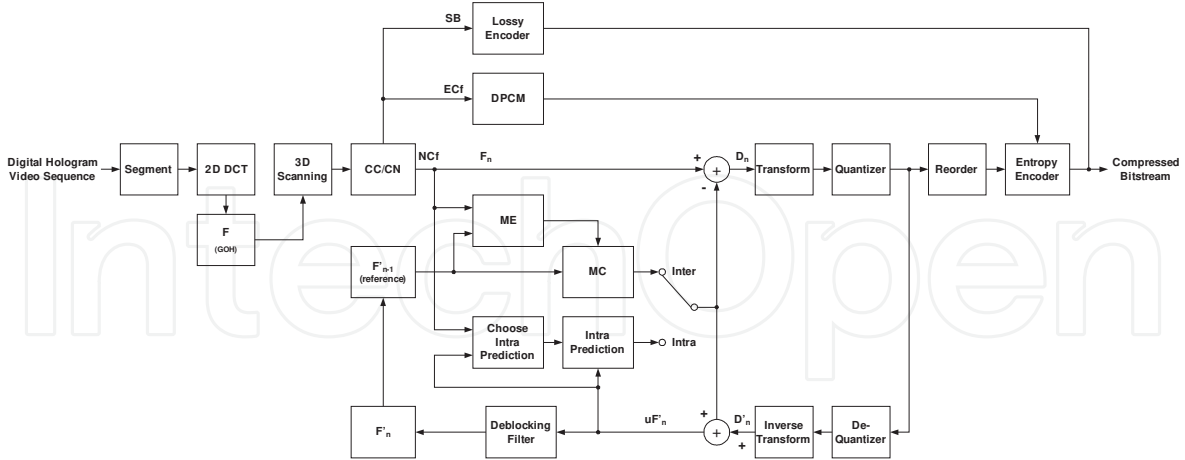


Figure 27. hybrid compression coding scheme for digital hologram.

4.4.3. Coding characteristics

In this section, we introduce the experimental results from applying various digital hologram videos to our scheme and discuss about the compression/reconstruction performance.

The fringe patterns with the size of 1,024×1,024 [pixel²] were used, and the size of segment was 64×64 and 128×128 [pixel²]. The bit rate of the generated video stream compressed by H.264/AVC was adjusted to satisfy the requirement of the amount of data. The options of H.264/AVC used for encoding and decoding are as follow.

- Profile: Baseline (High quality)
- Search range: 16
- Incorporate Hadamard transform
- Reference frames: 5
- Variable macro-block: from 16×16 to 4×4
- Entropy coding: CAVLC
- Bit rate: fixed (10:1~120:1)

4.4.3.1. Still object

First, we applied several still object to our technique. The experimental results are shown in Fig. 28 through Fig. 32, which are the lossy compression results without lossless compression. Since the reduced amount of data by lossless compression is negligible to lossy compression, it has little influence on the compression ratio. In Fig. 28, the average image qualities after reconstruction are graphically summarized for the various images as the function of compression ratio. Here, we used normalized correlation (NC) value as the measure for the image quality. Fig. 29 through Fig. 32 show visual image examples.

As can see from Fig. 28, segmentation results by the size of 64×64 [pixel²] showed the better quality than the size of 128×128 [pixel²]. The NC values of the reconstructed image were over 0.90 even at the compression ratio of 120:1.

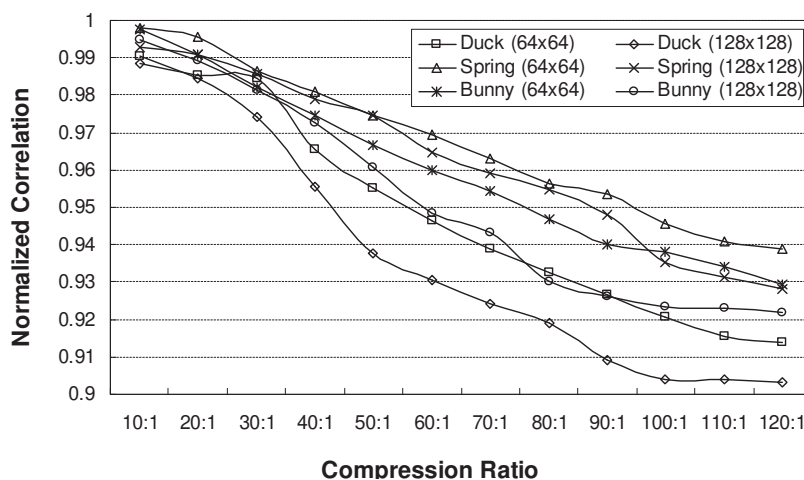


Figure 28. Estimated NC results of reconstructed object images

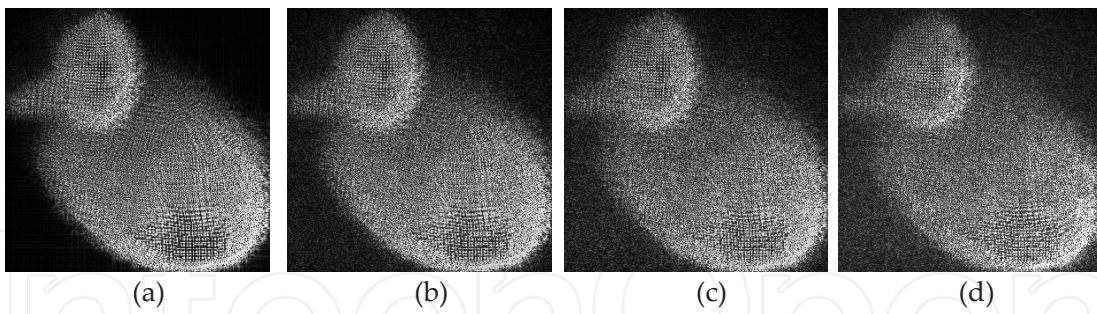


Figure 29. Example of the reconstructed duck object after compression with ratio of (a) 1:1, (b) 40:1, (c) 80:1, (d) 120:1.

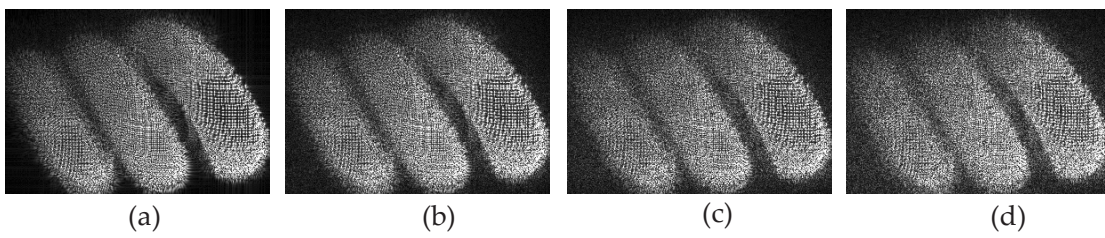


Figure 30. Example of the reconstructed spring object after compression with ratio of (a) 1:1, (b) 40:1, (c) 80:1, (d) 120:1.

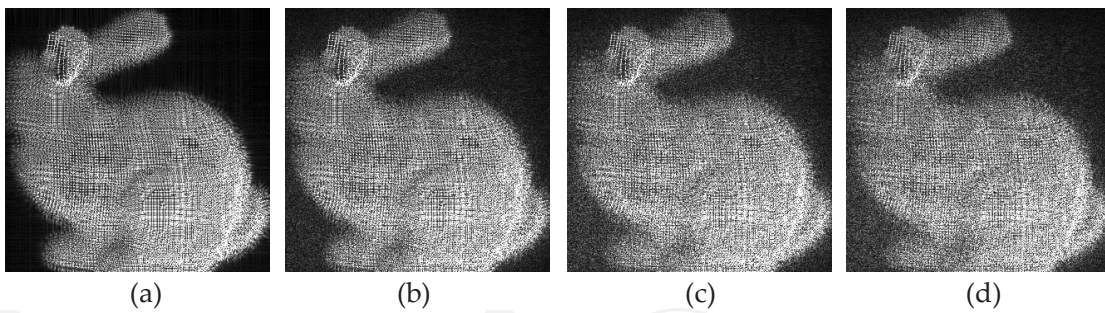


Figure 31. Example of the reconstructed rabbit object after compression with ratio of (a) 1:1, (b) 40:1, (c) 80:1, (d) 120:1 compression

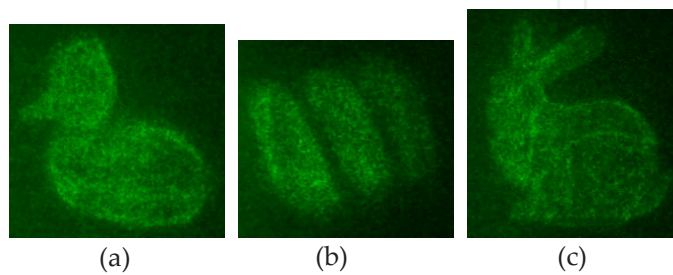


Figure 32. Optically-captured reconstructed objects after compression with 120:1 ratio (a) duck, (b) spring, (c) rabbit

4.4.3.2. Moving object

We compressed two digital hologram videos, which consist of several frames, with the proposed 3D segment-scanning technique. The compression ratio is the same as the case of still hologram. Fig. 33 and Fig. 34 are the NC results, and Fig. 35 is a visual example. The object for Fig. 33 shifts in parallel to 2D axes, and the one for Fig. 34 rotates at the same position. The NC value for the shifting object is better than for the rotating object. For the scanning methods, Method 2 showed the best performance in average, but the differences were not much. Comparing them to the result of the still object, the compression efficiency of our scheme for a moving object is better than a still object.

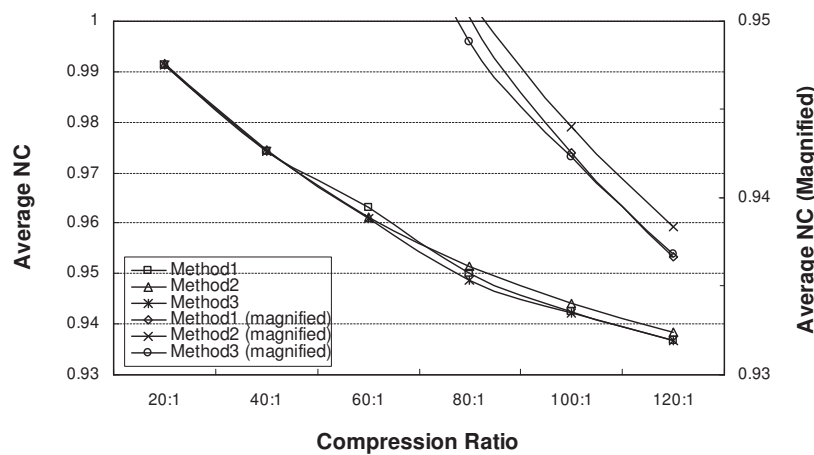


Figure 33. Estimated NC results of the reconstructed video stream for shifting movement

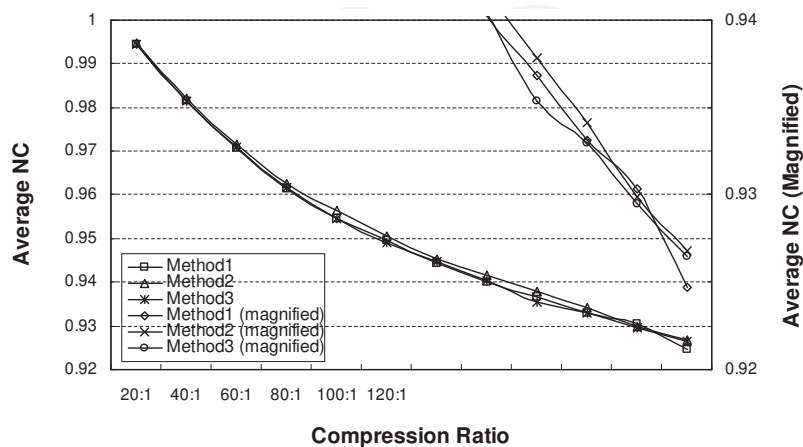


Figure 34. The NC result of the reconstructed video stream for rotating movement

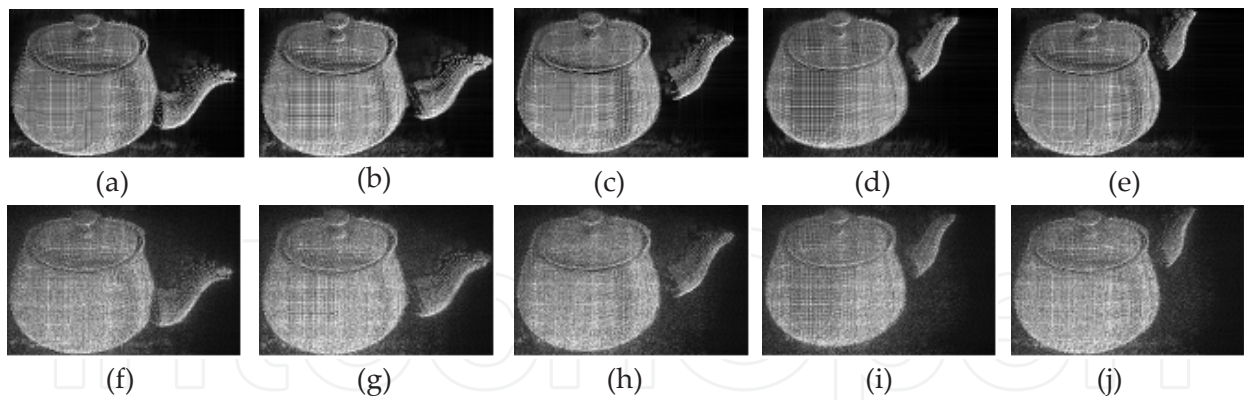


Figure 35. Reconstructed digital hologram video (teapot, 64×64 [pixel²] segment with 120:1 compression), (a) through (e): original frames, (f) through (j): reconstructed frames

4.5. MVC-based coding

In this section, a different kind of coding method for digital holograms is introduced. Fig. 36 shows the coding procedure based on the Multi-View Coding (MVC) technique. It includes various signal processing techniques, prediction techniques, and coding techniques. It is based on the assumption that the local area (segment) in a digital hologram has the similar feature to a view in a 3D multi-view image. As the previous sections, a digital hologram is applied to pre-processing, segmentation, conversion, classification, and normalization in order to deriving correlation between the segments. The difference in the method in this section lies on the multi-view prediction to remove the redundancy among the segments. Thus, this section focuses on this process [21].

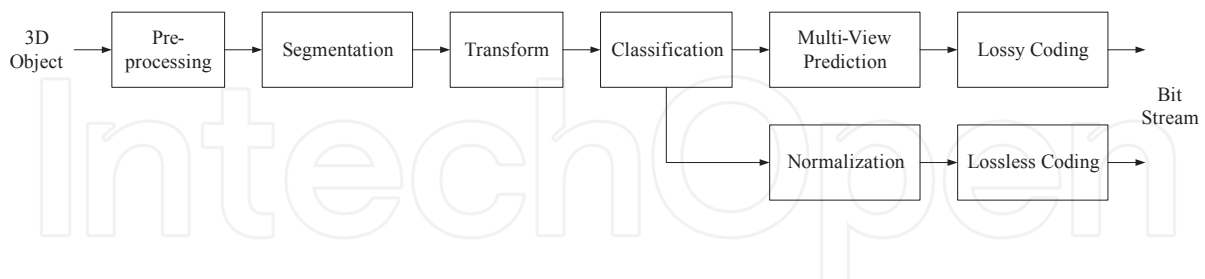


Figure 36. The MVC-based coding scheme.

4.5.1. Multi-view prediction

As explained above, we use a multi-view prediction algorithm to remove the spatial redundancy among the segments and thereby to obtain a high coding performance. Considering the characteristics of a digital hologram, the camera arrangement is limited to $1 \times N$, as an example. The processing order of the multi-view prediction is shown in Fig. 37.

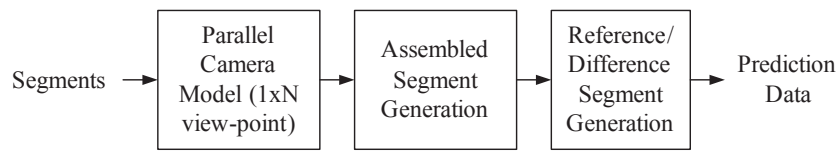


Figure 37. Multi-view prediction scheme

4.5.1.1. Assembled segment

As mentioned above, the transformed segments show visual characteristics similar to images obtained by parallel cameras. For each segment in a segment group (1×N segments), a global disparity is calculated for a base segment in the group with a general matching technique, and an assembled segment (AS) is generated with a method similar to the image mosaic technique. A matching function used in the assembling algorithm to find the global disparity (GD) is,

$$GD_{v+1} = \left\{ X_{r,v+1} \mid \min \sum_i \sum_j \left| I_v(i,j) - I_{v+1}^{T(r,X)}(i,j) \right|, r \in R \right\} \quad (2)$$

where, R represents the area of the AS and r is the area for a segment to be assembled. $I_v(i,j)$ is the coefficient value of a pixel in (i, j) of segment v . The superscripted $T(r,X)$ indicates that the area r is moved by the disparity vector X . In this equation, v is the base segment and $v+1$ is a segment to be assembled to v . The AS generation algorithm described above is shown in Fig. 38. AS is generated for the segments of a column direction and the third segment is defined as the base segment, which generally has the smallest disparity.

Fig. 39 shows an example of AS generation. An AS may become larger than the original resolution of a segment depending on the camera model, and contains all the information about other segments as well as the base segment (the third image in a column in this example). If $N=4$ as the example, the size of AS is about 1.5 times of the original segment.

4.5.1.2. View-point prediction by AS

For general video coding, coding efficiency is improved through motion prediction between images to be coded, using temporally adjacent images as the reference images. In case of multi-view prediction, disparity prediction technique is used to remove redundancy between images in various view-points. Segment at each view-point is coded using the reference segment that is re-separated from AS using MPEG-2.

Fig. 40 shows the scheme to separate the reference segments from AS's with the corresponding GD and to generate difference (residual) segments (DS) using motion estimation and compensation (ME/MC) between the transformed original segments and the reference segments. When the AS is created through global disparity prediction, the values of global disparities are preserved together with the resultant AS. The difference segments are generated by compensating the transformed segments to the predicted segments with the reference segments.

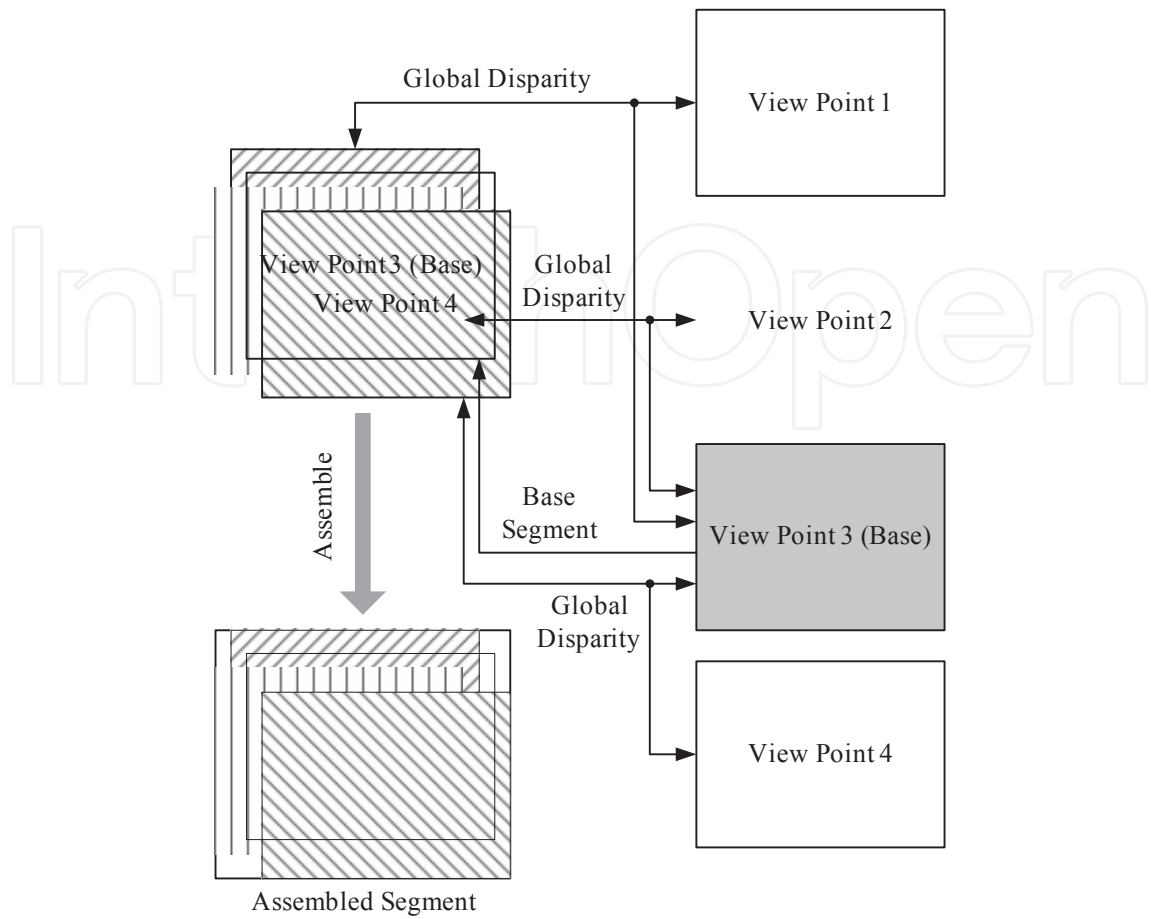


Figure 38. Procedure to generate an AS

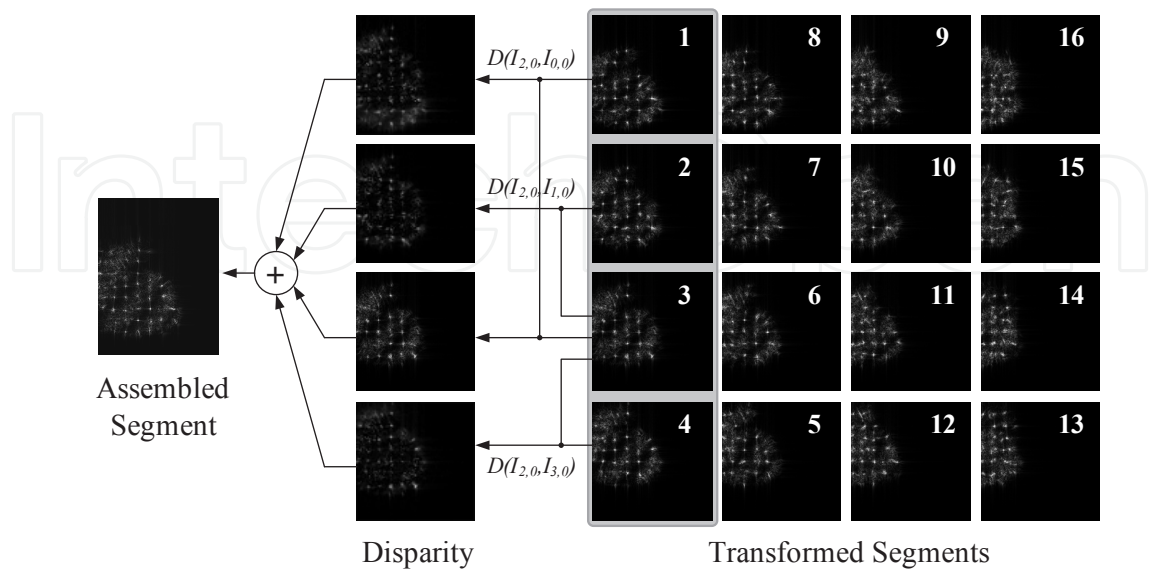


Figure 39. An example of AS generation when fringe pattern (1,024x1,024 pixels) is divided into 4x4 segments

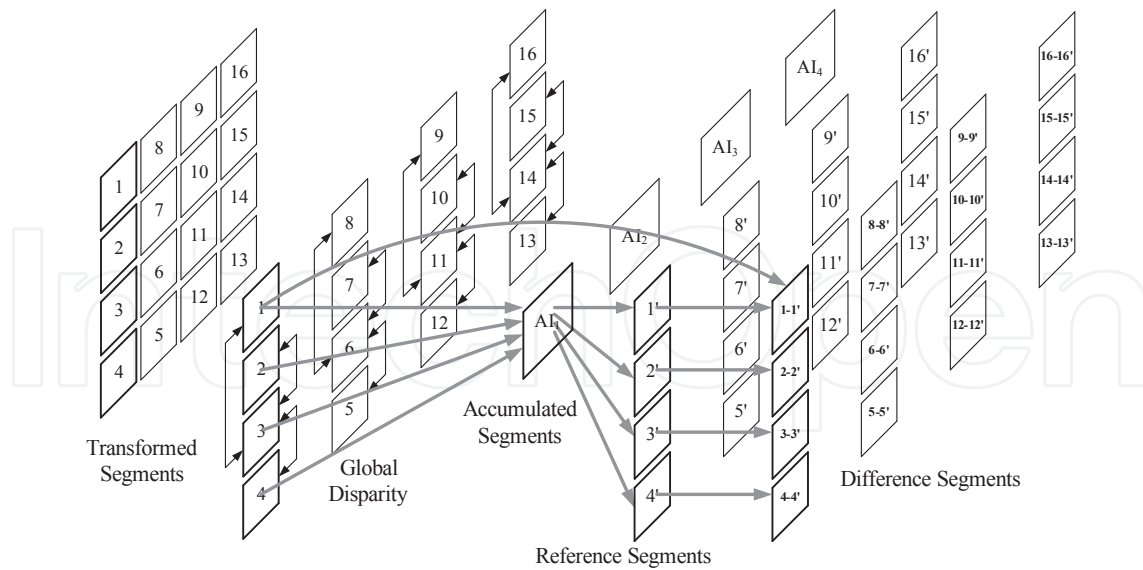


Figure 40. View-point prediction technique to generate AS and DS

4.5.1.3. Temporal prediction

By extending the view-point prediction scheme in the previous sub-section, a temporal prediction technique is applied to a digital holographic video consisting of multiple frames. For each DS generated through the MVP method, prediction in a view is performed in the temporal order. For temporal prediction in each view-point, motion prediction and compensation techniques are used. The scheme is shown in Fig. 41. The detailed are omitted to prevent duplicated explanation.

4.5.2. Video coding standard-based compression

After pre-processing, segmentation, classification and normalization, the segments is converted to DS through GDC (global disparity calculator), ASG (accumulated segment generator), and RSG (reference segment generator) according to the algorithm described above. AS's and DS's are inputted to the corresponding parts of a 2D video coding tool, which is MPEG-2 encoder as an example tool. The modified MPEG-2 encoder with predicting facility for multi-view is shown in Fig. 42. Other information such as, global disparities, motion vectors, and sign bit-planes are encoded by a lossless coding technique.

4.5.3. Coding characteristics

Fig. 43 shows an example of the part of the experiments: (a) is an original 3D object generated by CG, (b) is the hologram reconstructed from the fringe patterns created by the CGH technique, and (c) is an AS produced with the 4 segments in a column direction (when a fringe pattern with 1,024×1,024 in size are segmented into segments with 256×256 in size). Fig. 44 (a) through (d) show the four segments in a column after 2D DCT and Fig. 44(e) through (h) are the corresponding reference segments separated from AS.

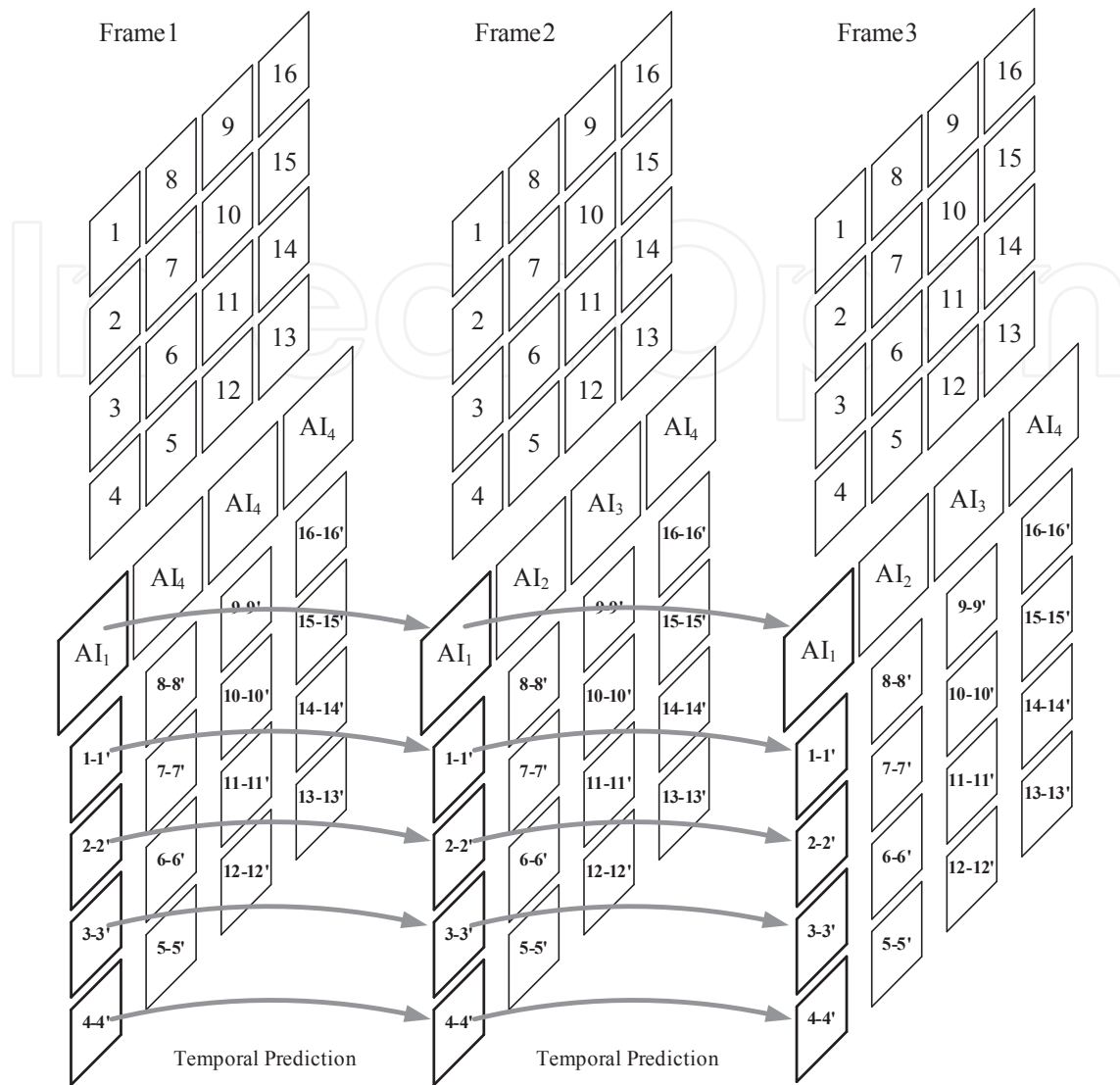


Figure 41. Temporal prediction technique of hologram video

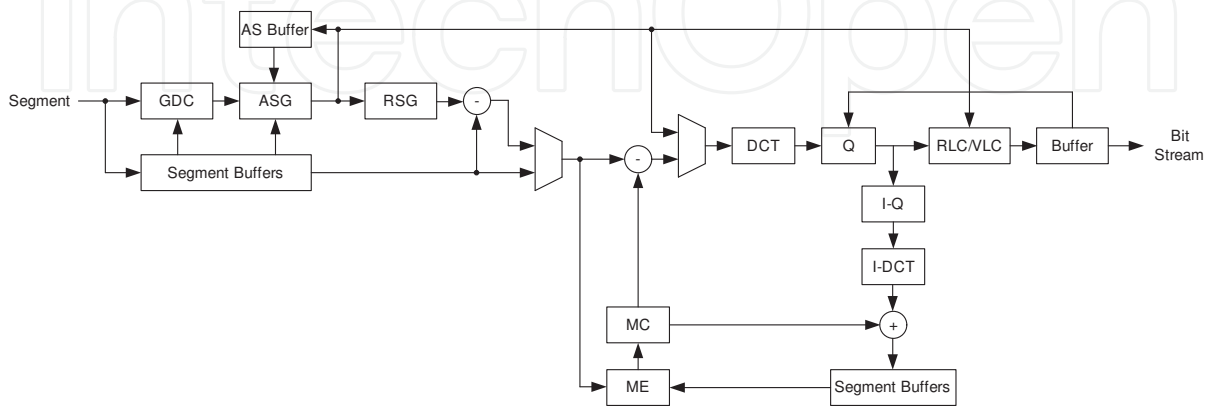


Figure 42. Modified MPEG-2 encoder

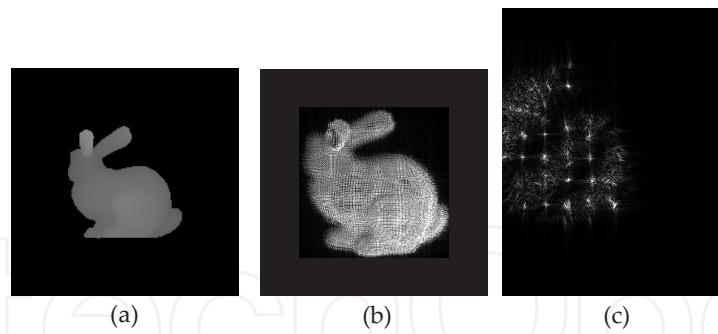


Figure 43. An example of (a) 3D object, (b) reconstructed hologram, (c) accumulated image

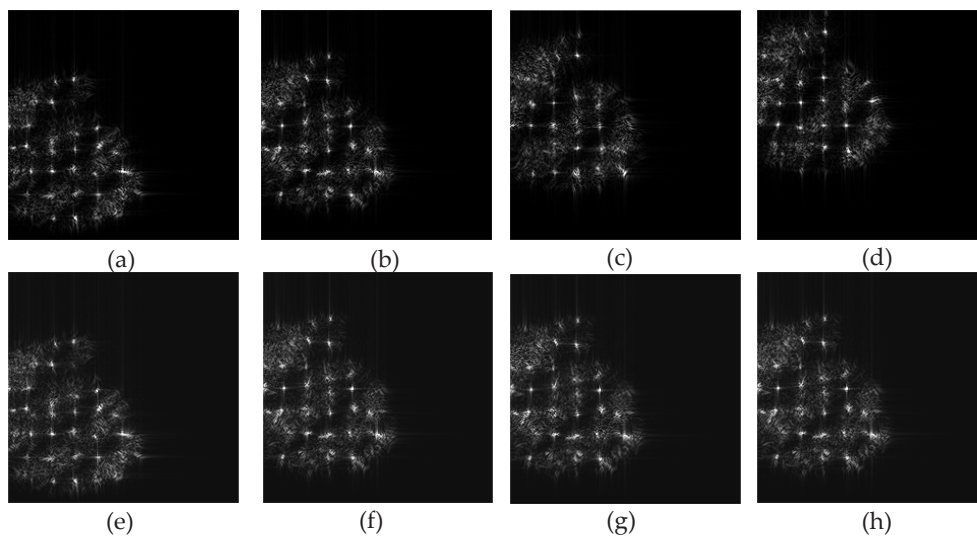


Figure 44. One column of fringe pattern segments after frequency transform (a~d) and the corresponding reference segments (e~h)

Fig. 45. shows the results after applying the multi-view prediction technique based on the AS's. With the proposed method with MPEG-2, the NC value was 0.0349 (approximately 3.6%) higher at a compression rate of 25:1 than the method of section 4.3.

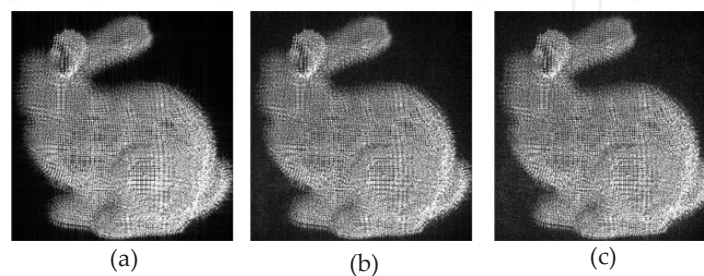


Figure 45. Reconstruction results by the proposed technique: (a) original, (b) 15:1 (NC:0.981842), (c) 25:1 (NC:0.975114)

4.6. Still image coding standard-based compression

Among the still image coding algorithm, the representative methods are JPEG and JPEG2000 standard. Since digital hologram is different from a natural 2D image, although they are used in compressing digital hologram, it is not a good method. In this section, we compress digital hologram using JPEG and JPEG2000 standard, and compare the results with the previous techniques.

4.6.1. JPEG

JPEG (Joint Photographic Experts Group) is the most commonly used still image compression method for photographic images. JPEG has 20:1 compression ratio for a natural image, but we rarely expect its property to be kept because of different visual characteristic of a digital hologram. The property is identified by the coding results, which showed that the maximum compression ratio is 6:1 with NC value of 0.927586. It shows that JPEG is not the powerful tool for digital hologram coding.

4.6.2. JPEG2000

JPEG2000 [22], the new still-image compression standard, was also examined for test, and the results are shown in Figs. 46 and 47. Fig. 46 shows an example of the compression/reconstruction results, which were processed for the color format of RGB. The compression ratio more than 10:1 degrades the image quality quite a lot. Fig. 47 shows the average NC values for each color component as the compression ratio increases.

One can identify that the coding performance has the nearly linear relationship with the algorithm cost. In the case of the compression rate of 20:1, the result of H.264/AVC has more than 0.975 in NC. But the result by JPEG2000 has between 0.73 and 0.83. Also, the object reconstructed by inverse CGH in Fig. 46 has more degradation than the previous results. If we identify the visual difference between Fig. 46 (c) and Fig. 31(d) that are the results from the compression rate of 50:1 and 80:1, Fig. 31(c) shows the clear shape of the object, but Fig. 46(c) is indistinguishable. By comparing the proposed scheme, it shows much better quality than JPEG2000. For reference, a lossless compression that can be reconstructed perfectly shows a compression ratio of about 1.3:1 in average.

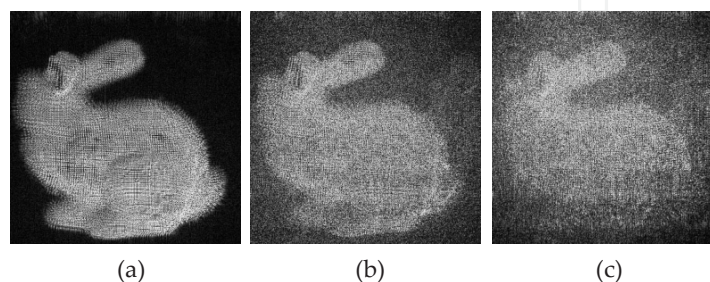


Figure 46. Reconstruction result of object image by JPEG2000 compression (a) Original object image (b) 30:1 (c) 50:1

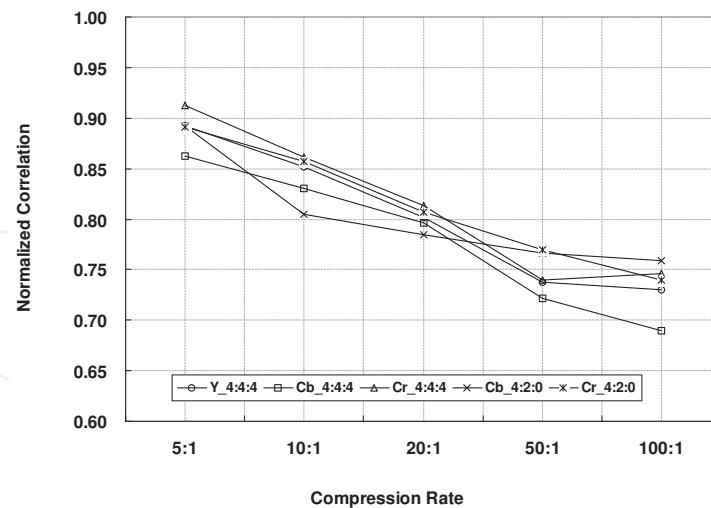


Figure 47. Reconstruction result using JPEG2000

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (MEST). (2010-0026245).

Author details

Young-Ho Seo¹, Hyun-Jun Choi² and Dong-Wook Kim^{3*}

*Address all correspondence to: dwkim@kw.ac.kr

1 College of Liberal Arts, Kwangwoon University, Seoul, Republic of Korea

2 Division of Maritime Electronic & Communication Engineering Mokpo National Maritime University, Mokpo, Republic of Korea

3 Department of Electronic Materials Engineering, Kwangwoon University, Seoul, Republic of Korea

References

- [1] B. Javidi and F. Okano eds, "Three Dimensional Television, Video, and Display Technologies," Springer Verlag Berlin, 2002.

- [2] P. Hariharan, *Basics of Holography*, Cambridge University Press, 2002.
- [3] H. Yoshikawa, "Digital holographic signal processing," Proc. TAO First International Symposium on Three Dimensional Image Communication Technologies, pp. S-4-2, 1993.
- [4] Joint Video Team of ITU-T and ISO/IEC JTC 1. "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264 ISO/IEC 14496-10 AVC)", Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVT-G050, 2003.
- [5] Young-Ho Seo, Hyun-Jun Choi, and Dong-Wook Kim, "Lossy Coding Technique for Digital Holographic Signal", *SPIE Optical Engineering*, Vol. 45, No. 6, pp. 065802-1~065802-10, Jun. 2006.
- [6] Young-Ho Seo, Hyun-Jun Choi, and Dong-Wook Kim, "3D Scanning-based Compression Technique for Digital Hologram Video", *Elsevier Signal Processing - Image Communication*, Vol.22, Issue 2, pp. 144-156. Feb. 2007.
- [7] K. R. Rao and P. Yip, "Discrete cosine transform - algorithms, advantage, applications," New York, Academic Press, 1990.
- [8] R. M. Rao and A. S. Bopardikar, *Wavelet Transforms, Introduction to Theory and Applications*, Addison-Wesley Inc., Reading, MA, 1998.
- [9] T. J. Naughton, Y. Frauel, B. Javidi and E. Tajahuerce, "Compression of digital holograms for three-dimensional object recognition," *SPIE Proc.* Vol 4471, pp. 280-289, 2001.
- [10] http://en.wikipedia.org/wiki/Run-length_encoding
- [11] ISO/IEC 10918-1:1994, "Information technology -- Digital compression and coding of continuous-tone still images: Requirements and guidelines," 1994.
- [12] http://en.wikipedia.org/wiki/Shannon%E2%80%93Fano_coding
- [13] http://en.wikipedia.org/wiki/Huffman_coding
- [14] http://en.wikipedia.org/wiki/Golomb_coding
- [15] http://en.wikipedia.org/wiki/Lempel%E2%80%93Ziv%E2%80%93Markov_chain_algorithm
- [16] H. Yoshikawa and J. tamai, "Holographic image compression by motion picture coding," editor, *SPIE Proc.* vol 2652 *Practical Holography*, pp. 2652-01, 1996.
- [17] Y.-H. Seo, H.-J. Choi, J.-S. Yoo, G.-S. Lee, C.-H. Kim, S.-H. Lee, S.-H. Lee and D.-W. Kim, "Digital hologram compression technique by eliminating spatial correlations based on MCTF", *Optics Communication*, 283, pp. 4261~4270, Jul. 2010
- [18] ISO/IEC 13818-1:2000 "Information technology -- Generic coding of moving pictures and associated audio information: Systems", 2000.

- [19] ISO/IEC 14496-1 MPEG-4 "Coding of Audio -Visual Objects - Part 2 : Visual", Aug. 2002.
- [20] [http://en.wikipedia.org/wiki/Zip_\(file_format\)#cite_note-1](http://en.wikipedia.org/wiki/Zip_(file_format)#cite_note-1)
- [21] Young-Ho Seo, Hyun-Jun Choi, Jin-Woo Bae, Hoon-Chong Kang, Seung-Hyun Lee, Ji-Sang Yoo and Dong-Wook Kim, "A New Coding Technique for Digital Holographic Video using Multi-View Prediction", IEICE Transactions on Information and Systems, Vol.E90-D, No.1, pp.118-125, Jan. 2007.
- [22] JPEG2000 Final Part I: Final Draft International Standard. ISO/IEC FDIS 15444-1, ISO/IEC JTC1/SC29/WG1 N1855 (2000).

