Compression and reconstruction of extremely-high resolution holograms based on hologram-lightfield transforms

Antonin Gilles, Patrick Gioia

Compression and reconstruction of extremely-high resolution holograms based on hologram-lightfield transforms

Antonin Gilles[1]* Patrick Gioia[1,2]

[1] IRT b<>com
Cesson-Sévigné
France

[2] Orange Labs
Rennes
France

Abstract

Holography is often considered as the most promising 3D visualization technique, creating virtual images indistinguishable from the real ones. However, one the main barrier to the adoption of holographic displays in wide 3D viewing systems is the very large amount of information contained in a hologram. Indeed, a hologram with a large size and wide viewing angle contains terabytes of data, urging the need for holographic data coding algorithms. In this paper, we propose a data coding algorithm suitable to the compression of holograms containing several billions of pixels. In our proposed approach, each holographic frame is subdivided into pixel blocks which are 2D Fourier transformed. The pixels thus obtained are rearranged to form new complex-valued segments whose amplitudes have characteristics close to orthographic projection images. These segments are ordered in sequence and their real and imaginary parts are encoded using the High-Efficiency Video Coding (HEVC) Main 4:4:4 coding profile with 4:0:0 chroma sampling.

Keywords : Holographic data coding, Computer-Generated Holography, Digital Holography, 3D Imaging

1 Introduction

Thanks to its ability to create the most natural and realistic depth illusion to the viewer, Holography has gained a significant research interest in the field of three-dimensional (3D) imaging. To display 3D images, a hologram diffracts an illuminating light beam to give it the shape of the light wave that would be emitted, transmitted or reflected by a given scene [1]. Contrarily to most current 3D visualization systems, which are based on Stereoscopy, Holography thus provides all the Human Visual System (HVS) depth cues without causing eye-strain [2]: when looking at a hologram, the viewer perceives the scene as if it were physically present in front of her.

To display holograms, several visualization systems have been investigated in the last few years, including near-eye displays for Augmented Reality (AR) and Virtual Reality (VR) [3, 4], table-top displays [5, 6], Spatial Light Modulators arrays [7, 8] and holographic or wavefront printers [9, 10]. However, one of the main barrier to the adoption of holographic displays in wide 3D viewing systems is the very large amount of information contained in a hologram. Indeed, since it creates the depth illusion thanks to light diffraction, the pixel pitch of a hologram should be close to the wavelength of visible light. Because of this microscopic size, a hologram with a large size and wide viewing angle requires several billions of pixels, leading to terabytes of data. For instance, a hologram of size 20cm × 15cm with a viewing angle of 120° requires a resolution of more than 720K × 540K, yielding more than 1TB per frame. As a consequence, novel compression algorithms able to significantly reduce the memory and bandwidth consumption of holographic signals need to be investigated.

Unfortunately, the fundamental nature of holographic data raises several issues that challenge traditional video coding basics [11]. Indeed, since holograms are 2D diffraction patterns, they have very different signal properties compared to conventional images and videos. In particular, they contain high-frequency coefficients which play a major role in 3D visualization and cannot be discarded as in typical image coding techniques. In light of this, several authors adapted image compression algorithms with alternative transforms to fit the characteristics of holographic data. In [12], Bang *et al.* proposed a novel wavelet-bandelet transform to determine the direction of holographic fringes, achieving higher compression ratios than Discrete Cosine Transform (DCT) and Haar wavelets. In [13], Blinder *et al.* modified the JPEG 2000 codec with two extensions: a fully arbitrary wavelet decomposition and a Directional Adaptive Discrete Wavelet Transform (DA-DWT). Since a significant

*This work has been achieved within the Research and Technology Institute b<>com, dedicated to digital technologies. It has been funded by the French government through the National Research Agency (ANR) Investment referenced ANR-A0-AIRT-07. Authors can be reached at {antonin.gilles, patrick.gioia}@b-com.com.

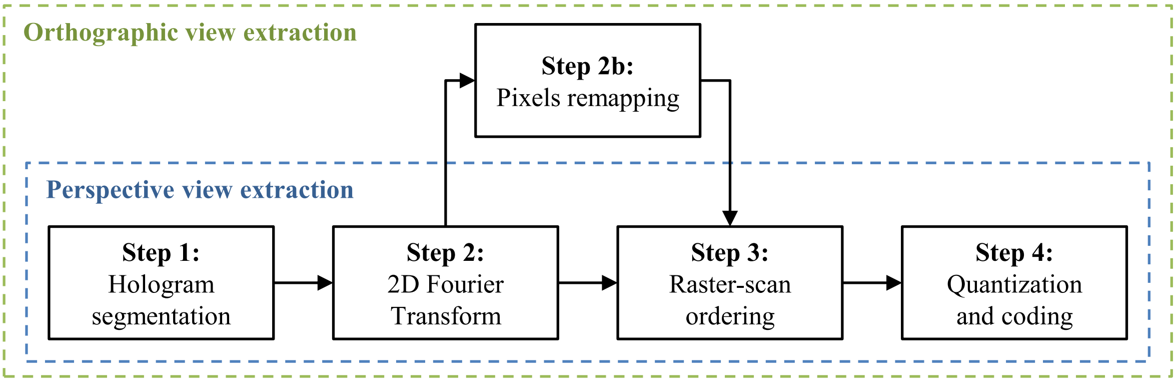


Figure 1: Overall block-diagrams of compression algorithms using perspective and orthographic view extraction methods.

amount of information is contained in the high frequencies, the fully arbitrary wavelet decomposition enables the high-pass sub-bands to be decomposed with more than three levels, allowing full packet decomposition. On the other hand, the DA-DWT uses oriented wavelets to better align with holographic fringes. In [14], Peixeiro *et al.* proposed to extend HEVC with a novel mode depend directional transform trained with holographic data to take into account the directionality of input fringes. In [15, 16], Xing *et al.* proposed separable and non-separable vector lifting schemes to construct wavelets specifically designed for holographic data. Finally, the suitability of wave atoms for hologram compression was demonstrated by Birnbaum *et al.* in [17], outperforming previously mentioned methods for most test cases thanks to their near-optimal space-frequency localization.

While these methods take into account specificities of holographic signals, they do not exploit their 3D visualization properties. Understanding the correlation between the 3D scene content and holographic data could enable better compression ratios and interesting functionalities such as edition, viewpoint and quality scalability, as well as motion estimation and compensation. Aiming at such functionalities, a few works tried to extract relevant 3D visualization features from hologram pixels. In [18], Onural showed that the convolution kernel of Fresnel diffraction forms a family of wavelets whose scale parameter represents the propagation distance. This observation was used by Liebling *et al.* to construct wavelet-like bases called Fresnelets by Fresnel transforming B-spline wavelets [19]. They demonstrated that the Fresnelet transform of a hologram corresponds to its numerical reconstruction at user-defined resolutions. This feature was exploited by Darakis et Soraghan to compress digital holograms using a Set Partitioning In Hierarchical Trees (SPIHT) coding algorithm on the Fresnelet coefficients [20]. More recently, Blinder *et al.* [21] used a piece-wise propagation operator over non-planar surfaces to retrieve the light wave of deep scenes with every part in focus, and encoded it using JPEG 2000. Another approach to retrieve relevant 3D visualization features from a hologram was proposed in [22]. In this paper, El Rhammad *et al.* used an over-complete Gabor wavelets dictionary and Matching Pursuit algorithm to decompose the hologram into a sparse set of light beams and compressed them using an entropy coder. This method was extended to provide viewpoint scalability in [23] and progressive transmission in [24].

Unfortunately, the latter methods have been designed and tested for the compression of holograms whose resolution does not exceed a few hundred thousand pixels, and their computational burden becomes intractable when it comes to very large holograms containing several billions of pixels used in holographic printing applications. To overcome this limitation, we propose a novel and simple holographic data coding algorithm based on a conversion into multi-view data which is suitable to the compression of extremely-high resolution holograms containing several billions of pixels. In our proposed approach, each holographic frame is subdivided into pixel blocks and a 2D discrete Fourier transform is applied to each of them. The pixels thus obtained are rearranged to form new segments with complex values whose amplitudes have characteristics close to orthographic projection images. These segments are ordered in sequence and their real and imaginary parts are encoded using the High-Efficiency Video Coding (HEVC) Main 4:4:4 coding profile with 4:0:0 chroma sampling.

The following of this paper is organized as follows: Section 2 presents the proposed compression method and Section 3 describes the input extremely-high resolution holograms and numerical reconstruction method used for the experiments. Finally, the coding efficiency of our method is analyzed in Section 4.

2 Proposed coding method

In the following, we present two compression methods based on hologram-lightfield transforms. The first method, derived from Seo *et al.* [25], uses a perspective view extraction method, while the second is based on a novel orthographic view extraction method.

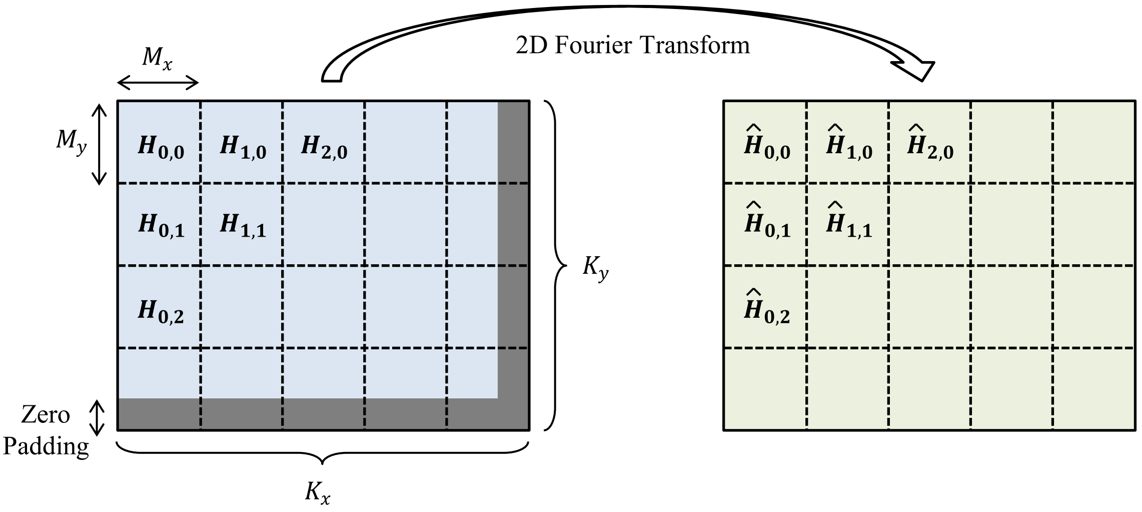


Figure 2: Steps 1 and 2: The hologram is subdivided into rectangular segments and a 2D discrete Fourier transform is applied to each of them.

2.1 Compression using a perspective view extraction method

Figure 1 shows the overall block diagram of the hologram compression using the perspective view extraction method, which comprises four steps. First, the hologram H is subdivided into $(K_x \times K_y)$ rectangular segments $H_{k,l}$ of resolution $(M_x \times M_y)$, such that

$$H_{k,l}(m,n) = H(m + kM_x, n + lM_y), \tag{1}$$

where $k \in \{0, K_x - 1\}$, $l \in \{0, K_y - 1\}$, $m \in \{0, M_x - 1\}$, $n \in \{0, M_y - 1\}$, as shown in Figure 2. If the hologram resolution $(N_x \times N_y)$ is not a multiple of $(M_x \times M_y)$, the blocks of indices $(k, K_y - 1)$ and $(K_x - 1, l)$ are padded with zeros. Then, as illustrated in Figure 2, a discrete 2D Fourier transform is applied to each hologram segment $H_{k,l}$, such that

$$\hat{H}_{k,l}(u,v) = \sum_{m=0}^{M_x-1} \sum_{n=0}^{M_y-1} H_{k,l} \exp\left(-j2\pi\left(\frac{um}{M_x} + \frac{vn}{M_y}\right)\right). \tag{2}$$

The amplitude distributions of Fourier transformed segments $\hat{H}_{k,l}$ have characteristics close to perspective view projections of the scene, and this process can therefore be interpreted as a hologram-lightfield transform. These blocks are then organized in sequence in a raster-scan order and their real and imaginary parts are scaled, quantized and encoded separately using the HEVC Main 4:4:4 coding profile with 4:0:0 chroma sampling, as shown in Figure 3.

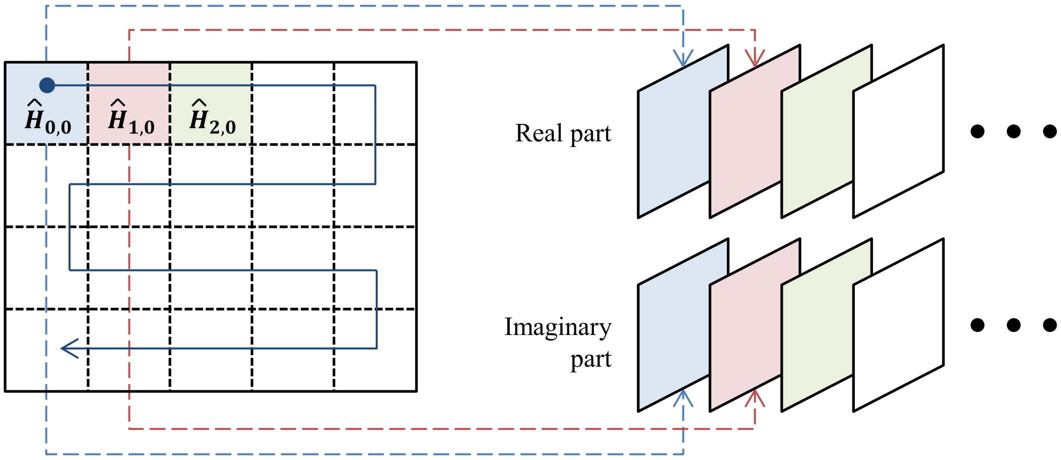
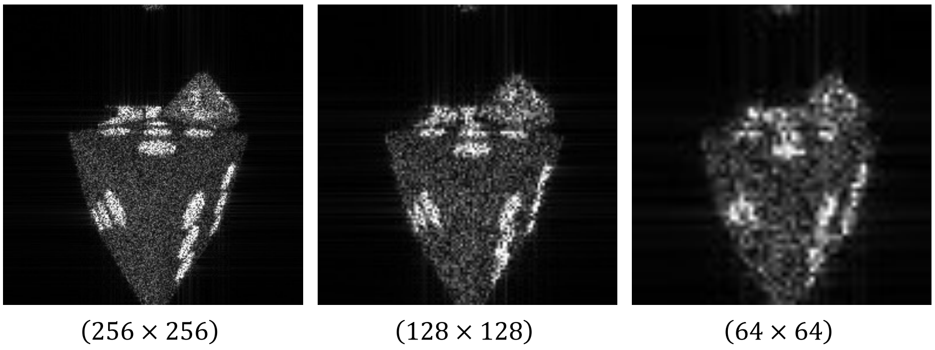


Figure 3: Steps 3 and 4: The hologram segments are ordered in sequence and their real and imaginary parts are encoded using the High-Efficiency Video Coding (HEVC) Main 4:4:4 coding profile with 4:0:0 chroma sampling.



(256 × 256) (128 × 128) (64 × 64)

Figure 4: Amplitude distributions of Fourier transformed blocks using different segment resolutions. The depth of field becomes shallower as the segment resolution increases, and the speckle noise increases when using smaller segment resolutions.

This compression scheme is derived from the method proposed by Seo *et al.* in [25], who used segments of size (64 × 64) and (128 × 128). However, our work differs from [25] with respect to two aspects. First, Seo *et al.* designed their method for amplitude Computer-Generated Holograms (CGH) containing only real and positive values. As a consequence, they applied a DCT instead of the Fourier transform and used data classification to differentiate and encode separately the DC and AC coefficients, corresponding to the zero and ±1 diffraction orders, respectively. Since we use complex-valued CGHs with amplitude and phase modulation, our input holograms do not contain any DC component.

Secondly, in [25], the authors only demonstrated the efficiency of this approach for off-axis Fresnel holograms of resolution (1024 × 1024), providing a rather small viewing angle. In this paper, we evaluate its performance for on-axis holograms containing several billions of pixels and providing a viewing angle of more than 160°. Indeed, when dealing with such holograms, the amplitude distributions of Fourier transformed segments present a large amount of speckle noise, resulting in images which are more complicated to encode using conventional video codecs such as HEVC.

To illustrate this point, we consider a hologram of resolution ($64K \times 64K$) with a pixel pitch of 0.24μm and a wavelength of 532nm. According to the grating equation [1], such a hologram can provide a field of view of 180°. Figure 4 shows the amplitude distributions of Fourier transformed blocks using different segment resolutions. As shown in this figure, the depth of field becomes shallower as the segment resolution increases. Furthermore, when using smaller segment resolutions, the blur decreases at the cost of a higher speckle noise. This trade-off between blur and speckle noise may strongly reduce the efficiency of conventional video codecs. To overcome this limitation, we propose a novel orthographic view extraction method, described in Section 2.2.

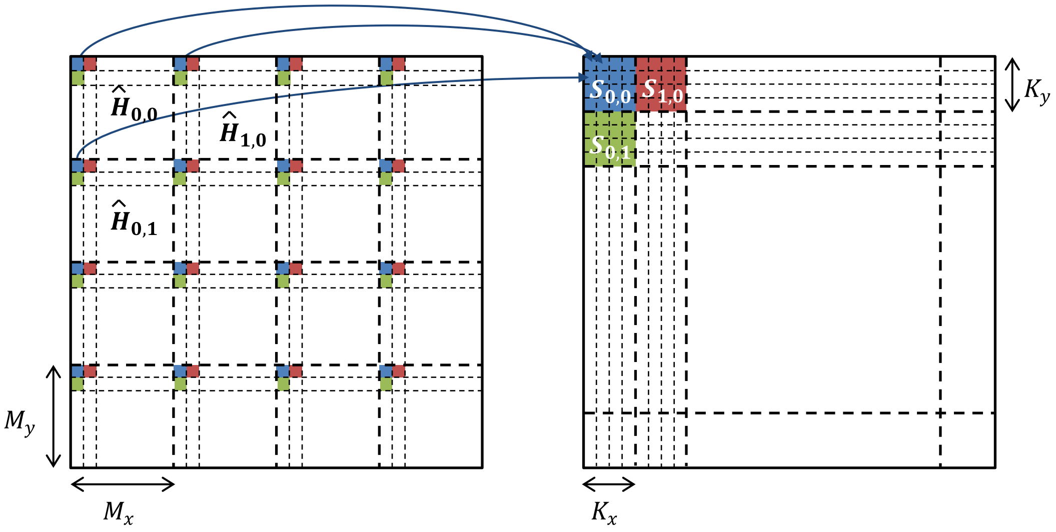
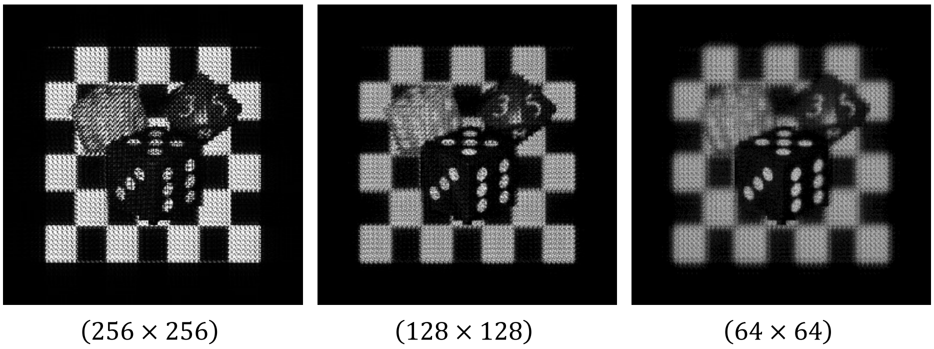


Figure 5: Step 2b: The pixels of each block are rearranged to form new segments with complex values whose amplitudes have characteristics close to orthographic projection images.



(256 × 256) (128 × 128) (64 × 64)

Figure 6: Amplitude distributions of blocks constructed in step 2b using different segment resolutions. The amplitude distributions of these new segments present a very large depth of field with reduced speckle noise, facilitating their encoding using conventional video codecs.

2.2 Compression using an orthographic view extraction method

To convert perspective views images into orthographic projections, we use an additional pixels remapping step between steps 2 and 3, as shown in Figure 1. During this step, a set of $(M_x \times M_y)$ new rectangular segments $S_{m,n}$ of resolution $(K_x \times K_y)$ are constructed by collecting coefficients at the same frequency coordinates in the different Fourier transformed segments $\hat{H}_{k,l}$, such that

$$S_{m,n}(k,l) = \hat{H}_{k,l}(m,n), \tag{3}$$

where $k \in \{0, K_x - 1\}$, $l \in \{0, K_y - 1\}$, $m \in \{0, M_x - 1\}$, $n \in \{0, M_y - 1\}$, as shown in Figure 5.

The Fourier transform of a complex-valued hologram corresponds to its decomposition into plane waves whose propagation vectors are given by the frequency coordinates [1]. Therefore, collecting coefficients at the same frequency coordinates in the Fourier transformed segments enables orthographic or parallel projection views of the scene to be extracted. As shown in Figure 6, the amplitude distributions of these new segments present a very large depth of field with reduced speckle noise, facilitating their encoding using conventional video codecs.

Finally, these new segments $S_{m,n}$ are organized in sequence in a raster-scan order and their real and imaginary parts are scaled, quantized and encoded separately using the HEVC Main 4:4:4 coding profile with 4:0:0 chroma sampling, as shown in Figure 1.

3 Calculation and reconstruction of extremely-high resolution holograms

3.1 Calculation of extremely-high resolution holograms

To evaluate the proposed compression methods, we computed two holograms of resolution $(64K \times 64K)$ using a layered holographic stereogram calculation method [26]. Figure 7 shows the overview of the hologram calculation algorithm, which comprises four steps.

First, the hologram is subdivided into $(L_x \times L_y)$ rectangular segments of resolution $(N'_x \times N'_y)$. Then, for each holographic segment, a 2D+depth perspective projection of the scene is synthesized using a virtual camera located at the segment's center and with a vertical and horizontal field-of-view of $\theta = 2 \arctan\left(\frac{\lambda}{2p}\right)$, where λ is the wavelength of light and p is the hologram pixel pitch. From this 2D+depth projection, the 3D scene geometry is reconstructed as a set of depth layers parallel to the hologram plane with a complex amplitude given by

$$o_{k,l,d}(x,y) = \begin{cases} \sqrt{I_{k,l}(x,y)} \exp\left(j\phi(x,y)\right) & \text{for } D_{k,l}(x,y) = d \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where $I_{k,l}$ and $D_{k,l}$ are the acquired intensity and depth maps of the scene, respectively, and $\phi(x,y) \in [0; 2\pi[$ is the initial phase, set to a random value to render a diffuse scene. Finally, the light waves scattered by depth layers are numerically propagated towards the hologram plane and summed up to get the hologram segment $H_{k,l}$, such that

$$H_{k,l}(x,y) = \sum_{d=0}^{N_z - 1} \mathcal{P}_{z_d}^{Fr} \left\{ o_{k,l,d} \right\} (x,y), \tag{5}$$

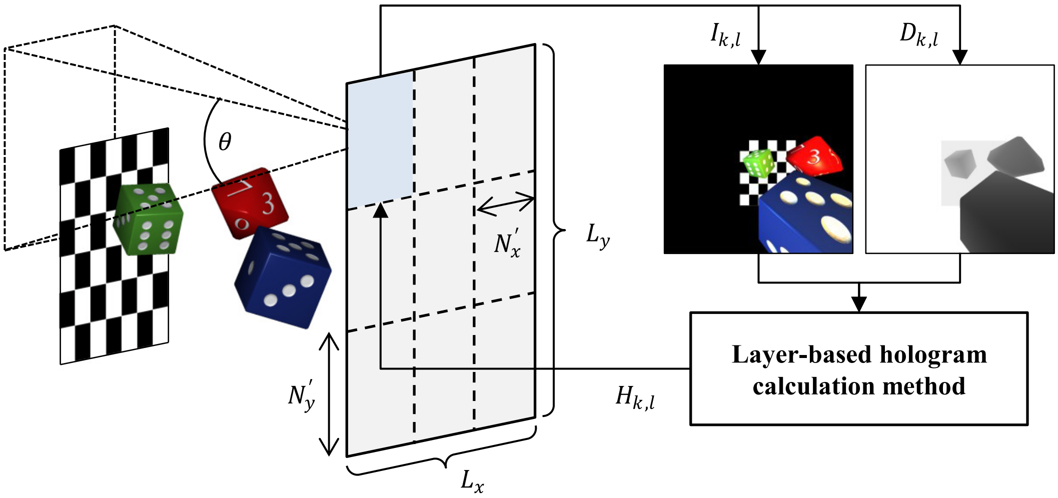


Figure 7: Overview of the hologram calculation method: for each segment $H_{k,l}$, a 2D+depth perspective projection of the scene is rendered and used to compute the corresponding sub-hologram using a layer-based calculation algorithm.

where z_d is the depth of layer d, $\mathcal{P}_z^{Fr}$ stands for the inverse Fresnel transform, given by

$$\mathcal{P}_z^{Fr}\{o\}(x,y) = \frac{j\lambda z}{e^{j\frac{2\pi}{\lambda}z}} \exp\left(j\frac{\pi}{\lambda z}(x^2+y^2)\right) \mathcal{F}^{-1}\left\{o(\xi,\eta)\exp\left(j\frac{\pi}{\lambda z}(\xi^2+\eta^2)\right)\right\}, \tag{6}$$

and $\mathcal{F}^{-1}$ is the inverse Fourier transform. This process is repeated for each hologram segment $H_{k,l}$.

3.2 Numerical reconstruction by light beams extraction

To assess and compare the coding efficiency of compression algorithms using perspective and orthographic view extraction methods, we need a reconstruction process able to simulate the visualisation of the scene from the observer's viewpoint. Such a reconstruction is conventionally performed using a numerical propagation formula such as the Angular Spectrum method, given by

$$\mathcal{P}_z^{AS}\{H\}(x,y) = \mathcal{F}^{-1}\left\{\mathcal{F}\{H\}\mathrm{e}^{j2\pi z\sqrt{\lambda^{-2}-f_x^2-f_y^2}}\right\}(x,y), \tag{7}$$

where f_x and f_y are the Fourier transform spatial frequencies coordinates. Nevertheless, to compute this formula, the sampling window of the input field (in this case, the hologram) needs to be doubled with zero-padding to avoid aliasing in the reconstructed image. As a consequence, given the extremely-high resolution of the hologram, using the Angular Spectrum method would require more than 128GB of available memory, together with a very large calculation time.

In [27], the authors proposed a shifted Angular Spectrum diffraction method to compute the propagation of large wave fields from one plane to another. The procedure is shown in Figure 8(a). In this method, the input and output windows are subdivided into $(L_x \times L_y)$ rectangular segments. For each output window segment (k,l), the light waves scattered by every block in the input window are propagated and summed to obtain the light wave $O_{k,l}$, such that

$$O_{k,l}(x,y) = \sum_{m=0}^{L_x-1}\sum_{n=0}^{L_y-1} \mathcal{P}_z^{AS}\{H_{m,n}\}(x,y). \tag{8}$$

Thanks to the sub-division into segments, this method enables the memory usage to be drastically reduced. However, since it requires $2(L_xL_y)^2$ Fourier transforms to be computed, this method has a significantly larger computational complexity than the conventional Angular Spectrum propagation, resulting in prohibitive calculation times.

Another approach for the propagation of extremely-high resolution holograms over long distances was proposed in [28]. In this paper, the authors took advantage of the separability of the Fourier transform to compute the Angular Spectrum propagation by dividing the hologram into strips which are processed independently. While having a much lower computational complexity than the shifted Angular Spectrum diffraction method, this approach still involves many reads and writes to the disk to transpose data between strip-wise Fourier transforms when the full hologram does not fit into the main memory.

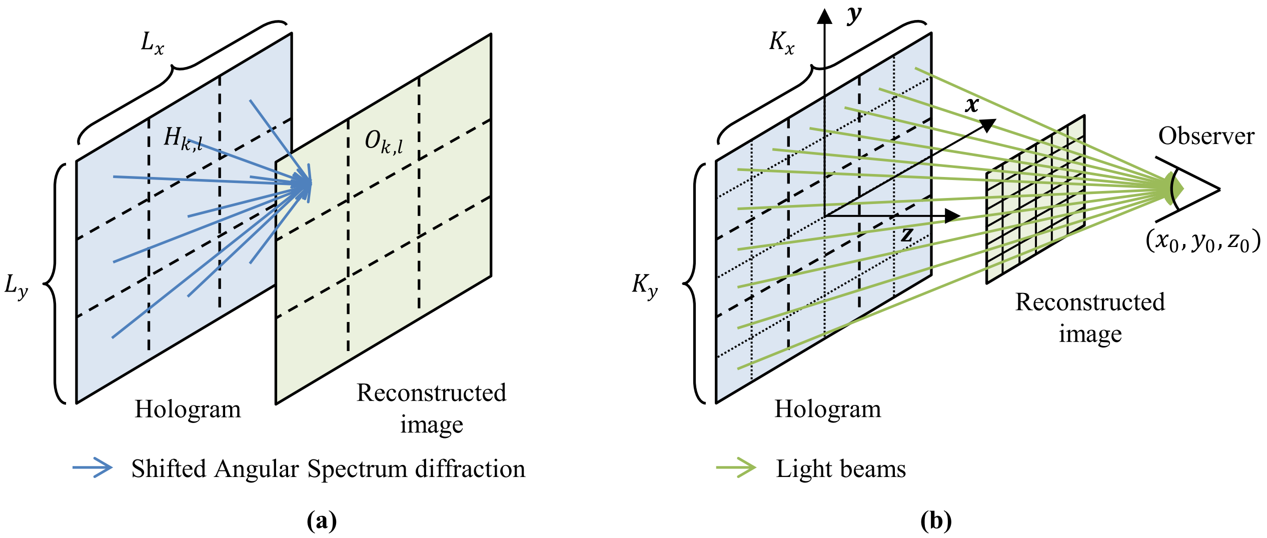


Figure 8: Hologram reconstruction using: (a) the shifted Angular Spectrum diffraction method, (b) our proposed light beams extraction method.

To overcome these limitations, we propose a numerical reconstruction algorithm based on a light beams extraction method. Our approach consists of extracting the light beams scattered by the hologram towards the observer's position, as shown in Figure 8(b). To this end, the hologram is subdivided into $(K_x \times K_y)$ rectangular segments which are Fourier transformed. As stated before, the Fourier transform of a complex-valued hologram corresponds to its decomposition into plane waves. The relation between the propagation angles (θ_x, θ_y) of these plane waves relative to the hologram optical axis and the frequency coordinates (f_x, f_y) is given by the grating equation such that

$$\sin(\theta_x) = \lambda f_x \quad \text{and} \quad \sin(\theta_y) = \lambda f_y. \tag{9}$$

Therefore, the reconstructed image I of resolution $(K_x \times K_y)$ is computed by collecting from the Fourier transformed segments the intensity of coefficients corresponding to plane waves propagating towards the observer's position (x_0, y_0, z_0), such that

$$I(k,l) = \left| \mathcal{F}\left\{H_{k,l}\right\} \left(\frac{\sin(\theta_x)}{\lambda}, \frac{\sin(\theta_y)}{\lambda} \right) \right|^2, \tag{10}$$

where

$$\theta_x = \arctan\left(\tfrac{x_0 - x_k}{z_0}\right) \quad \text{and} \quad \theta_y = \arctan\left(\tfrac{y_0 - y_l}{z_0}\right) \tag{11}$$

are the light beams propagation angles relative to the hologram optical axis, and (x_k, y_l) are the coordinates of the center of hologram segment $H_{k,l}$.

Since it involves only one Fourier transform per hologram segment, this method is computationally very efficient and does not require a large amount of memory. However, since the number of hologram segments depends on their resolution, this approach implies a trade-off between the spatial and angular resolutions: using a larger amount of hologram segments with a reduced resolution enables a better spatial resolution at the cost of a lower angular resolution, and vice-versa. To improve the angular resolution, it is possible to increase the size of the Fourier transform by zero-padding the hologram segments, or to use other space-frequency transforms such as the Short-Term Fourier Transform (STFT) or Gabor transform, at the cost of an increased computational complexity.

Finally, it must be noted that this algorithm only simulates the perception of the scene by the observer: it does not provide a complete nor physically accurate description of the complex light wave scattered by the hologram in the reconstruction plane.

4 Experimental results

The proposed method was implemented in C++/CUDA on a PC system employing an Intel Core i9-9900X CPU operating at 3.50 GHz, a main memory of 32 GB and an operating system of Microsoft Windows 10 as well as a GPU NVIDIA GeForce RTX 2080Ti.

Parameter	***Dices***	***Piano***
Location of the scene	Between 0.1cm and 1.5cm	Between 0.2cm and 1.5cm
Number of holographic segments $(L_x \times L_y)$	(64×64)	
Holographic segments resolution $(N'_x \times N'_y)$	(1024×1024)	
Total hologram resolution $(N_x \times N_y)$	$(65,536 \times 65,536)$	
Pixel pitch p	0.24μm	
Light wavelength λ	532nm	

Table 1: Hologram and scene parameters used for the experiments

For the experiments, we computed two holograms from different scenes: *Dices*, representing three dices in front of a chessboard, and *Piano*, representing a grand piano. Table 1 shows the parameters used for the calculation. To compute the holograms, we used the method described in Section 3.1, with (64×64) segments of resolution (1024×1024). The total resolution of the holograms was therefore $(64K \times 64K)$, corresponding to more than 4 billions of pixels. The holograms pixel pitch was set to 0.24μm and the light wavelength to 532nm, enabling a field of view of 180°. Finally, the scenes were located between 0.1cm and 1.5cm, and between 0.2cm and 1.5cm in front of the hologram, for the *Dices* and *Piano* holograms, respectively.

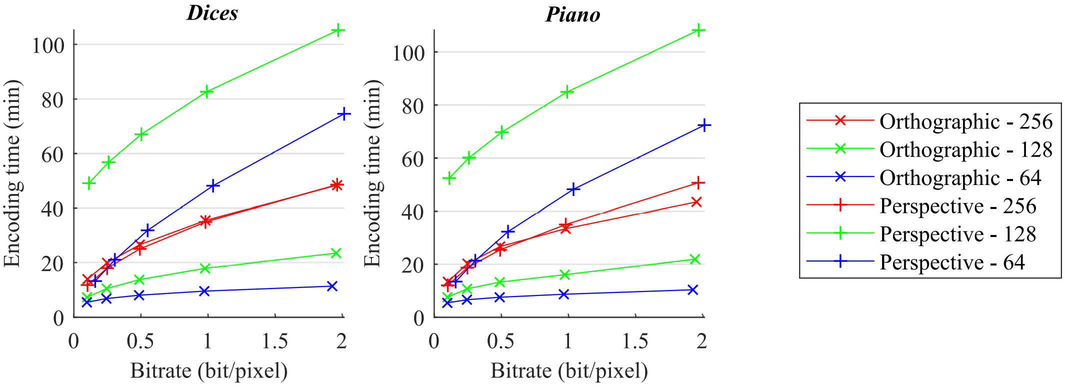


Figure 9: Hologram encoding times using the perspective and orthographic view extraction methods for different segment resolutions.

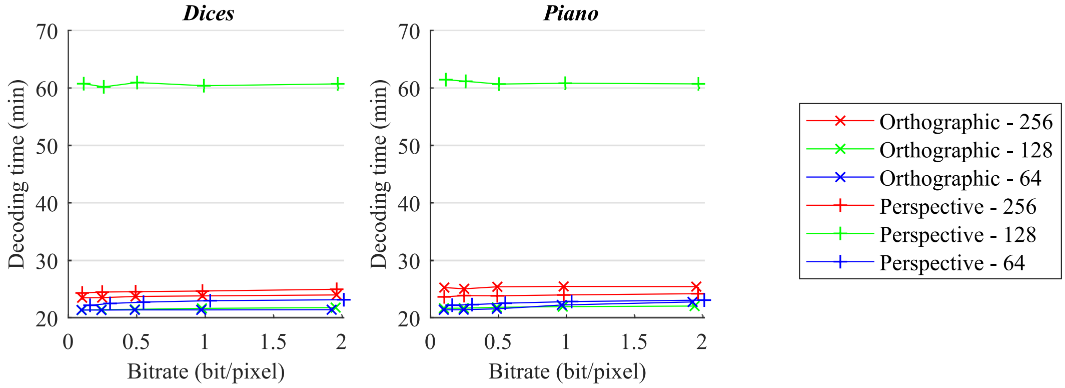


Figure 10: Hologram decoding times using the perspective and orthographic view extraction methods for different segment resolutions.

In the following, we assess and compare the calculation times and coding efficiency of the perspective and orthographic view extraction methods, using segments $H_{k,l}$ of resolution (64×64) and (128×128), as proposed in [25], as well as (256×256). For the encoding of segments in step 4, we used the ffmpeg 4.2.3 [29] x265 encoder with two-pass target bitrates of 0.1bpp, 0.25bpp, 0.5bpp, 1bpp and 2bpp. For the decoding, we used the OpenCV 4.3.0 [30] library with ffmpeg backend.

4.1 Coding and decoding times

Figure 9 shows the hologram encoding time using the perspective and orthographic view extraction methods for the different segment resolutions. A shown in this figure, the encoding time increases with the target bitrate and varies between 11.7 and 108.3 minutes and between 5.4 and 48.4 minutes for the perspective and orthographic view extraction methods, respectively, depending on the segment resolution. As a consequence, although the orthographic view extraction method comprises an additional pixels remapping step, it enables holograms to be encoded faster than the perspective view extraction method. This is likely due to the fact that the pixels remapping step produces a fewer number of orthographic segments $S_{m,n}$ with a larger resolution than perspective segments $H_{k,l}$, as shown in Table 2. In this configuration, the selected codec and profile spend less time in predicting spatial than temporal redundancies, encoding segments $S_{m,n}$ faster than segments $H_{k,l}$.

Figure 10 shows the hologram decoding time using the perspective and orthographic view extraction methods for the different segment resolutions. A shown in this figure, the decoding time does not depend on the target bitrate and varies between 22.2 and 61.4 minutes and between 21.3 and 25.4 minutes for the perspective and orthographic view extraction methods, respectively, depending on the segment resolution. In this configuration, the orthographic view extraction method therefore also enables holograms to be decoded faster than the perspective view extraction method.

4.2 Objective quality assessment in the hologram plane

To assess the objective quality in the hologram plane, we computed the Signal to Noise Ratio (SNR) between the compressed and reference holograms. The SNR (in dB) is calculated on the complex valued wavefield in

Resolution of segments $H_{k,l}$	Number of segments $H_{k,l}$	Resolution of segments $S_{m,n}$	Number of segments $S_{m,n}$
(64×64)	(1024×1024)	(1024×1024)	(64×64)
(128×128)	(512×512)	(512×512)	(128×128)
(256×256)	(256×256)	(256×256)	(256×256)

Table 2: Relation between the number and resolution of segments $H_{k,l}$ and $S_{m,n}$.

the hologram plane and is given by

$$\mathrm{SNR} = 10\log_{10}\left(\frac{\sum_{m=0}^{N_x-1}\sum_{n=0}^{N_y-1}|H(m,n)|^2}{\sum_{m=0}^{N_x-1}\sum_{n=0}^{N_y-1}\left|H(m,n)-\hat{H}(m,n)\right|^2}\right) \tag{12}$$

where H and $\hat{H}$ are the reference and compressed holograms, respectively, and $(N_x \times N_y)$ is the hologram resolution.

Figure 11 shows the rate-distortion (RD) performance of the perspective and orthographic view extraction methods in terms of SNR values for the different segment resolutions. As shown in this figure, for a given segment resolution, the orthographic view extraction method always provides higher SNR values than the perspective view extraction method, expect for the segment resolution of (256 × 256) at a bitrate of 2bpp. Furthermore, for low and medium bitrates (0.1, 0.25 and 0.5bpp), holograms encoded using the orthographic view extraction method have higher SNR values than those encoded using the perspective view extraction method, regardless of the segment resolution.

These RD performance results are summarized in terms of Bjontegaard Delta (BD) metrics [31] in Table 3. The Bjontegaard metrics are used to calculate the average SNR and bitrate differences between two codecs. Two metrics are computed: the BD-SNR, which corresponds to the average SNR gains in dB for the same bitrate, and the BD-Rate, which corresponds to the average bitrate savings in percentage for the same SNR. Since the perspective view extraction method with a segment resolution of (64 × 64) presents the worst RD performance, it was used as reference. As shown in this table, for the *Dices* hologram, the best RD performance is obtained using the orthographic view extraction method with a segment resolution of (256 × 256), achieving a bitrate saving of 61.25% compared to the reference method. For the *Piano* hologram, the best RD performance is obtained using the orthographic view extraction method with a segment resolution of (128 × 128), achieving a bitrate saving of 45.55% compared to the reference method.

4.3 Quality assessment of the numerically reconstructed scene images

To evaluate the visual quality of the numerically reconstructed scene images, we used the method described in Section 3.2, with a segment resolution of $(K_x \times K_y) = (32 \times 32)$. To improve the visual quality of reconstructed images, we increased the size of the Fourier transform three times by zero-padding the hologram segments. We used five different viewpoints, whose corresponding observer's coordinates are given in Table 4.

Figures 12 and 13 show respectively the Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) of the numerical reconstructions of compressed holograms *Dices*. These objective quality results

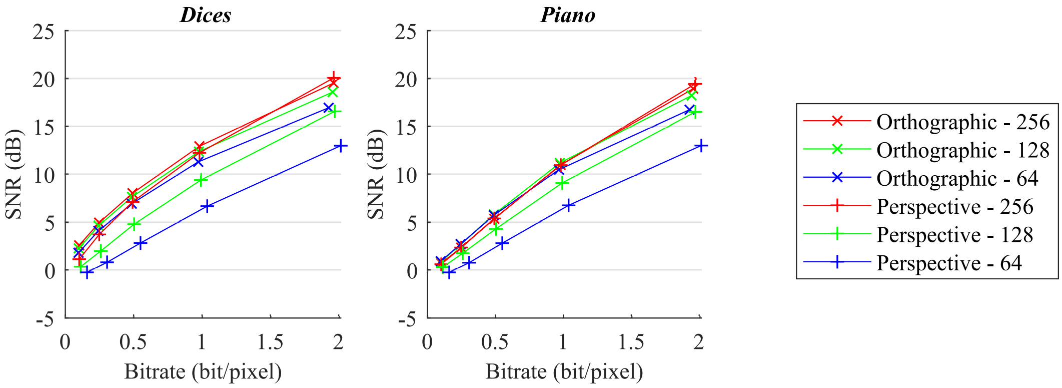


Figure 11: Signal-to-Noise Ratio (SNR) of the holograms encoded using the perspective and orthographic view extraction methods for different segment resolutions.

		Orthographic			Perspective	
Hologram	**Metric**	**(256 × 256)**	**(128 × 128)**	**(64 × 64)**	**(256 × 256)**	**(128 × 128)**
Dices	BD-SNR (dB)	**5,74**	5,30	4,47	4,95	2,44
	BD-Rate (%)	**-61,25**	-59,32	-54,55	-54,83	-33,55
Piano	BD-SNR (dB)	3,55	**3,79**	3,44	3,60	2,13
	BD-Rate (%)	-41,77	**-45,55**	-44,45	-41,92	-29,92

Table 3: Bjontegaard delta metrics for the different compression algorithms with respect to the perspective view extraction method with a segment resolution of (64 × 64)

are consistent with observations made in the hologram plane: for a given segment resolution, the orthographic view extraction method always provides higher PSNR and SSIM values than the perspective view extraction method, expect for the segment resolution of (256 × 256) in the Bottom-left and Bottom-right views at a bitrate of 2bpp. Furthermore, for low bitrates (0.1 and 0.25bpp), holograms encoded using the orthographic view extraction method have higher PSNR and SSIM values than those encoded using the perspective view extraction method, regardless of the segment resolution. Overall, the orthographic view extraction method with a segment resolution of (256 × 256) outperforms all the other methods in terms of compression efficiency.

These results are confirmed by the Center and Top-left numerical reconstructions of hologram *Dices* compressed at bitrates of 0.5bpp and 0.25bpp, shown in Figures 14 and 15, respectively. As visualized in these figures, the perspective view extraction method produces strong visual artifacts when using segment resolutions of (64 × 64) and (128 × 128), making the Arabic numeral dice is barely visible. On the other hand, the dices are always clearly visible using the orthographic view extraction method, regardless of the segment resolution.

Similarly, Figures 16 and 17 show the PSNR and SSIM of the numerical reconstructions of compressed holograms *Piano*. These objective quality results are consistent with previous observations, with the exception that the perspective view extraction method provides the best PSNR and SSIM results for the Center, Bottom-left and Bottom-right views at bitrates of 1 and 2bpp. Nevertheless, the best RD performance is obtained using the orthographic view extraction method with a segment resolution of (128 × 128) for bitrates of 0.1, 0.25 and 0.5bpp, and using the orthographic view extraction method with a segment resolution of (256×256) for bitrates of 1 and 2bpp in the Top-left and Top-right views.

These results are also confirmed by the Center and Top-right numerical reconstructions of hologram *Piano* compressed at bitrates of 1bpp and 0.5bpp, shown in Figures 18 and 19, respectively. As visualized in these figures, the perspective view extraction method produces stronger visual artifacts than the orthographic view extraction method, making the piano and chair legs partially disappear for segment resolutions of (64 × 64) and (128 × 128) at 1bpp and for every segment resolution at 0.5bpp.

These visual and objective quality results confirm the superiority of the orthographic view extraction method over the perspective view extraction method in terms of coding performance.

	Center	**Top-left**	**Top-right**	**Bottom-left**	**Bottom-right**
x (cm)	0	−0.8	0.8	−0.8	0.8
y (cm)	0	0.8	0.8	−0.8	−0.8
z (cm)	3.2	3.2	3.2	3.2	3.2

Table 4: Reconstruction viewpoints and corresponding observer's coordinates

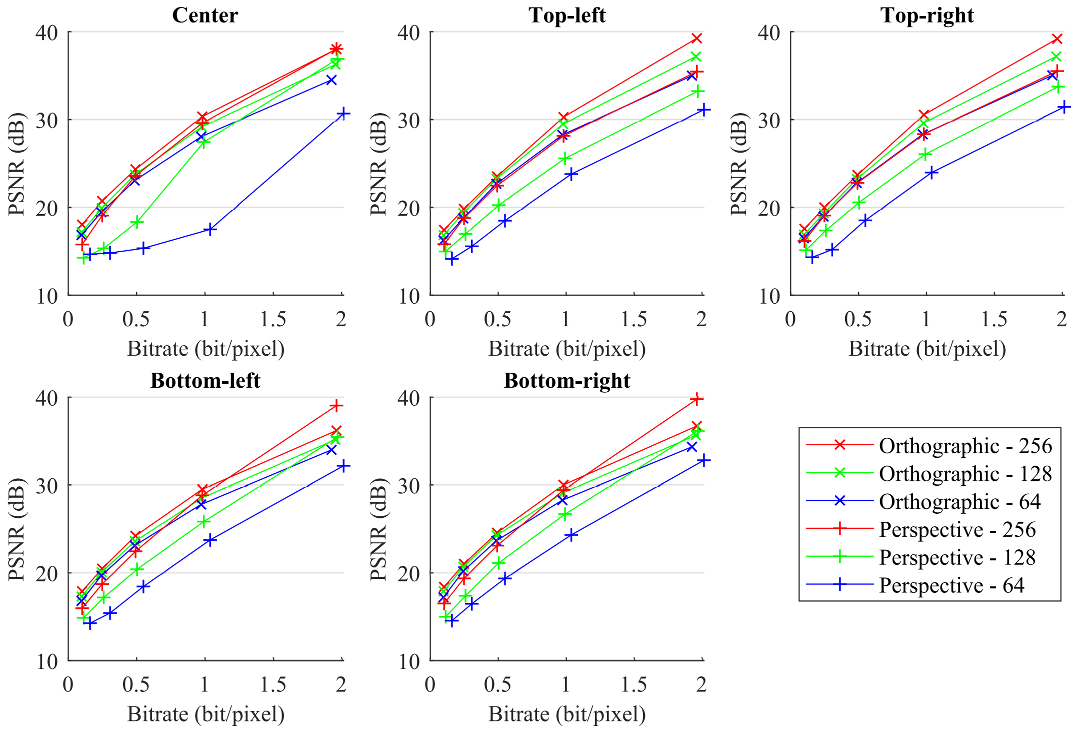


Figure 12: Peak Signal-to-Noise Ratio (PSNR) of the numerical reconstructions of hologram *Dices*.

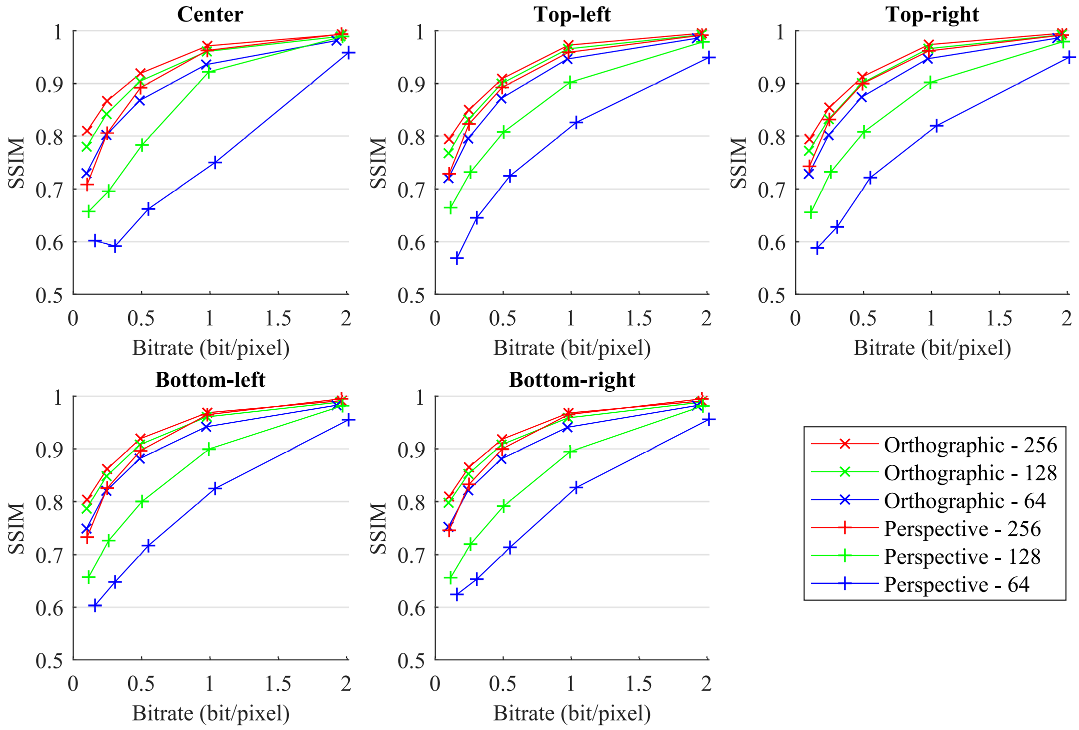
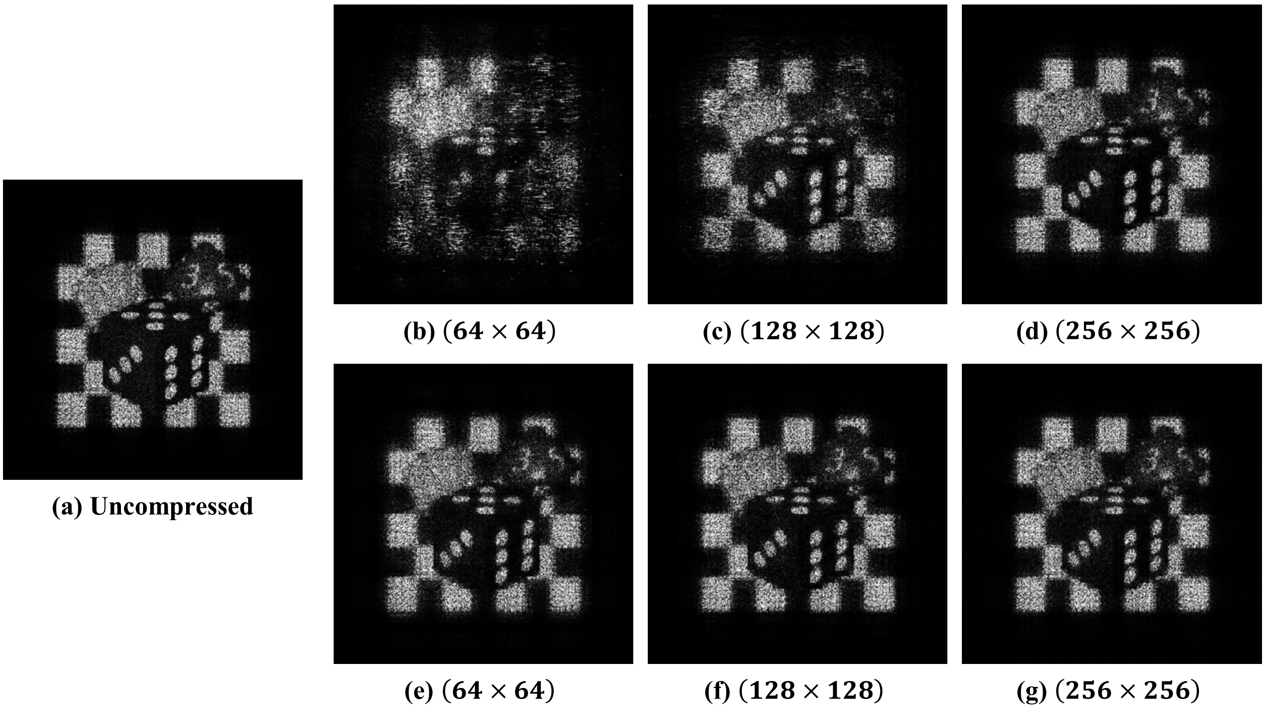


Figure 13: Structural Similarity Index Measure (SSIM) of the numerical reconstructions of hologram *Dices*.



(a) Uncompressed

(b) (64 × 64)

(c) (128 × 128)

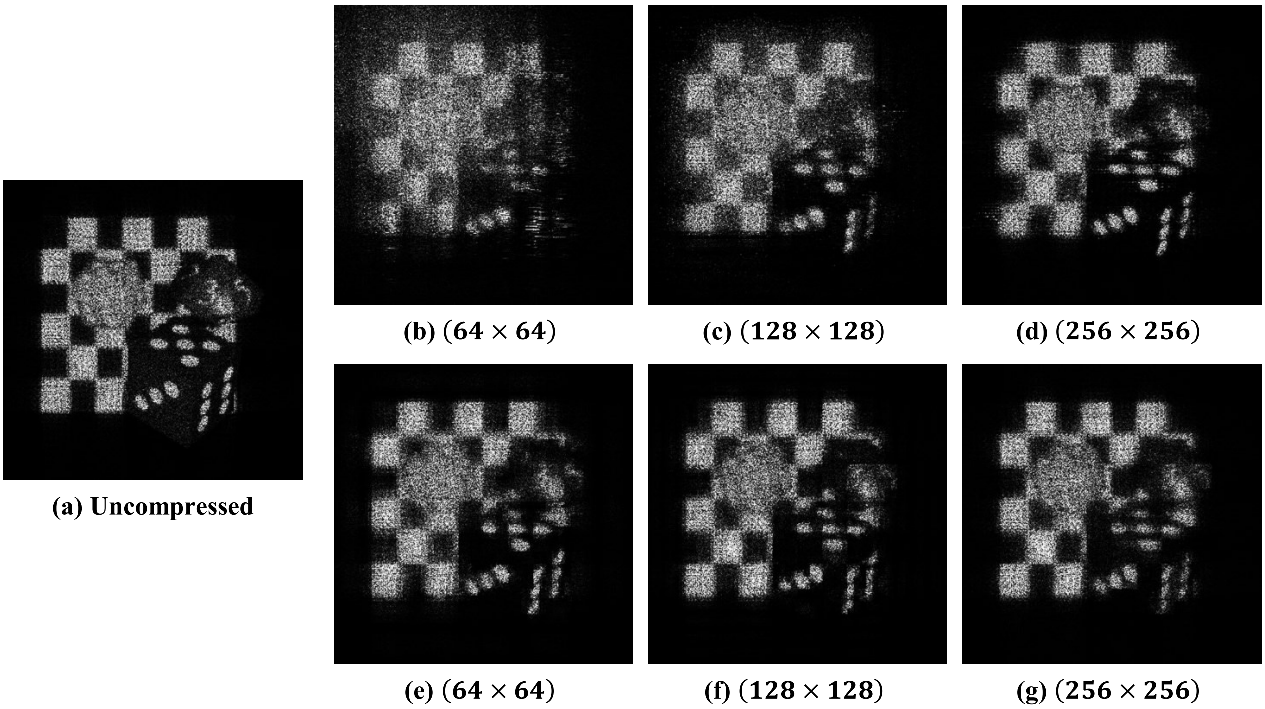
(d) (256 × 256)

(e) (64 × 64)

(f) (128 × 128)

(g) (256 × 256)

Figure 14: Center numerical reconstructions from *Dices*, encoded for a bitrate of 0.5bpp using the perspective (b-d) and orthographic (e-g) view extraction methods for different segment resolutions.



(a) Uncompressed

(b) (64 × 64)

(c) (128 × 128)

(d) (256 × 256)

(e) (64 × 64)

(f) (128 × 128)

(g) (256 × 256)

Figure 15: Top-left numerical reconstructions from *Dices*, encoded for a bitrate of 0.25bpp using the perspective (b-d) and orthographic (e-g) view extraction methods for different segment resolutions.

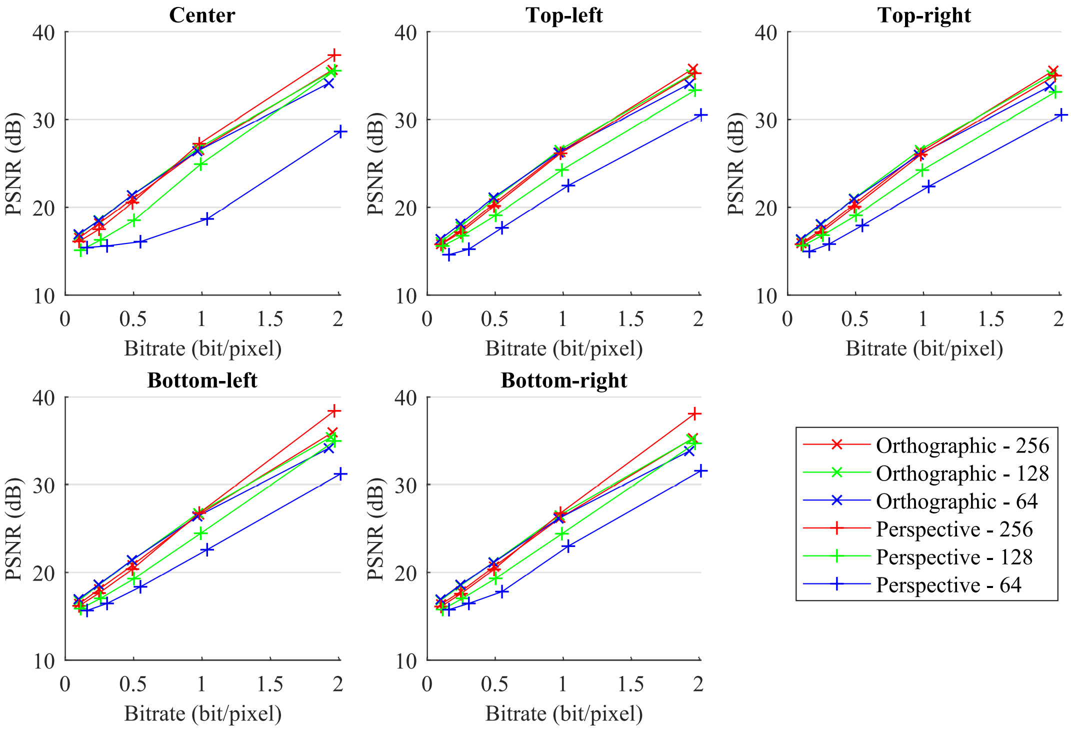


Figure 16: Peak Signal-to-Noise Ratio (PSNR) of the numerical reconstructions of hologram *Piano*.

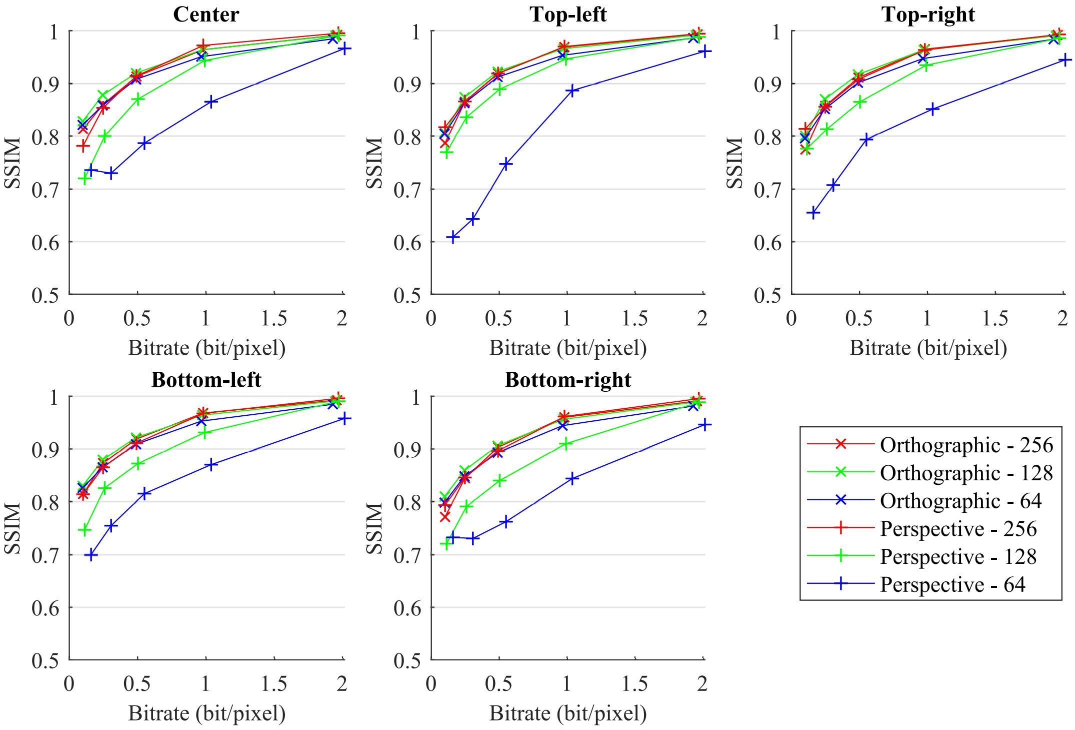
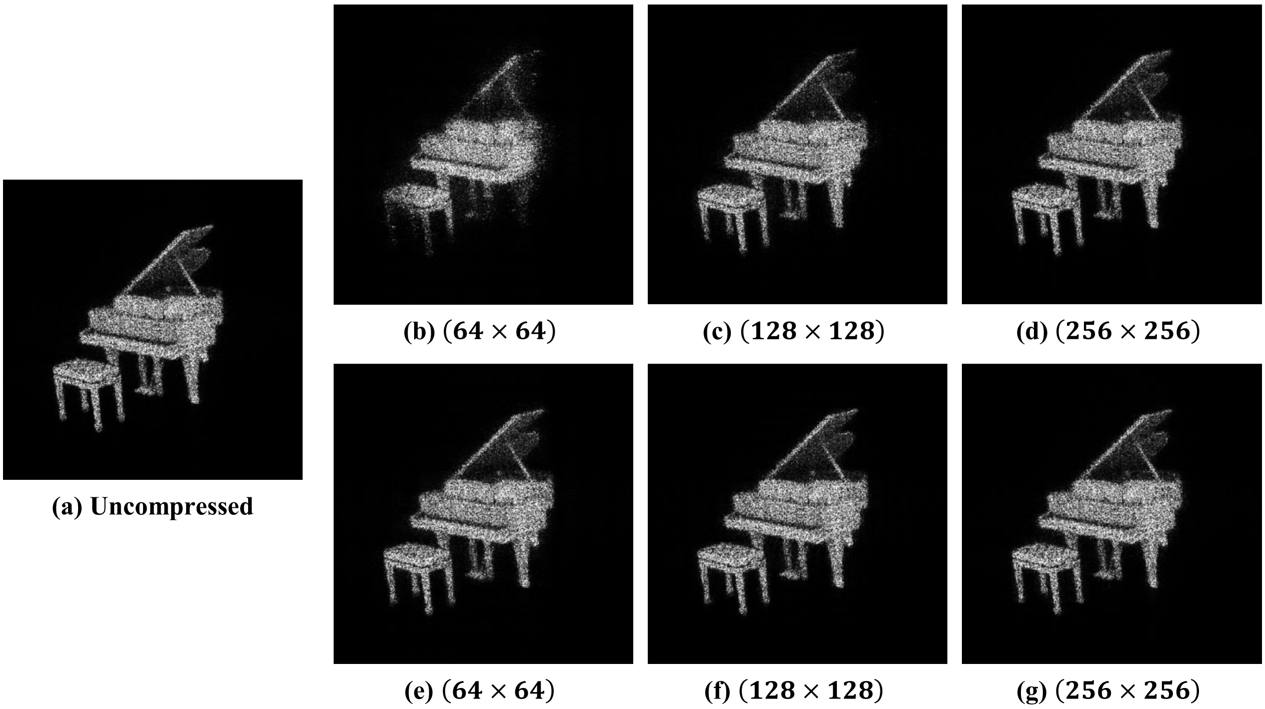


Figure 17: Structural Similarity Index Measure (SSIM) of the numerical reconstructions of hologram *Piano*.



(a) Uncompressed

(b) (64×64)

(c) (128×128)

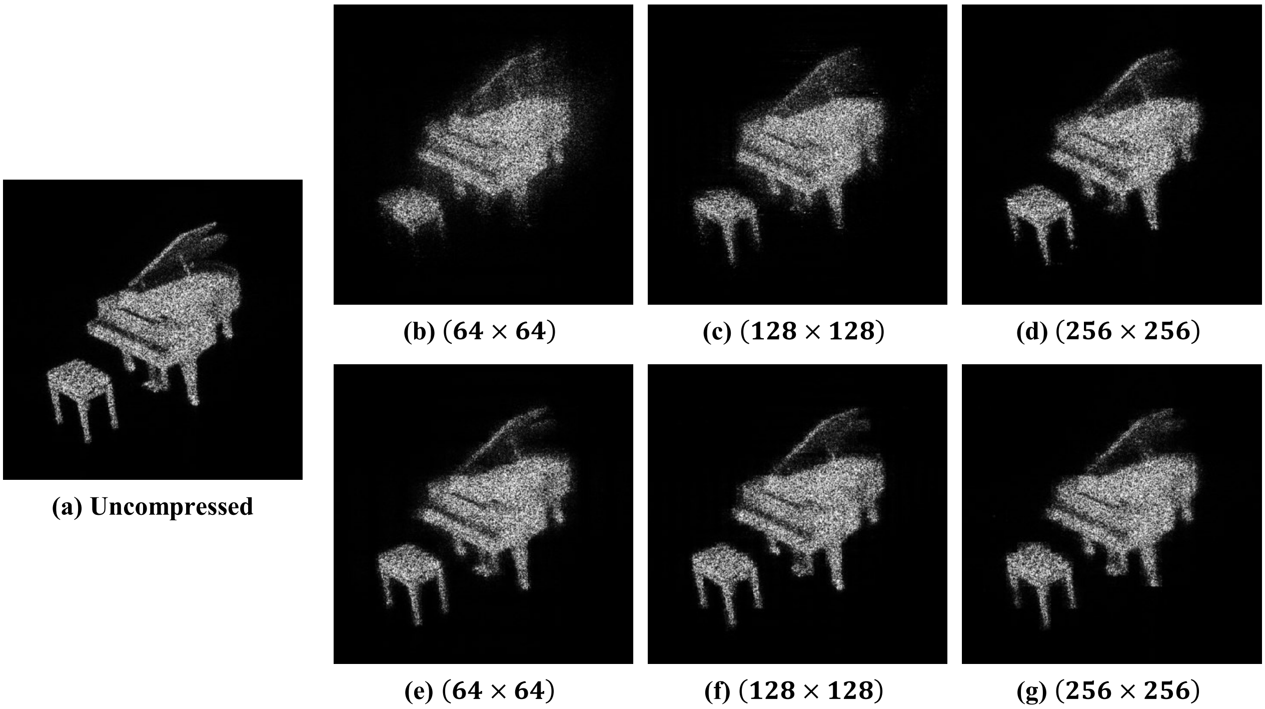
(d) (256×256)

(e) (64×64)

(f) (128×128)

(g) (256×256)

Figure 18: Center numerical reconstructions from *Piano*, encoded for a bitrate of 1.0bpp using the perspective (b-d) and orthographic (e-g) view extraction methods for different segment resolutions.



(a) Uncompressed

(b) (64×64)

(c) (128×128)

(d) (256×256)

(e) (64×64)

(f) (128×128)

(g) (256×256)

Figure 19: Top-right numerical reconstructions from *Piano*, encoded for a bitrate of 0.5bpp using the perspective (b-d) and orthographic (e-g) view extraction methods for different segment resolutions.

5 Conclusion

In this paper, we presented and compared two compression methods based on hologram-lightfield transforms, which are suitable to the compression of extremely-high resolution holograms containing several billions of pixels. The first method, derived from Seo *et al.* [25], uses a perspective view extraction method, while the second is based on a novel orthographic view extraction method. The orthographic view extraction method presents the advantage of producing images with a very large depth of field and reduced speckle noise compared to the perspective view extraction method, facilitating their encoding using conventional video codecs.

To evaluate and compare the coding efficiency of these compression algorithms, we computed two holograms of resolution ($64K \times 64K$) using a layered holographic stereogram calculation method. We also proposed a computationally and memory efficient numerical reconstruction algorithm based on a light beams extraction method, which consists of extracting the light beams scattered by the hologram towards the observer's position.

The rate-distortion performance of the perspective and orthographic view extraction methods was assessed in the hologram plane using the SNR metric and in the numerical reconstructions using PSNR and SSIM. Experimental results show that the orthographic view extraction method overcomes the perspective view extraction method in terms of coding efficiency and coding and decoding times. These objective quality metrics are also confirmed visually in the numerical reconstructions. In a future work, we will investigate other space-frequency transforms, such as the Gabor transform or S-method [32] for the extraction of perspective and orthographic views from the hologram.

References

[1] Joseph W. Goodman. *Introduction to Fourier Optics.* Roberts and Company Publishers, Englewood, Colo, 3rd edition, 2005.

[2] Ulf Schnars and Werner Jüptner. *Digital Holography: Digital Hologram Recording, Numerical Reconstruction, and Related Techniques.* Springer Science & Business Media, December 2005.

[3] Andrew Maimone, Andreas Georgiou, and Joel S. Kollin. Holographic Near-eye Displays for Virtual and Augmented Reality. *ACM Trans. Graph.*, 36(4):85:1–85:16, July 2017.

[4] Zehao He, Xiaomeng Sui, Guofan Jin, and Liangcai Cao. Progress in virtual reality and augmented reality based on holographic display. *Applied Optics*, 58(5):A74–A81, February 2019.

[5] Yongjun Lim, Keehoon Hong, Hwi Kim, Hyun-Eui Kim, Eun-Young Chang, Soohyun Lee, Taeone Kim, Jeho Nam, Hyon-Gon Choo, Jinwoong Kim, and Joonku Hahn. 360-degree tabletop electronic holographic display. *Optics Express*, 24(22):24999–25009, October 2016.

[6] Jinwoong Kim, Yongjun Lim, Keehoon Hong, Eun-Young Chang, and Hyon-gon Choo. 360-degree Tabletop Color Holographic Display. In *Digital Holography and Three-Dimensional Imaging (2017), paper W3A.1*, page W3A.1. Optical Society of America, May 2017.

[7] Kenji Yamamoto, Yasuyuki Ichihashi, Takanori Senoh, Ryutaro Oi, and Taiichiro Kurita. 3D objects enlargement technique using an optical system and multiple SLMs for electronic holography. *Optics express*, 20(19):21137–21144, 2012.

[8] Weronika Zaperty, Tomasz Kozacki, and Małgorzata Kujawińska. Multi-SLM Color Holographic 3D Display Based on RGB Spatial Filter. *Journal of Display Technology*, 12(12):1724–1731, December 2016.

[9] H. Yoshikawa and T. Yamaguchi. Review of Holographic Printers for Computer-Generated Holograms. *IEEE Transactions on Industrial Informatics*, 12(4):1584–1589, August 2016.

[10] Youngmin Kim, Elena Stoykova, Hoonjong Kang, Sunghee Hong, Joosup Park, Jiyong Park, and Jisoo Hong. Seamless full color holographic printing method based on spatial partitioning of SLM. *Optics Express*, 23(1):172–182, January 2015.

[11] Peter Schelkens, Touradj Ebrahimi, Antonin Gilles, Patrick Gioia, Kwan-Jung Oh, Fernando Pereira, Cristian Perra, and Antonio M. G. Pinheiro. JPEG Pleno: Providing representation interoperability for holographic applications and devices. *ETRI Journal*, 41(1):93–108, February 2019.

[12] Le Thanh Bang, Zulfiqar Ali, Pham Duc Quang, Jae-Hyeung Park, and Nam Kim. Compression of digital hologram for three-dimensional object using Wavelet-Bandelets transform. *Optics Express*, 19(9):8019, April 2011.

[13] David Blinder, Tim Bruylants, Heidi Ottevaere, Adrian Munteanu, and Peter Schelkens. JPEG 2000-based compression of fringe patterns for digital holographic microscopy. *Optical Engineering*, 53(12):123102–123102, December 2014.

[14] J. P. Peixeiro, C. Brites, J. Ascenso, and F. Pereira. Holographic Data Coding: Benchmarking and Extending HEVC With Adapted Transforms. *IEEE Transactions on Multimedia*, 20(2):282–297, February 2018.

[15] Yafei Xing, Mounir Kaaniche, Béatrice Pesquet-Popescu, and Frédéric Dufaux. Vector lifting scheme for phase-shifting holographic data compression. *Optical Engineering*, 53(11):112312–112312, May 2014.

[16] Yafei Xing, Mounir Kaaniche, Béatrice Pesquet-Popescu, and Frédéric Dufaux. Adaptive nonseparable vector lifting scheme for digital holographic data compression. *Applied Optics*, 54(1):A98–A109, January 2015.

[17] Tobias Birnbaum, Ayyoub Ahar, David Blinder, Colas Schretter, Tomasz Kozacki, and Peter Schelkens. Wave atoms for digital hologram compression. *Applied Optics*, 58(22):6193–6203, August 2019.

[18] Levent Onural. Diffraction from a wavelet point of view. *Optics letters*, 18(11):846–848, 1993.

[19] Michael Liebling, Thierry Blu, and Michael Unser. Fresnelets: new multiresolution wavelet bases for digital holography. *IEEE transactions on image processing: a publication of the IEEE Signal Processing Society*, 12(1):29–43, 2003.

[20] E. Darakis and J. J. Soraghan. Use of Fresnelets for Phase-Shifting Digital Hologram Compression. *IEEE Transactions on Image Processing*, 15(12):3804–3811, December 2006.

[21] D. Blinder, C. Schretter, H. Ottevaere, A. Munteanu, and P. Schelkens. Unitary Transforms Using Time-Frequency Warping for Digital Holograms of Deep Scenes. *IEEE Transactions on Computational Imaging*, 4(2):206–218, June 2018.

[22] Anas El Rhammad, Patrick Gioia, Antonin Gilles, Marco Cagnazzo, and Béatrice Pesquet-Popescu. Color digital hologram compression based on matching pursuit. *Applied Optics*, 57(17):4930–4942, June 2018.

[23] Anas El Rhammad, Patrick Gioia, Antonin Gilles, Marco Cagnazzo, and Béatrice Pesquet-Popescu. View-dependent compression of digital hologram based on matching pursuit. In *Optics, Photonics, and Digital Technologies for Imaging Applications V*, volume 10679, page 106790L. International Society for Optics and Photonics, May 2018.

[24] Anas El Rhammad, Patrick Gioia, Antonin Gilles, and Marco Cagnazzo. Progressive hologram transmission using a view-dependent scalable compression scheme. *Annals of Telecommunications*, 75(5):201–214, June 2020.

[25] Young-Ho Seo, Hyun-Jun Choi, and Dong-Wook Kim. 3D scanning-based compression technique for digital hologram video. *Signal Processing: Image Communication*, 22(2):144–156, February 2007.

[26] Hao Zhang, Yan Zhao, Liangcai Cao, and Guofan Jin. Layered holographic stereogram based on inverse Fresnel diffraction. *Applied Optics*, 55(3):A154, January 2016.

[27] Kyoji Matsushima. Shifted angular spectrum method for off-axis numerical propagation. *Optics Express*, 18(17):18453, August 2010.

[28] David Blinder and Tomoyoshi Shimobaba. Efficient algorithms for the accurate propagation of extreme-resolution holograms. *Optics Express*, 27(21):29905, October 2019.

[29] FFmpeg 4.2.3. https://ffmpeg.org/.

[30] OpenCV 4.3.0. https://opencv.org/.

[31] Gisle Bjontegaard. Improvements of the BD-PSNR model. pages ITU–T SG16 Q.6 Document, VCEG–AI11, ITU 35th meeting VCEG, Berlin, Germany, July 2008.

[32] Tobias Birnbaum, Tomasz Kozacki, and Peter Schelkens. Providing a Visual Understanding of Holography Through Phase Space Representations. *Applied Sciences*, 10(14):4766, January 2020. Number: 14 Publisher: Multidisciplinary Digital Publishing Institute.