Integer Fresnel Transform for Lossless Hologram Compression

Blinder, David; Schelkens, Peter

Published in:
2019 Data Compression Conference (DCC)

DOI:
10.1109/DCC.2019.00047

Publication date:
2019

Integer Fresnel Transform for Lossless Hologram Compression

David Blinder*,† and Peter Schelkens*,†

*Vrije Universiteit Brussel
Pleinlaan 2
B-1050 Brussels, Belgium
dblinder@etrovub.be

†imec
Kapeldreef 75
B-3001 Leuven, Belgium
pschelke@etrovub.be

Abstract

Digital holograms fully encode the wavefield of light, thereby having many applications both for 3D object measurements as well as for display purposes, accounting for all human visual cues. Because the statistics and properties of holographic signals differ considerably from natural imagery such as photographs, conventional coding solutions will be sub-optimal. In this paper, we propose the integer Fresnel transform, which is – to our knowledge – the first lossless transform tailored for hologram coding. By combining the proposed transform with JPEG 2000, we report bit-rate savings from 0.12 up to 2.83 bits per channel on a collection of 8 digital holograms obtained from 3 different databases.

Introduction

Holography can fully represent the plenoptic function of a three-dimensional (3D) scene [1], and thereby has various applications in microscopy [2], tomography [3], interferometry [4] and storage technology [5]. However, the primary application on which we will focus is the use of digital holography for 3D display technology [6], [7].

Since holography encodes the wave field of light, it can account for all human visual cues, including continuous parallax, stereopsis, occlusion and exhibits no vergence-accommodation conflicts [8]. However, current holographic displays do not yet possess the requisite display size and field-of-view for acceptable visual quality. Although several technological obstacles remain, steady progress in photonics, microelectronics and computer engineering indicate that high-quality holography displays are coming within reach [9].

Because the working principle of holography differs fundamentally from conventional imaging systems, they exhibit highly different signal statistics and properties as well [10]. For example, holograms have much more high-frequency content, can have strong directionality because of the interference fringes, often possess speckle noise, and localised information in the 3D scene will diffract, spreading out and potentially affect all pixels of the digital hologram. That is why conventional representation and encoding algorithms, such as the standard JPEG and MPEG families of codecs, are suboptimal for hologram coding. Furthermore, holograms for multi-user holographic displays need resolutions of several Gigapixels [10], increasing the need for efficient compression methodologies.

Hence, novel coding solutions and transforms are needed to compress holograms more effectively. Several techniques have been proposed over recent years to tackle this problem: Fresnelets [11], wavelet-bandelets [12], directional-adaptive wavelets and arbitrary packet decompositions [13], vector quantisation lifting schemes [14], wave atom transforms [15], mode dependent directional transform-based HEVC [16], overcomplete Gabor wavelet dictionary [17] and nonlinear canonical transforms [18].

In particular, we will use the principle of *object plane coding* [19], [20]: by back-propagating the hologram wavefield using *e.g.* the Fresnel transform to the object plane, we effectively refocus the hologram to a representation resembling an image, making it subsequently better compressible by conventional transforms and codecs. This approach has been shown to be especially effective when the hologram's depth of focus is large.

However, no lossless transforms have been proposed specifically designed for holographic signals, to our knowledge. This is important for archival purposes, e.g. when storing holographic measurements, especially biological samples; another application is to store holograms intended for display at the highest possible quality settings. In this paper, we propose the integer Fresnel transform, which is the first lossless transform designed for hologram coding. In the remainder of this paper, we will first introduce the basic principles of holography, lay out how to implement the lossless integer variant of the Fresnel transform, and report increased compression performance on a data set of holograms for display systems.

Holography

Holography is a 3D interference-based imaging methodology, which can capture and reproduce the scalar complex-valued wavefield of light, i.e. both amplitude and phase. Because the phase carries no energy, it cannot be measured directly. That is why in digital holography, we capture wavefields indirectly through interference with a digital, lens-less camera [21]. This sampled wavefield can then be processed digitally to extract 3D information from the imaged scene, or alternatively be fed to a Spatial Light Modulator (SLM) to display the hologram for viewing. Digital holograms can also be computer-generated by simulating numerical diffraction; this is often preferred for display systems, because it is not bound by the limitations of holographic measurement systems (camera resolutions, object size and placement limitations, aberrations, complexity of building and operating holographic setups) and can display synthetic content.

Scalar diffraction theory can model the propagation of light, mathematically describing how the complex-valued wavefield $u(x, y, z)$ evolves over space [23]. To model diffraction between parallel planes, we can use the Fresnel approximation:

$$u(x, y, z+d) = \frac{\exp(ikd)}{i\lambda d} \iint_{\mathbb{R}^2} u(x', y', z) \exp\left(\frac{ik}{2d}\left[(x-x')^2 + (y-y')^2\right]\right) dx'dy' \quad (1)$$

for a inter-planar propagation distance d along dimension z, for a hologram wavelength λ and corresponding wave number $k = \frac{2\pi}{\lambda}$. This is represented on Fig. 1.

Contrary to the more physically accurate Angular spectrum method [23], the Fresnel operator is separable, which will be useful later on. Henceforth, we will describe all operations on one-dimensional signals for notational simplicity.

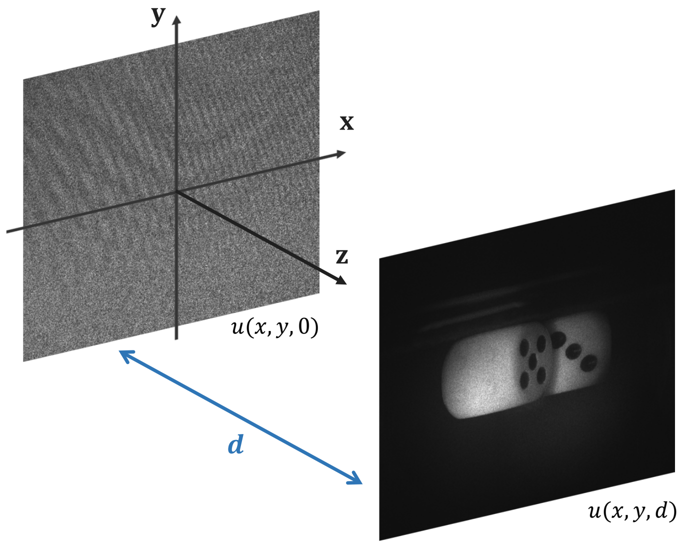


Figure 1: Diagram depicting the scene geometry and coordinate system. The hologram plane is centered at the origin, where $|u(x, y, 0)|$ is drawn. This represents the hologram at $u(x, y, 0)$ that needs to be compressed. The propagation direction z is perpendicular to the hologram plane. Using the Fresnel operator for a distance d, $u(x, y, d)$ can be computed. The object plane for the hologram "Dices 2" is shown (source: EmergImg-HoloGrail v2 database [22]), where $|u(x, y, d)|$ is in focus.

There are several techniques for calculating the Fresnel transform [24]. The diffraction operator is an all-pass filter, which can be written as a concatenation of Fourier transforms and pure phase delay functions. For example, it can be written as a convolution (operator $*$) with a quadratic phase delay filter [24]:

$$u(x, z + d) = \frac{\exp(ikd)}{i\lambda d} u(x, z) \, * \, \exp\left(\frac{i\pi}{\lambda d} x^2\right) \tag{2}$$

We will consider the two most common numerical implementations, which we will call the *convolutional* form and the *Fourier* form. The main difference between the techniques is the relation between the *pixel pitches* of the source plane (p_s) and destination plane (p_d): this is the physical sampling distance between successive pixel centres of the regularly sampled hologram. Depending on the (virtual) object dimensions, hologram placement and scene geometry, one method will often be preferable to better visualise the reconstructed object.

The convolutional form F_d preserves the pixel pitch, i.e. $p_\mathrm{s} = p_\mathrm{d} = p$. It can be written in matrix form:

$$F_d = c \cdot \mathcal{F}^{-1} \cdot D^{-\lambda d/p^2} \cdot \mathcal{F} \tag{3}$$

where $\mathcal{F}$ is a (unitary) $n \times n$ DFT matrix (computed using the FFT), $c = \exp(ikd)$ and D is a diagonal matrix with a quadratic phase profile and unit amplitude, defined as

$$D = \mathrm{diag}\Big(\exp\left(i\pi(\tfrac{x}{n})^2\right) \Big) \quad \text{for} \quad x \in \left\{-\frac{n}{2}, -\frac{n}{2}+1, ..., \frac{n}{2}-1\right\} \tag{4}$$

for a signal consisting of n samples. We omit the additional constant division of c by $i\lambda d$ to obtain a unitary Fresnel transform, which is otherwise present in the typical definition [24]; this is done to obtain a similar dynamic range in the object plane w.r.t. the hologram plane for the coding efficiency of the integer Fresnel transforms.

The Fourier form Ψ_d will change the pixel pitch of the destination plane depending on the wavelength and propagation distance, namely $p_{\mathrm{d}} = \frac{p_{\mathrm{s}}}{\lambda d}$:

$$\Psi_d = c \cdot D^{p_{\mathrm{d}}^2/\lambda d} \cdot \mathcal{F} \cdot D^{p_{\mathrm{s}}^2/\lambda d} \tag{5}$$

As is, these numerical Fresnel transforms cannot be used for lossless coding, because they are not defined for integer implementations. This will be addressed in the next section.

Methodology

For lossless coding, the aim is to transform the signal in a reversible manner as to obtain a more compressible representation, leading to smaller file sizes. Formally, suppose we have an discrete signal x consisting of $2n$ integer samples. We want to construct a bijective $f : \mathbb{Z}^{2n} \to \mathbb{Z}^{2n}$ tailored to the statistical properties of the signal under investigation.

To that end, we can use the lifting scheme [25]. In its general form, it will partition the input signal x in two parts a_0 and b_0. Then, a cascade of reversible functions P_j (prediction) and U_j (update) are applied alternatingly on all elements of a_j and b_j:

$$b_{j+1} = \mathrm{P}_j(a_j, b_j) \quad \text{and} \quad a_{j+1} = \mathrm{U}_j(a_j, b_{j+1}) \tag{6}$$

where $\mathrm{P}_j(a_j, \cdot\,)$ and $\mathrm{U}_j(\,\cdot\,, b_{j+1})$ must be bijections for any integer signal a_j or b_{j+1}, $\forall j \in \{0, 1, ..., m\}$. Typically, the following implementation is used for lossless coding:

$$\mathrm{P}_j(a_j, b_j) = [\Pi_j(a_j)] \pm b_j \quad \text{and} \quad \mathrm{U}_j(a_j, b_{j+1}) = [\Upsilon_j(b_{j+1})] \pm a_j \tag{7}$$

with Π_j and Υ_j being (potentially non-linear) functions, and $[\,\cdot\,]$ is the rounding operator. When Π_j and Υ_j are linear, and $a_j, b_j \in \mathbb{Z}^n$ are vectors of equal length, we can rewrite them in matrix form:

$$\begin{pmatrix} a_j \\ b_{j+1} \end{pmatrix} = \left[\begin{pmatrix} I_n & 0_n \\ \Pi_j & I_n \end{pmatrix} \begin{pmatrix} a_j \\ b_j \end{pmatrix} \right] \quad \text{and} \quad \begin{pmatrix} a_{j+1} \\ b_{j+1} \end{pmatrix} = \left[\begin{pmatrix} I_n & \Upsilon_j \\ 0_n & I_n \end{pmatrix} \begin{pmatrix} a_j \\ b_{j+1} \end{pmatrix} \right] \tag{8}$$

where 0_n is a $n \times n$ matrix of zeros, I_n is a $n \times n$ identity matrix and Π_j, Υ_j can be any $n \times n$ matrix. We will call this particular matrix form a *lifting matrix*. The goal is to factor the sought transform into a product of lifting matrices, thereby obtaining an integer approximation of a transform defined on $\mathbb{R}$ (or $\mathbb{C}$).

In [26], the authors propose the following lifting matrix decomposition applicable for any invertible matrix M:

$$\begin{pmatrix} M & 0_n \\ 0_n & M^{-1} \end{pmatrix} = \begin{pmatrix} -I_n & 0_n \\ M^{-1} & I_n \end{pmatrix} \begin{pmatrix} I_n & -M \\ 0_n & I_n \end{pmatrix} \begin{pmatrix} 0_n & I_n \\ I_n & M^{-1} \end{pmatrix} \tag{9}$$

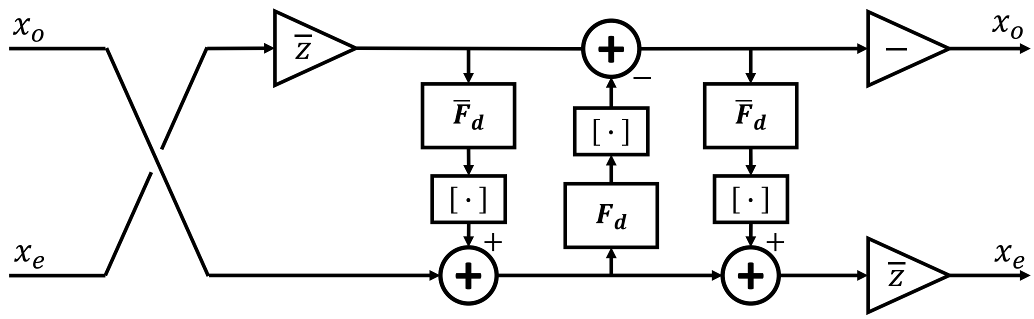


Figure 2: Graphical representation of the lifting scheme used for the convolutional integer Fresnel transform. The lifting is applied on the odd (x_o) and even (x_e) rows/columns of the hologram.

Thus, half of the signal will be transformed by M, and the other half by M^{-1}.

This property is useful because the Fresnel operators are unitary. We can redefine the inverse Fresnel transform as a function of the forward Fresnel transform. For the *convolutional* form, we have that

$$F_d^{-1} = \left(c\,\mathcal{F}^{-1}\,D^{-\lambda d/p^2}\,\mathcal{F}\right)^{-1} = \overline{c}\,\mathcal{F}^{-1}\,D^{\lambda d/p^2}\,\mathcal{F} = F_{-d} = \overline{F}_d \tag{10}$$

where $\overline{z}$ is the complex conjugate operator on some $z \in \mathbb{C}$. This means that after the lifting procedure, the first part of the signal will be transformed by F_d, and the second part by $\overline{F}_d$. We can thus take the complex conjugate on the second signal part to apply a convolutional Fresnel transform with distance d on both signals, *cf.* Fig. 2. For the *Fourier* form case, we can use the following decomposition:

$$\Psi_d = c\,D^{(p_\mathrm{d}^2 - p_\mathrm{s}^2)/\lambda d}\,\left(D^{p_\mathrm{s}^2/\lambda d}\,\mathcal{F}\,D^{p_\mathrm{s}^2/\lambda d}\right) = c\,D^{(p_\mathrm{d}^2 - p_\mathrm{s}^2)/\lambda d}\,\Phi_d \tag{11}$$

where we have

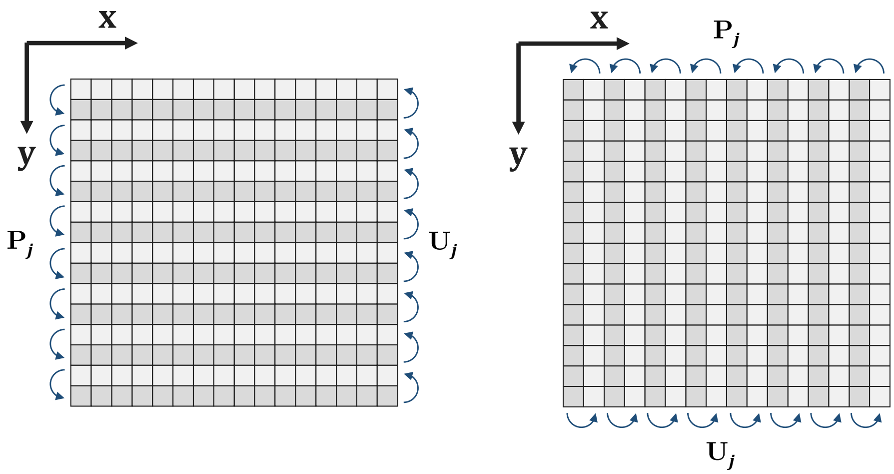
$$\Phi_d^{-1} = \left(D^{p_\mathrm{s}^2/\lambda d}\,\mathcal{F}\,D^{p_\mathrm{s}^2/\lambda d}\right)^{-1} = D^{-p_\mathrm{s}^2/\lambda d}\,\mathcal{F}^{-1}\,D^{-p_\mathrm{s}^2/\lambda d} = \Phi_{-d} = \overline{\Phi}_d \tag{12}$$

We can transform the signal for Φ_d with lifting using the same approach as for the convolutional case F_d. However, we still need to do a point-wise multiplication with a pure phase delay function $c\,D^{(p_\mathrm{d}^2 - p_\mathrm{s}^2)/\lambda d}$. For this, we can use the invertible integer lifting approximation of a Givens rotation [26]: since $\exp(i\varphi) = \cos(\varphi) + i\sin(\varphi)$, we apply

$$\begin{pmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{pmatrix} = \begin{pmatrix} 1 & \frac{\cos(\varphi)-1}{\sin(\varphi)} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ \sin(\varphi) & 1 \end{pmatrix} \begin{pmatrix} 1 & \frac{\cos(\varphi)-1}{\sin(\varphi)} \\ 0 & 1 \end{pmatrix} \tag{13}$$

on the real and imaginary components of every sample, given the specific needed phase delay φ at that position.

Since the Fresnel transform is separable, we apply the transform both along the rows and along the columns. For every pair of consecutive rows (or columns), the odd rows x_o will predict the even rows x_e, and the even rows will update the odd rows in the lifting procedure (see Fig. 3).



(a) (b)

Figure 3: Because Fresnel diffraction is separable, we can apply the transform once on the rows (a) and once on the columns (b). The lifting is applied independently on pairs of rows/columns as shown in the diagrams.

Table 1: Tested digital holograms and their properties. The "Form" column indicates which numerical Fresnel diffraction operator is needed. The last three columns indicate the wavelength (λ), pixel pitch (p) and used propagation distance (d).

Name	Database	Form	Resolution (pixels)	λ (nm)	p (µm)	d(mm)
Ball	Interfere-II	Conv.	8192×8192	633.0	1.0	140.0
Dragon	Interfere-II	Conv.	8192×8192	633.0	1.0	140.0
Venus	Interfere-II	Conv.	8192×8192	633.0	1.0	145.0
Dices 1	b-com repository	Conv.	7680×4320	640.0	4.8	11.40
Diffuse Car	b-com repository	Conv.	8192×8192	532.0	4.0	23.16
Astronauts	EmergImg-HoloGrail	Fourier	2588×1940	632.8	2.2	172.1
Dices 2	EmergImg-HoloGrail	Fourier	2588×1940	632.8	2.2	159.5
Skull	EmergImg-HoloGrail	Fourier	2588×1940	632.8	2.2	168.9

Experiments

For the experiments, we utilised 8 holograms taken from 3 different databases: The b-com hologram repository [27], the Interfere-II database [28] and the EmergImg-HoloGrail v2 database [22]. We reported all the main hologram parameters in Table 1, namely the resolution, wavelength, pixel pitch and chosen propagation distance. To obtain a uniform data representation for all holograms, both the real and imaginary channels were quantized to a dynamic range of 8 bits per channel (bpc).

We compressed the holograms with JPEG 2000 in lossless mode, using a 4-level Mallat CDF53 integer wavelet decomposition and 32×32-sized codeblocks. We tested two different configurations: (1) only JPEG 2000, and (2) JPEG 2000 after the application of the integer Fresnel transform. For the latter, we used 16-bit for the input channels because the integer Fresnel transform can modify the dynamic range

Table 2: Bit-rates of the lossless hologram compression, comparing default JPEG 2000 with the proposed integer Fresnel transform + JPEG 2000.

Hologram	JPEG 2000 (bpc)	Proposed (bpc)	Gain (bpc)
Ball	6.24	3.87	**2.37**
Dragon	5.72	2.89	**2.83**
Venus	5.84	3.12	**2.72**
Dices 1	4.76	4.64	**0.12**
Diffuse Car	6.81	5.35	**1.46**
Astronauts	5.86	5.47	**0.39**
Dices 2	6.16	5.92	**0.24**
Skull	5.99	5.45	**0.54**

of the integer values.

The resulting bit-rates for both configurations are shown in Table 2. We report bit-rate gains ranging from 0.12 up to 2.83 bpc. We observe that on average, the gains for the computer-generated holograms from the b-com hologram repository and the Interfere-II database are larger than the optically acquired ones from the EmergImg-HoloGrail v2 database. One possible reason for this difference could be due to phenomena such as speckle noise or slight optical aberrations. The transform is also less effective on the “Dice 1” hologram; this is likely because the hologram plane very close to the scene objects, so the scene will have much less defocus and spread signal to begin with.

The computational complexity of the proposed transform is mainly determined by the FFTs in the Fresnel transform, which has $\mathcal{O}(n \log n)$ complexity. The Fresnel transform is applied 3 times on half the data, so the total calculation cost will roughly be about 150% as much as the conventional floating-point base implementation.

Conclusion

We developed an integer variant of the Fresnel transform for the lossless encoding of digital holograms. By integrating the integer Fresnel transform with JPEG 2000, we obtain bit-rate savings ranging from 0.12 up to 2.83 bpc over the default lossless JPEG 2000 implementation. Avenues for future research include evaluating the integer Fresnel transform on a broader set of holograms (such as data for microscopic holography), combining it with more advanced transforms and codecs, and generalizing the lifting scheme to other linear canonical transforms.

Funding

The research leading to these results has received funding from the European Research Council under the European Union’s Seventh Framework Programme (FP7/2007-2013)/ERC Grant Agreement Nr. 617779 (INTERFERE), and from the Research Foundation - Flanders (FWO) project Nr. G024715N.

References

[1] Edward H. Adelson and James R. Bergen, “The Plenoptic Function and the Elements of Early Vision,” in *Computational Models of Visual Processing*. 1991, pp. 3–20, MIT Press.

[2] Etienne Cuche, Pierre Marquet, and Christian Depeursinge, “Simultaneous amplitude-contrast and quantitative phase-contrast microscopy by numerical reconstruction of fresnel off-axis holograms,” *Appl. Opt.*, vol. 38, no. 34, pp. 6994–7001, Dec 1999.

[3] Abbie E. Tippie James R. Fienup, “Gigapixel synthetic-aperture digital holography,” 2011.

[4] Florian Charrière, Jonas Kühn, Tristan Colomb, Frédéric Montfort, Etienne Cuche, Yves Emery, Kenneth Weible, Pierre Marquet, and Christian Depeursinge, “Characterization of microlenses by digital holographic microscopy,” *Appl. Opt.*, vol. 45, no. 5, pp. 829–835, Feb 2006.

[5] Hans J Coufal, Demetri Psaltis, Glenn T Sincerbox, et al., *Holographic data storage*, vol. 8, Springer, 2000.

[6] Fahri Yaraş, Hoonjong Kang, and Levent Onural, “State of the art in holographic displays: A survey,” *J. Display Technol.*, vol. 6, no. 10, pp. 443–454, Oct 2010.

[7] J. Y. Son, H. Lee, B. R. Lee, and K. H. Lee, “Holographic and light-field imaging as future 3-d displays,” *Proceedings of the IEEE*, vol. 105, no. 5, pp. 789–804, May 2017.

[8] Jason Geng, “Three-dimensional display technologies,” *Adv. Opt. Photon.*, vol. 5, no. 4, pp. 456–535, Dec 2013.

[9] Mark Lucente, “The first 20 years of holographic videoand the next 20,” June 2011.

[10] David Blinder, Ayyoub Ahar, Stijn Bettens, Tobias Birnbaum, Athanasia Symeonidou, Heidi Ottevaere, Colas Schretter, and Peter Schelkens, “Signal processing challenges for digital holographic video display systems,” *Signal Processing: Image Communication*, 2018.

[11] M. Liebling, T. Blu, and M. Unser, “Fresnelets: New multiresolution wavelet bases for digital holography,” *Trans. Img. Proc.*, vol. 12, no. 1, pp. 29–43, Jan. 2003.

[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,” *Opt. Express*, vol. 19, no. 9, pp. 8019–8031, Apr 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*, vol. 53, no. 12, pp. 123102, 2014.

[14] Yafei Xing, Mounir Kaaniche, Batrice Pesquet-Popescu, and Frdric Dufaux, “Adaptive nonseparable vector lifting scheme for digital holographic data compression,” *Appl. Opt.*, vol. 54, no. 1, pp. A98–A109, Jan 2015.

[15] Tobias Birnbaum, David Blinder, Colas Schretter, and Peter Schelkens, “Compressing macroscopic near-field digital holograms with wave atoms,” in *Imaging and Applied Optics 2018*. 2018, p. DW2F.5, Optical Society of America.

[16] J. P. Peixeiro, C. Brites, J. Ascenso, and F. Pereira, “Holographic data coding: Benchmarking and extending hevc with adapted transforms,” *IEEE Transactions on Multimedia*, vol. 20, no. 2, pp. 282–297, Feb 2018.

[17] Anas El Rhammad, Patrick Gioia, Antonin Gilles, Arco Cagnazzo, and Beatrice Pesquet-Popescu, “Color digital hologram compression based on matching pursuit,” *Applied Optics*, vol. 57, no. 17, pp. 4930–4942, June 2018.

[18] 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*, vol. 4, no. 2, pp. 206–218, June 2018.

[19] Emmanouil Darakis and John J. Soraghan, "Reconstruction domain compression of phase-shifting digital holograms," *Appl. Opt.*, vol. 46, no. 3, pp. 351–356, Jan 2007.

[20] Marco V. Bernardo, Pedro Fernandes, Angelo Arrifano, Marc Antonini, Elsa Fonseca, Paulo T. Fiadeiro, Antnio M.G. Pinheiro, and Manuela Pereira, "Holographic representation: Hologram plane vs. object plane," *Signal Processing: Image Communication*, vol. 68, pp. 193–206, Oct. 2018.

[21] Ting-Chung Poon, *Digital holography and three-dimensional display: Principles and Applications*, Springer Science & Business Media, 2006.

[22] "Emergimg-holograil holographic database," `http://emergimg.di.ubi.pt/downloads.html`.

[23] Joseph W. Goodman, *Introduction to Fourier Optics*, Roberts & Company, 2004.

[24] Tomoyoshi Shimobaba, Jiantong Weng, Takahiro Sakurai, Naohisa Okada, Takashi Nishitsuji, Naoki Takada, Atsushi Shiraki, Nobuyuki Masuda, and Tomoyoshi Ito, "Computational wave optics library for c++: Cwo++ library," *Computer Physics Communications*, vol. 183, no. 5, pp. 1124 – 1138, 2012.

[25] Wim Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," *Applied and Computational Harmonic Analysis*, vol. 3, no. 2, pp. 186 – 200, 1996.

[26] R. Geiger, Y. Yokotani, G. Schuller, and J. Herre, "Improved integer transforms using multi-dimensional lifting," in *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, May 2004, vol. 2, pp. ii–1005.

[27] "b-com hologram repository," `https://hologram-repository.labs.b-com.com/#/hologram-repository`.

[28] D. Blinder, A. Ahar, A. Symeonidou, Y. Xing, T. Bruylants, C. Schretter, B. Pesquet-Popescu, F. Dufaux, A. Munteanu, and P. Schelkens, "Open access database for experimental validations of holographic compression engines," in *2015 Seventh International Workshop on Quality of Multimedia Experience (QoMEX)*, May 2015, pp. 1–6, `http://erc-interfere.eu/downloads.html`.