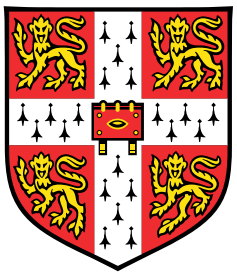
In Search For The Optimal Phase-Only Computer-Generated Hologram



Jinze Sha

Department of Engineering
University of Cambridge

This dissertation is submitted for the degree of
Doctor of Philosophy

King's College

January 2025

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Jinze Sha

January 2025

Acknowledgements

And I would like to acknowledge ...

@TODO

Why did the photon check into a hotel? Because it needed to travel light!

Abstract

@TODO

List of Publications

[1] Jana Skirnewskaja, Yunuen Montelongo, Jinze Sha, and Timothy D. Wilkinson. Holographic lidar projections with brightness control. In *Imaging and Applied Optics Congress 2022 (3D, AOA, COSI, ISA, pcAOP)*, page 3F2A.6. Optica Publishing Group, 2022

[2] Jinze Sha, Andrew Kadis, Fan Yang, and Timothy D. Wilkinson. Limited-memory bfgs optimisation of phase-only computer-generated hologram for fraunhofer diffraction. In *Digital Holography and 3-D Imaging 2022*, page W3A.3. Optica Publishing Group, 2022

[3] Andrew Kadis, Benjamin Wetherfield, Jinze Sha, Fan Yang, Youchao Wang, and Timothy D. Wilkinson. Effect of bit-depth in stochastic gradient descent performance for phase-only computer-generated holography displays. *London Imaging Meeting*, 3:36–40, 7 2022

[4] Jinze Sha, Andrew Kadis, Fan Yang, Youchao Wang, and Timothy D. Wilkinson. Multi-depth phase-only hologram optimization using the l-bfgs algorithm with sequential slicing. *J. Opt. Soc. Am. A*, 40(4):B25–B32, Apr 2023

[5] Jinze Sha, Adam Goldney, Andrew Kadis, Jana Skirnewskaja, and Timothy D. Wilkinson. Digital pre-distorted one-step phase retrieval algorithm for real-time hologram generation for holographic displays. *Journal of Imaging Science and Technology*, 67(3):030405–1–030405–1, 2023

[6] Jana Skirnewskaja, Yunuen Montelongo, Jinze Sha, Phil Wilkes, and Timothy D. Wilkinson. Accelerated augmented reality holographic 4k video projections based on lidar point clouds for automotive head-up displays. *Advanced Optical Materials*, 12(12):2301772, 2024

[7] Roubing Meng, Jinze Sha, Zhongling Huang, and Timothy D. Wilkinson. Extending FOV of holographic display with alternating lasers. In Peter Schelkens and Tomasz Kozacki,

editors, *Optics, Photonics, and Digital Technologies for Imaging Applications VIII*, volume 12998, page 129981J. International Society for Optics and Photonics, SPIE, 2024

[8] Jinze Sha, Andrew Kadis, Benjamin Wetherfield, Roubing Meng, Zhongling Huang, Dilawer Singh, Antoni Wojcik, and Timothy D. Wilkinson. Information capacity of phase-only computer-generated holograms for holographic displays. In Peter Schelkens and Tomasz Kozacki, editors, *Optics, Photonics, and Digital Technologies for Imaging Applications VIII*, volume 12998, page 129980J. International Society for Optics and Photonics, SPIE, 2024

[9] Jinze Sha, Antoni Wojcik, Benjamin Wetherfield, Jianghan Yu, and Timothy D. Wilkinson. Multi frame holograms batched optimization for binary phase spatial light modulators. *Scientific Reports*, 14(1):19380, Aug 2024

Table of contents

List of figures

List of tables

Chapter 1

Introduction

The pursuit of three-dimensional (3D) displays has continued at a pace over the past 20 years. Currently, most commercially available '3D display' products such as 3D cinema, 3D TV, handheld 3D devices (e.g. Nintendo 3DS, HTC Evo 3D) and Virtual Reality (VR) and Augmented Reality (AR) head sets are in fact stereoscopic displays [20] where two different two-dimensional (2D) images are displayed to the left and right eyes respectively, creating a 3D illusion in the brain. Despite its high image quality, the major issue with stereoscopic displays is that they cannot provide real optical defocusing effect in depth [21]. Modern 3D cinemas are able to provide good comfort because polarisation glasses are as light as regular glasses, and the variable defocusing issue can be avoided by the combinations of good design of point of interest in each scene and the according defocusing effect as captured by the camera, so most audience won't experience much discomfort for around 2 to 3 hours, making 3D films accepted by the general public [22]. However, the content, viewing angle and depth of focus of 3D films are fixed after they are captured. To provide an interactive and real-time rendered immersive experience, VR/AR headsets have frequently been advertised as the 'gateway to the metaverse' in recent years [23]. However, personal experiences with VR headsets are far from comfortable, not only because of their heavy weight, but also because the display is physically at a very close distance, while the brain thinks the objects are at various distances and yet are still all in focus, which is very unnatural, because in real life, when the eye is focused on a near object, the far background blurs out. And also, the two displays in the VR headset need to be rendered in real-time based on the location and viewing angle of the user, the delays in rendering often causes dizziness and sickness. Hence, the

heavy weight, the lack of real depth of focus and the delay in rendering collectively often causes discomfort, dizziness and sickness of VR headsets users [24, 25].

In comparison, holography techniques can produce a full 3D light field, which does not rely on any head mounted device, has true depth of focus, and does not need to be re-rendered according to change in viewer position and viewing angle.

Fig. 1.1 A photo of the holographic portrait of Dennis Gabor [10]

Holography, taking its name from the Greek word $o\lambda o\sigma$ (holos), meaning *whole*, was first introduced in 1948 by Dennis Gabor [26], originally named as *wavefront reconstruction* [27]. It is a technology which generates 3D images via the diffraction of light. Similar to 2D photography, the earliest holography used a piece of film to record the diffraction pattern, which was then used to reconstruct the 3D field, as shown in Fig. 1.1 which is a holographic recording of Dennis Gabor himself. After the invention of digital cameras, digital holography emerged. The limitation of both methods is that they require a physical object priori to record the hologram. In order to generate hologram for objects that do not physically exist, computer-generated holography (CGH) emerged where a hologram can be calculated through various algorithmic approaches and then displayed on a spatial light modulator (SLM) modulating the wavefront of a coherent light source in order to produce 3D reconstructions. Currently available SLM's can only modulate either phase or amplitude, so algorithms are needed to compute amplitude-only or phase-only holograms. The classic phase-retrieval algorithms include direct binary search [28], simulated annealing [29] and Gerchberg-Saxton [30]. With the developments in modern numerical optimisation methods and increases in computational power, phase retrieval using numerical optimisation methods has also been found in the literature, such as the gradient descent [31, 32] and its stochastic

variations [33, 34, 3]. However, all the existing phase retrieval methods still have some fundamental issues, mainly their poor image quality and/or the heavy computation required, the solutions of which are the ultimate goals of this research.

This thesis therefore explores the development and optimisation of phase-only CGHs for holographic displays. The research investigates various phase retrieval algorithms and proposes novel methods to enhance the reconstruction quality and computational efficiency of CGHs, and then investigates the fundamental limits of discretised phase holograms from an information theory point of view. The thesis is structured as following.

Chapter 2 provides a comprehensive literature review, covering the fundamental theories of light, the principles of holography, and the evolution of CGH methods. The chapter reviews various phase retrieval algorithms, including the Direct Binary Search (DBS), Simulated Annealing (SA), Gerchberg-Saxton (GS), One-Step Phase Retrieval (OSPR), and Adaptive One-Step Phase Retrieval (AD-OSPR) algorithms and their adaptations for 3D CGHs, emphasizing the limitations of current algorithms and introducing the motivation for pursuing more advanced CGH techniques.

Chapter 3 proposes the Digital Pre-Distorted One-Step Phase Retrieval (DPD-OSPR) method. By experimentally evaluating the non-linearities in the holographic projection system and applying a digital pre-distortion (DPD) curve, significant improvements in reconstruction quality and reductions in mean squared error are demonstrated.

Chapter 4 introduces the optimisation of phase-only holograms using the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) optimisation algorithm, and then proposes the novel Target Image Phase Optimisation (TIPO) technique, which optimises the phase of the target image instead of the phase of the hologram. Then the multi-depth phase-only hologram optimisation for 3D targets is investigated, and a novel technique called Sequential Slicing (SS) is proposed, which evaluates the loss for a single slice of the 3D target at each iteration instead of a full evaluation on all slices, reducing computational time while maintaining overall quality and minimizing quality imbalances across all slices.

Chapter 5 extends the optimisation method on generating time-multiplexing binary-phase holograms, and proposes the novel Multi-Frame Holograms Batched Optimisation (MFHBO) technique. By optimising a batch of holograms for time multiplexing, which leverages the finite response time of human vision to average out noise, resulting in improved visual quality.

The MFHBO algorithm shows a much better quality than the existing time-multiplexing hologram generation methods such as OSPR and AD-OSPR.

Chapter 6 investigates the information capacity of phase-only CGH. This chapter examines the effects of quantisation on hologram bit depth and their impact on reconstruction quality, and looks for the correlation between the entropy of the target image and the reconstruction error, providing insights into the fundamental limits of discretised CGH.

Chapter 7 marks the conclusion and lists potential further work that could be done.

Chapter 2

Background Theory

This chapter lays down the fundamental theories of optoelectronics and search algorithms, which are essential to the research outlined in the later chapters of this thesis.

2.1 The Nature of Light

2.1.1 Wave-Particle Duality

The problem of how light propagates has been troubling scientists for centuries. In the 17th century, Sir Isaac Newton made a significant step. He proposed that light consists of particles, or 'corpuscles', with a mass varying with colour, which explained phenomena such as reflection and refraction [35]. In contrast, Christiaan Huygens, a contemporary of Newton, demonstrated that light behaves as a wave, as it is capable of diffraction, which is the bending of light around the edge of an object, leading to the non-sharp edges of shadows [36].

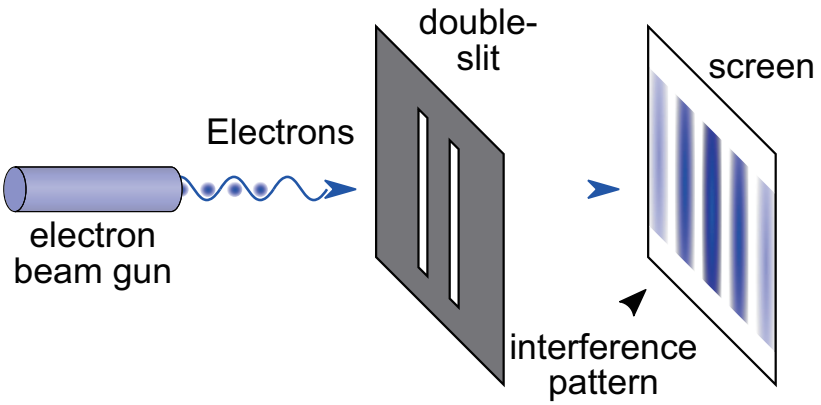


Fig. 2.1 An illustration of the Young's Double-slit experiment [11]

The wave theory gained significant support in the early 19th century through the experiments by Thomas Young. Young's double-slit experiment in 1801 provided clear evidence of the wave nature of light by showing that light passing through two slits creates an interference pattern on a screen [37], as illustrated in Fig. 2.1. Augustin-Jean Fresnel further advanced the wave theory by developing a comprehensive mathematical framework to describe light as a wave, explaining phenomena such as polarization and the diffraction of light [38].

This understanding was further reinforced by James Clerk Maxwell in the mid-19th century. In 1864, James Clerk Maxwell organised a set of four equations describing the space and time dependence of the electromagnetic field, which are:

$$\nabla \times \mathbb{E} = -\frac{\partial \mathbb{B}}{\partial t} \tag{2.1}$$

$$\nabla \times \mathbb{H} = \mathbb{J} + \frac{\partial \mathbb{D}}{\partial t} \tag{2.2}$$

$$\nabla \cdot \mathbb{D} = \rho \tag{2.3}$$

$$\nabla \cdot \mathbb{B} = 0 \tag{2.4}$$

where $\mathbb{D}$ is the electric flux density, $\mathbb{E}$ is the electric field intensity, $\mathbb{B}$ is the magnetic flux density, $\mathbb{H}$ is the magnetic field intensity, ρ is the volume charge density, and $\mathbb{J}$ is the current density [39].

And the relation between $\mathbb{D}$ and $\mathbb{E}$ and between $\mathbb{B}$ and $\mathbb{H}$ for linear materials are:

$$\mathbb{B} = \mu\mathbb{H} \tag{2.5}$$

$$\mathbb{D} = \varepsilon\mathbb{E} \tag{2.6}$$

where μ is the magnetic permeability and ε is the dielectric permittivity of the material [40].

Maxwell's equations unified electricity and magnetism into a single theory of electromagnetism, predicting that light is an electromagnetic wave that propagates through space [41].

Despite the success of the wave theory, it could not explain all light-related phenomena. The early 20th century brought a pivotal development with Albert Einstein's explanation of the photoelectric effect. In 1905, Einstein proposed that light also behaves as particles, or 'quanta' (later called photons), which could eject electrons from a metal surface when light is shone upon it [42]. This particle nature of light was critical in explaining observations that wave theory alone could not address and earned Einstein the Nobel Prize in Physics in 1921.

These discoveries collectively revealed that light exhibits both wave and particle properties, depending on the experimental context. This wave-particle duality became a cornerstone of quantum mechanics, fundamentally altering human's understanding of the nature of light. Although to date, it is still not yet known what light exactly is, it is now known how light behaves.

2.1.2 Wave Equation

To mathematically describe the propagation of light in free space (i.e. in absence of free charge), the Maxwell equations in Eq. (2.1) - Eq. (2.4) can be simplified as:

$$\nabla \times \mathbb{E} = -\mu \frac{\partial \mathbb{H}}{\partial t} \tag{2.7}$$

$$\nabla \times \mathbb{H} = \varepsilon \frac{\partial \mathbb{E}}{\partial t} \tag{2.8}$$

$$\nabla \cdot \varepsilon\mathbb{E} = 0 \tag{2.9}$$

$$\nabla \cdot \mu\mathbb{H} = 0 \tag{2.10}$$

Taking the curl on both the left and right hand sides of Eq. (2.7), and using the vector identity of $\nabla \times (\nabla \times \mathbf{u}) = \nabla(\nabla \cdot \mathbf{u}) - \nabla^2 \mathbf{u}$, we get:

$$\nabla \times (\nabla \times \mathbb{E}) = -\nabla \times (\mu \frac{\partial \mathbb{H}}{\partial t}) \tag{2.11}$$

$$\nabla(\nabla \cdot \mathbb{E}) - \nabla^2 \mathbb{E} = -\frac{\partial}{\partial t} \nabla \times (\mu \mathbb{H}) \tag{2.12}$$

Then, by substituting Eq. (2.8) and Eq. (2.9) in, Eq. (2.12) becomes:

$$-\nabla^2 \mathbb{E} = -\frac{\partial}{\partial t}(\mu \varepsilon \frac{\partial \mathbb{E}}{\partial t}) \tag{2.13}$$

Hence, we have a generic form of wave equation, relating the space and time domain relation of electromagnetic waves propagating in free space:

$$\nabla^2 \mathbb{E} = \mu \varepsilon \frac{\partial^2 \mathbb{E}}{\partial t^2} \tag{2.14}$$

A valid solution to Eq. (2.14) is:

$$\mathbb{E} = \mathbb{E}_0 e^{j(\omega t - kr)} \tag{2.15}$$

where ω is the angular velocity of the wave, t is time, r is the propagation distance and k is called the wave number ($k = \frac{2\pi}{\lambda}$, where λ is the wavelength). From Eq. (2.15) we can see that the propagation of light in free space is essentially a phase shift. This suggests that, if we have a coherent light source and a device to manipulate light (called SLM, further explained in Section 2.2.3), we can produce an interference pattern reconstructing the target field we desire, and such method is called holographic projection.

2.2 Fundamentals of Holography

Holography is a technology that can fully reconstruct the wavefront of 3D objects, which is usually achieved by modulating a coherent light source. This section explains what a coherent light source is and how it is modulated and diffracted.

2.2.1 Light Source

The mechanism of holographic projection is to control the propagation of light in a way that, after diffraction, reconstructs a wavefront that matches the target field. We usually prefer to start from a coherent light source rather than a random one which will be a lot more difficult or even impossible to analyse and predict the interference pattern.

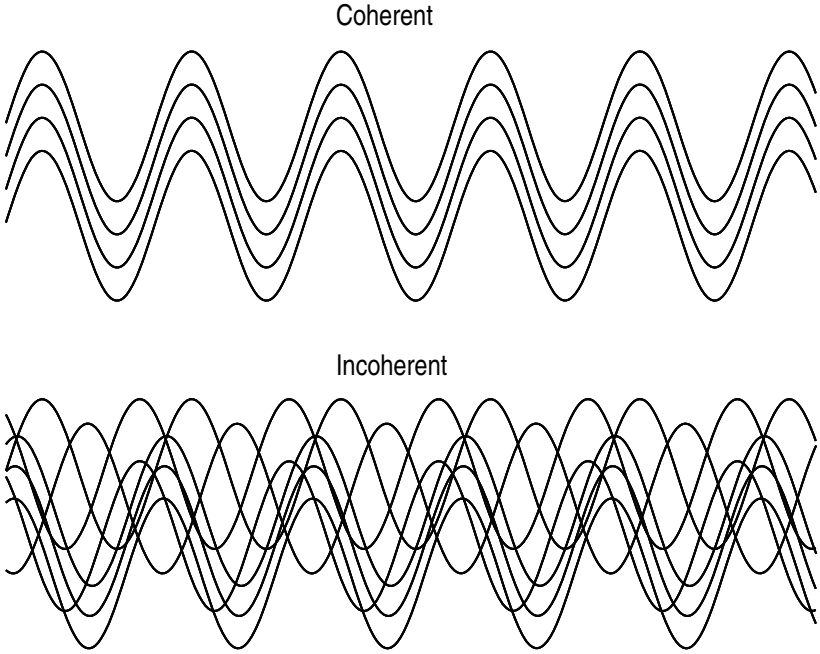


Fig. 2.2 Coherent v.s. incoherent light

The coherence of light refers to the property of light waves where the phase relationship between the waves is consistent over time and space, corresponding to temporal and spatial coherence:

- **Temporal coherence:** Temporal coherence describes the correlation between the phases of a light wave at different points along its propagation direction. It indicates how monochromatic (i.e. single-frequency) a light source is.

- **Spatial coherence:** Spatial coherence describes the correlation between the phases of a light wave at different points across the wavefront, perpendicular to the direction of propagation. It indicates the uniformity of the phase across the wavefront, as illustrated in Fig. 2.2. High spatial coherence means that the light waves across different points on the wavefront are in phase.

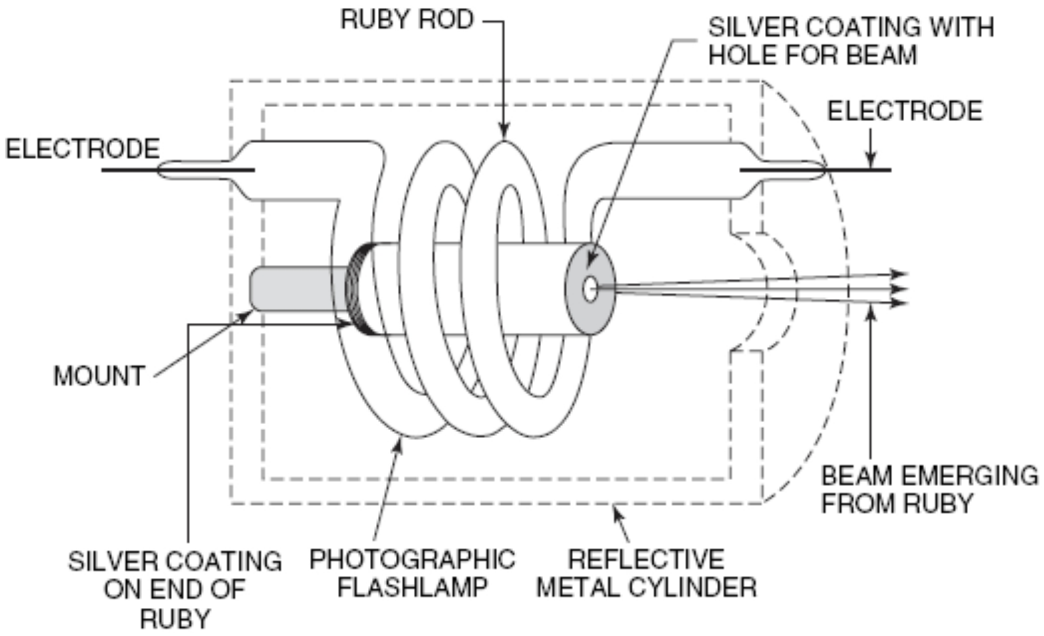


Fig. 2.3 Structure of the first laser [12]

The most common coherent visible light source is Laser, which stands for ***L**ight **A**mplification by the **S**timulated **E**mission of **R**adiation*. It was first invented by Theodore Maiman in 1959, with the structure shown in Fig. 2.3 [12, 43, 44]. It differs from other sources of light in that it emits coherent light, which is suitable for holographic projection. However, the coherent and monochromatic property of laser also has a side effect of speckle noise in the reconstructed image [45], which is one of the major problems affecting the image quality of holographic projections and has seen lots of efforts to cope with it in the literature [46–49].

2.2.2 Diffraction

This section delves into how light interacts with apertures, leading to diffractions. Understanding diffraction is essential for holography, as it explains how light can be manipulated to reconstruct three-dimensional light fields. The principles of diffraction and interference

underpin the essential process of holographic projection, making it possible to accurately recreate complex wavefronts and achieve true 3D visualization.

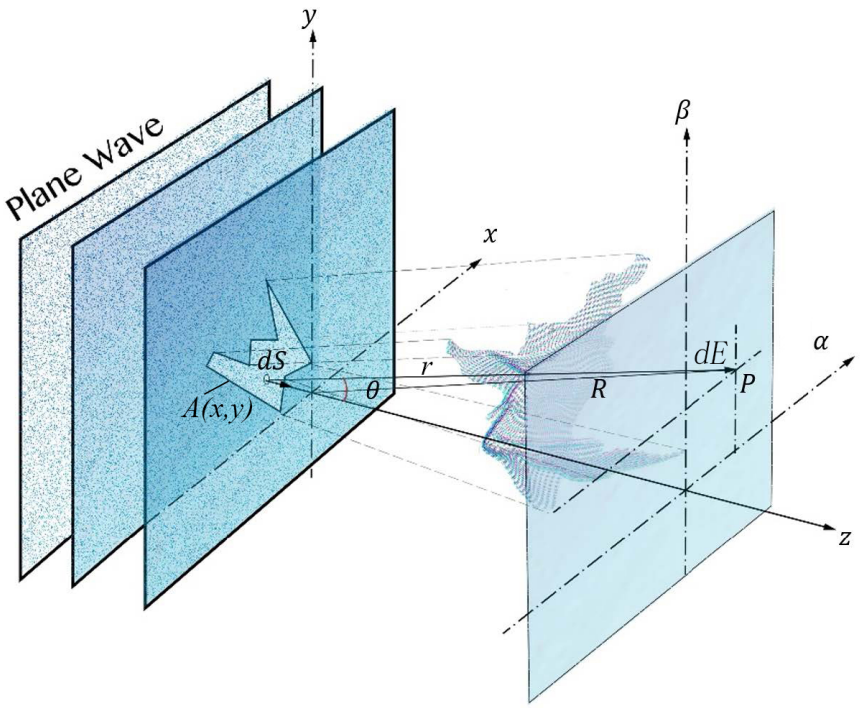


Fig. 2.4 Diffraction geometry

To model how light diffracts through a 2D aperture, we first set up a coordinate system as shown in Fig. 2.4, where the aperture is denoted by $A(x,y)$ and the diffracted field is denoted by $E(\alpha,\beta,z)$. R defines the distance between point P and the origin of the aperture ($(x,y) = (0,0)$), r defines the distance between point P and a point on the aperture, and θ defines the angle r from the z-axis. Then by trigonometry we can have the following identities:

$$cos(\theta) = \frac{z}{r} \tag{2.16}$$

$$R^2 = \alpha^2 + \beta^2 + z^2 \tag{2.17}$$

$$r^2 = (\alpha - x)^2 + (\beta - y)^2 + z^2 \tag{2.18}$$

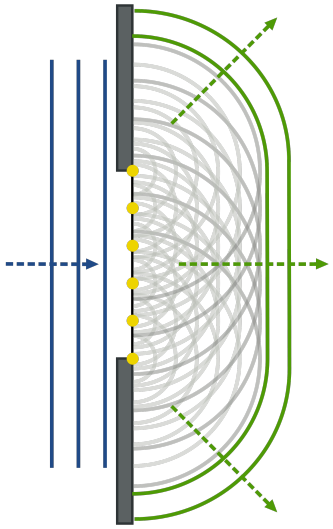


Fig. 2.5 Huygens-Fresnel wavelet principle [13]

The Huygens-Fresnel principle states that every point on a wavefront is itself the source of outgoing secondary spherical wavelets, which can be expressed mathematically as follows when $r \gg \lambda$ [50]:

$$E(\alpha,\beta,z) = \frac{1}{j\lambda}\iint A(x,y)\frac{e^{jkr}}{r}cos(\theta)dxdy \tag{2.19}$$

Applying the identities in Eq. (2.16) - Eq. (2.18), Eq. (2.19) becomes:

$$E(\alpha,\beta,z) = \frac{z}{j\lambda}\iint A(x,y)\frac{e^{jkr}}{r^2}dxdy \tag{2.20}$$

$$= \frac{z}{j\lambda}\iint A(x,y)\frac{e^{jk\sqrt{(\alpha-x)^2+(\beta-y)^2+z^2}}}{(\alpha-x)^2+(\beta-y)^2+z^2}dxdy \tag{2.21}$$

Unfortunately, Eq. (2.21) cannot be solved analytically except for few specific aperture functions $A(x,y)$, so we have to make some approximations in order to solve for arbitrary $A(x,y)$, the common methods are *Fresnel* and *Fraunhofer* approximations for regions depicted in Fig. 2.6.

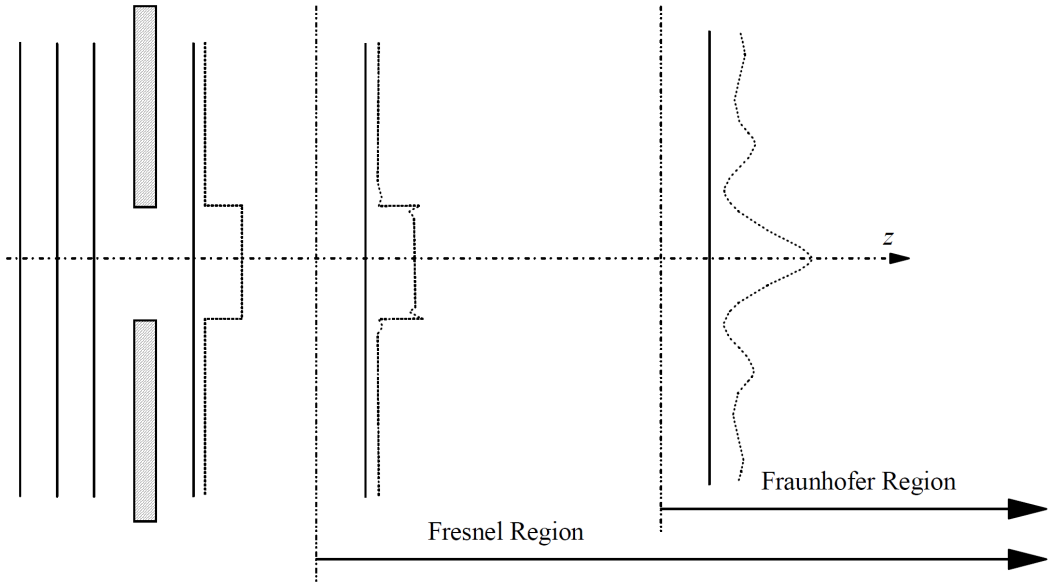


Fig. 2.6 Fresnel and Fraunhofer region [14]

Fresnel Approximation

$$\sqrt{1+d} = 1 + \frac{1}{2}d - \frac{1}{8}d^2 + \cdots \tag{2.22}$$

Fresnel approximation replaces expressions for spherical waves by quadratic-phase exponentials, using the binomial expansion of the square root (given in Eq. (2.22)) to approximate r in Eq. (2.20) [50].

Retaining only the first two terms of the expansion gives:

$$r = \sqrt{(\alpha - x)^2 + (\beta - y)^2 + z^2} \tag{2.23}$$

$$= z\sqrt{1 + \left(\frac{\alpha - x}{z}\right)^2 + \left(\frac{\beta - y}{z}\right)^2} \tag{2.24}$$

$$\approx z\left[1 + \frac{1}{2}\left(\frac{\alpha - x}{z}\right)^2 + \frac{1}{2}\left(\frac{\beta - y}{z}\right)^2\right] \tag{2.25}$$

For the r^2 in the denominator of Eq. (2.20), the error introduced by dropping all terms but z is generally acceptably small (i.e. $r^2 \approx z^2$), and for the r appearing in the exponent in the numerator of Eq. (2.20), errors are much more critical [50]. So, by substituting Eq. (2.25) for

the r in the numerator of Eq. (2.20) and substituting z for the r in the denominator, we have:

$$E(\alpha,\beta,z) \approx \frac{z}{j\lambda}\iint A(x,y)\frac{e^{jkz\left[1+\frac{1}{2}\left(\frac{\alpha-x}{z}\right)^2+\frac{1}{2}\left(\frac{\beta-y}{z}\right)^2\right]}}{z^2}dxdy \tag{2.26}$$

$$= \frac{e^{jkz}}{j\lambda z}e^{j\frac{k}{2z}(\alpha^2+\beta^2)}\iint\left\{A(x,y)e^{j\frac{k}{2z}(x^2+y^2)}\right\}e^{-j\frac{2\pi}{\lambda z}(\alpha x+\beta y)}dxdy \tag{2.27}$$

$$= \frac{e^{jkz}}{j\lambda z}e^{j\frac{k}{2z}(\alpha^2+\beta^2)}\mathscr{F}\left\{A(x,y)e^{j\frac{k}{2z}(x^2+y^2)}\right\} \tag{2.28}$$

where $\mathscr{F}$ denotes the Fourier Transform, implemented on computers using the Fast Fourier Transform (FFT) function. Such method of including Fourier Transform (FT) in the study of optics is also named 'Fourier Optics'.

Now we have a more simple and solvable expression than Eq. (2.21). And also, as we are only interested in the scaling of relative points at P with respect to each other, so it is safe to normalise the multiplier term before the Fourier Transform to 1 [14]. So we can express the diffraction pattern in Fresnel region as:

$$E_{Fresnel\ region}(\alpha,\beta,z) = \mathscr{F}\left\{A(x,y)e^{j\frac{k}{2z}(x^2+y^2)}\right\} \tag{2.29}$$

Fraunhofer Approximation

Fraunhofer diffraction is a form of diffraction in which the distance between the light source and the receiving screen are in effect at infinite, so that the wave fronts can be treated as planar rather than spherical [39]. Fraunhofer approximation is very stringent, it assumes that the distance between the light source and the receiving screen are in effect at infinite:

$$z \gg \frac{k(x^2+y^2)_{max}}{2} \tag{2.30}$$

so that the wave fronts can be treated as planar rather than spherical [39], then the $e^{j\frac{k}{2z}(x^2+y^2)}$ term tends to 1, and Eq. (2.29) becomes:

$$E_{Fraunhofer\ region}(\alpha,\beta) = \mathscr{F}\{A(x,y)\} \tag{2.31}$$

which suggests that the far field pattern is simply the Fourier Transform of the aperture function.

2.2.3 Spatial Light Modulator (SLM)

SLMs are critical components in computer-generated holography (CGH). An SLM is a device used to control the amplitude or phase of light waves in a spatially varying manner. SLMs typically consist of a two-dimensional (2D) array of pixels, each of which can modulate the light either passing through or reflected from it. These pixels are usually addressed by electronic signals, allowing precise manipulation of the light wavefront. The modulation can be achieved through various mechanisms, such as liquid crystal (LC) SLMs, magneto-optic SLMs, deformable mirror SLMs, multiple-quantum-well SLMs, or acousto-optic Bragg cells [50].

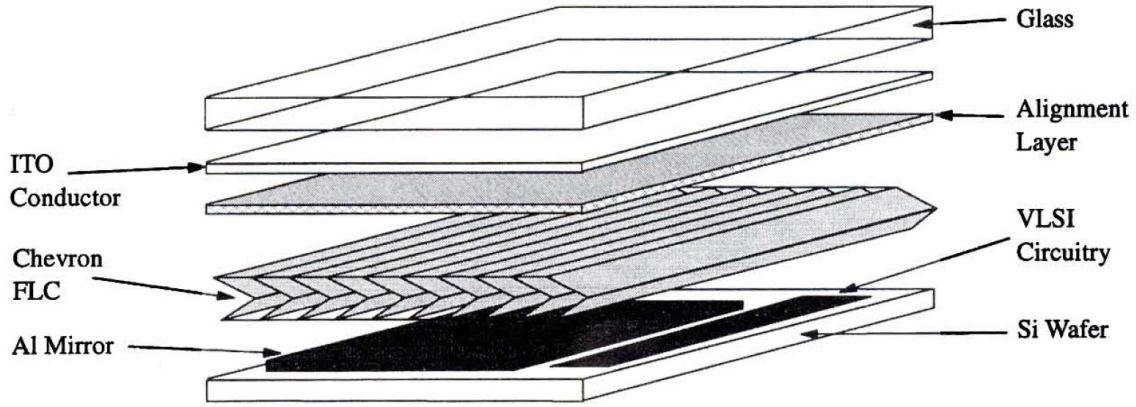


Fig. 2.7 A single FLC backplane SLM pixel structure [15]

The typical structure of a Ferroelectric Liquid Crystal (FLC) SLM pixel is shown in Fig. 2.7. The horizontal dimension of a single pixel, which is called the pixel pitch, is on scale of a few μm. LCs are anisotropic materials, which have different refraction index in different axis. And the modulation is achieved by controlling the tilt of the angle of the LCs. For the FLCs, the tilt angle is controlled by the electric field across them. Therefore the FLCs are sandwiched between the Indium Tin Oxide (ITO) conductor and the aluminium (Al) mirror, aside which is a Very Large Scale Integration (VLSI) circuitry to control the voltage and the electric field across the FLCs, hence modulates the phase of the incident light. Aligning millions of such pixels in a grid produces an SLM which can spatially modulate the wavefront of the incident light.

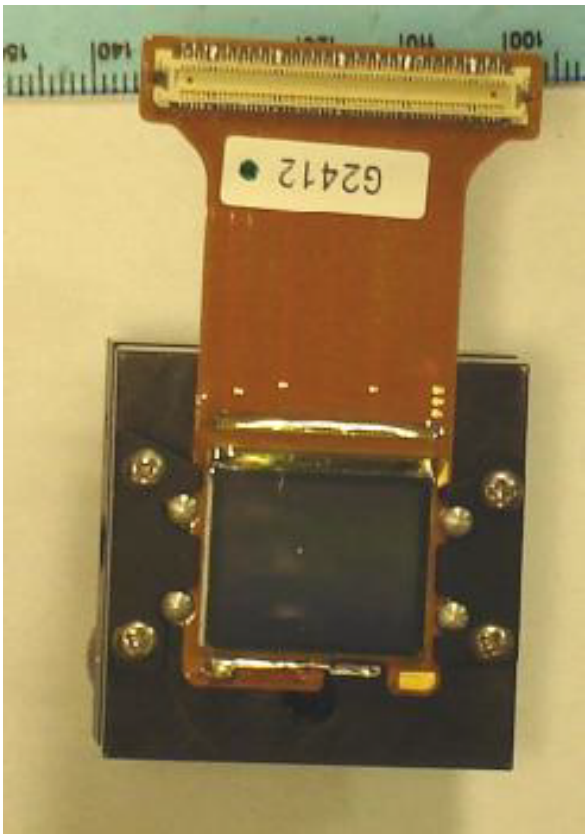


Fig. 2.8 Photo of the FLC SLM used in this research [16]

The SLM used in this research, as shown in Fig. 2.8, is a binary phase SXGA-R2 ForthDD FLC SLM with a refresh rate of 1440Hz, a pixel pitch of 13.6 µm and a resolution of $1280px \times 1024px$.

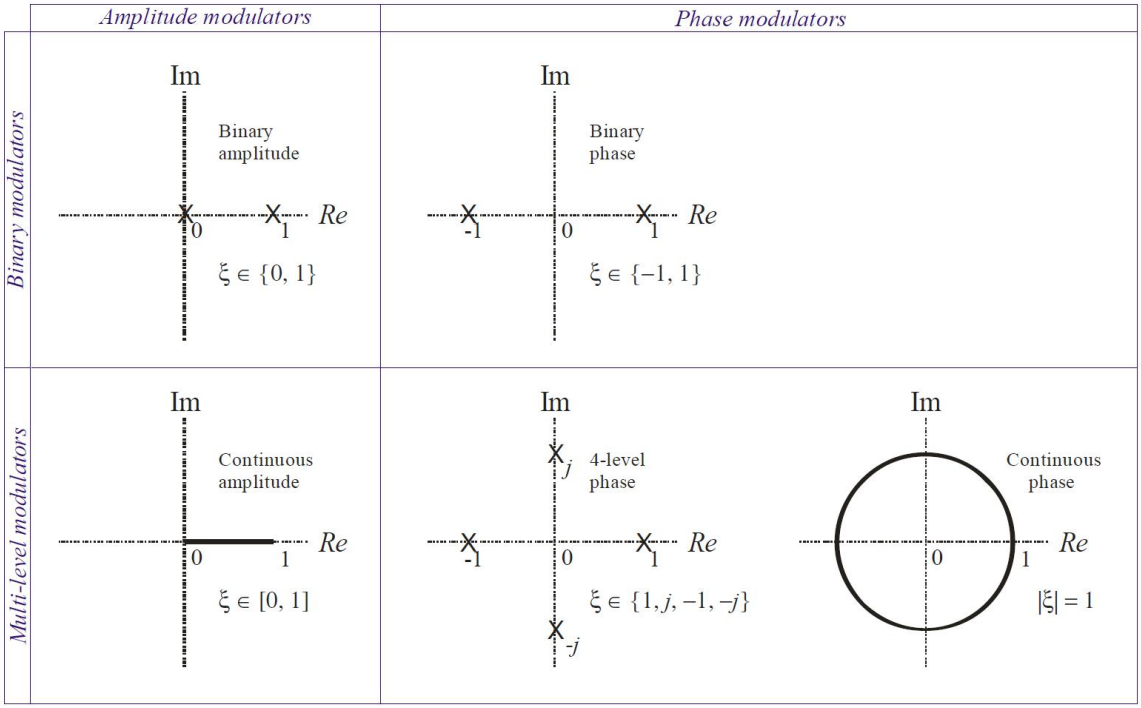


Fig. 2.9 Modulation schemes of the SLMs [17]

Currently available SLMs have one of the four modulation schemes, as illustrated in Fig. 2.9 [17], which are:

- **Multi-level Amplitude** modulators can modulate each pixel from zero transmission (0) to full transmission (1), either continuously or in discrete steps. (e.g. nematic liquid crystal display [51], found for example in laptops and many conventional video projectors)
- **Binary Amplitude** modulators can switch each pixel to zero transmission (0) or full transmission (1), but nothing in between. (e.g. deformable mirror device [52], ferroelectric liquid crystal display [53], both used in high-end video projectors)
- **Multi-level Phase** modulators can modulate the phase shift imparted by each pixel from 0 to 2π radians, either continuously or in discrete steps. (e.g. Nematic liquid crystal devices [54])
- **Binary Phase** modulators can switch each pixel for a phase shift of either 0 or π radians. (e.g. Ferroelectric liquid crystal displays [55])

Among the four modulation schemes, phase modulations are of higher interest for the purpose of holography, because amplitude modulators, either multi-level or binary, block light at the SLM, causing waste of energy, leading to poorer energy efficiency. And also, amplitude modulation always has a zero-order (forming a central bright spot), because the average amplitude is always between 0 and 1; on the contrary, phase modulation can suppress the zero-order by designing the hologram to have zero average.

As there is no fully complex modulator available yet, we need algorithms to generate phase-only holograms, and such process is called phase retrieval. There are currently many algorithms for such purpose, which will be discussed in Section 2.3.

Rotational symmetries in the binary phase modulation

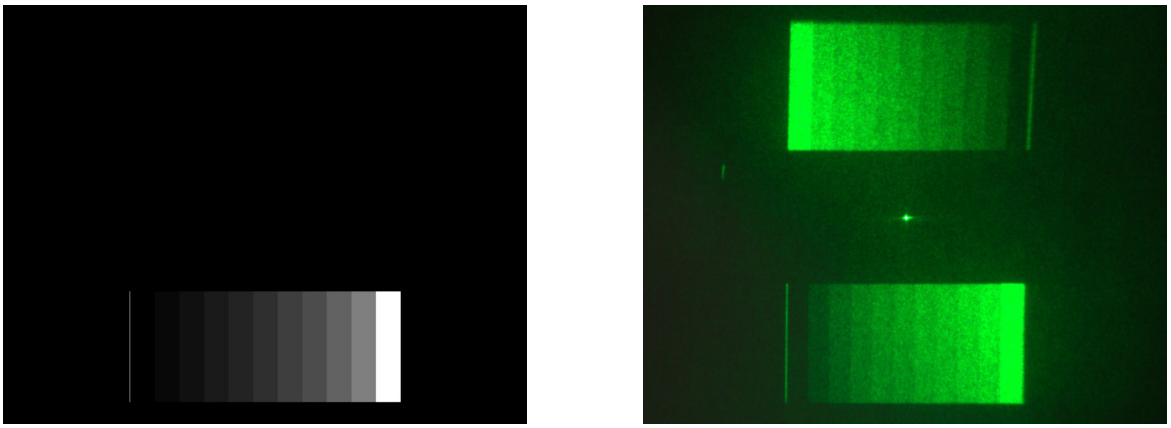
The spatial light modulator (SLM) used in this thesis is a binary phase modulator. As the binary phase modulation is purely real (i.e. it's only switching between 0° and 180°, corresponding to 1 and -1 values of $A^*(x,y)$), the complex conjugate $A^*(x,y)$ is the same as $A(x,y)$:

$$A^*(x,y) = A(x,y) \tag{2.32}$$

because the Fourier transform of $A^*(x,y)$ is the same as the Fourier transform of $A(x,y)$

$$E(-\alpha,-\beta) = \mathscr{F}[A^*(x,y)] = \mathscr{F}[A(x,y)] = E(\alpha,\beta) \tag{2.33}$$

So, at the Fraunhofer region, there is no distinction between the desired image and its 180° rotation in the replay field, causing a symmetrical conjugate image rotated 180° from the target image.



(a) Target image (b) Projection result

Fig. 2.10 Rotational symmetry in the projection result using the binary phase SLM

To demonstrate such phenomena, the example target image shown in Fig. 2.10a generated the binary-phase hologram using the algorithm called one-step phase-retrieval (OSPR), which will be explained in detail in Section 2.3.5, and the projection output is shown in Fig. 2.10b. The simplest workaround for this issue is to use only half of the reconstruction field, like the target image in Fig. 2.10a. However, the side effect of this is that half of the energy will be wasted, leading to higher power consumption and heat dissipation.

2.3 Computer-Generated Holography (CGH)

Different from the traditional analogue optical holography and digital holography which both need a physically existing object to record the hologram, the CGH is a method to use computer algorithms to generate holograms without the need for the physical target object. Ideally, holograms can be simply computed using the inverse Fourier Transform, using the inverse functions of Eq. (2.29) and Eq. (2.31) derived in Section 2.2.2. The inverse Fourier Transforms on most targets will end up with results in complex numbers; however, as mentioned in Section 2.2.3, currently available SLMs cannot achieve fully complex modulation yet. Therefore, computer algorithms are needed to compute either phase-only or amplitude-only holograms, among which the former is preferred, as explained in Section 2.2.3.

This section reviews the existing methods in the literature for calculating phase-only holograms. The phase hologram is labelled as H, so that it differs from the previous notation of A for complex-valued hologram apertures. And the propagation function is unified as $\mathscr{P}$, which can be either the Fraunhofer propagation equation in Eq. (2.29) or the Fresnel propagation equation in Eq. (2.31) depending on the distance of the target field. Lastly, R denotes the reconstruction intensity of the hologram (i.e. $R = |\mathscr{P}[e^{jH}]|^2$).

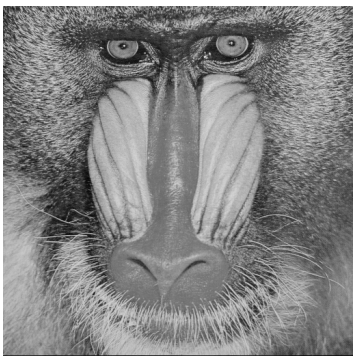


Fig. 2.11 Sample target image of a mandrill (T) [18]

A sample image of a mandrill (shown in Fig. 2.11) is chosen from the University of Southern California Signal and Image Processing Institute (USC-SIPI) Image Database [18] to test and compare the classical phase retrieval algorithms in the literature. As the square of the amplitude, which is the intensity of light, is the only visible component with the human eye [56], the diffracted electric field amplitude ($|E|$) will be targeted to match the square root of T.

To quantitatively analyse the reconstruction quality, two metrics are used in this sections, which are the normalised mean squared error (NMSE) [57] and the structural similarity index (SSIM) [58].

The NMSE is calculated using Eq. (2.34):

$$NMSE(x,y) = \frac{\frac{1}{n} * \sum(x-y)^2}{\sum(x)^2} \tag{2.34}$$

where x is the target, y is the measured output and n is the number of pixels in x and y. As the NMSE quantifies the total error between the measured output and the target, lower NMSE value indicates better quality. Different from the NMSE which calculates the absolute errors, the SSIM is a perception-based metric that considers the structural similarity as perceived, defined in Eq. (2.35):

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1) + (2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{2.35}$$

where μ_x and μ_y are the mean values of the pixels in x and y respectively, σ_x^2 and σ_y^2 are the variance, σ_{xy} is the covariance of x and y, $C_1 = (k_1 L)^2$ and $C_2 = (k_2 L)^2$, where L is the dynamic range of the pixel values and $k_1 = 0.01$ and $k_2 = 0.03$ by default [58]. Higher SSIM indicates better reconstruction quality.

2.3.1 Phase Unwrapping Method

The simplest method to get a phase-only hologram H is by directly extracting the phase of the reverse propagation from the target field, discarding the amplitude component (e.g. for Fraunhofer propagation, H will simply be the phase of the inverse Fourier transform $\mathscr{F}^{-1}$ of the target field). This method is named as the Phase Unwrapping method in this thesis as it directly unwraps the phases from the complex values. The pseudocode of the Phase Unwrapping method is shown in Algorithm 1 below:

Algorithm 1 Phase Unwrapping method

Input: Target image T, Propagation function $\mathscr{P}$ (e.g. Fresnel or Fraunhofer propagation)

Output: Phase hologram H and its reconstruction intensity R

$E \leftarrow \sqrt{T}$

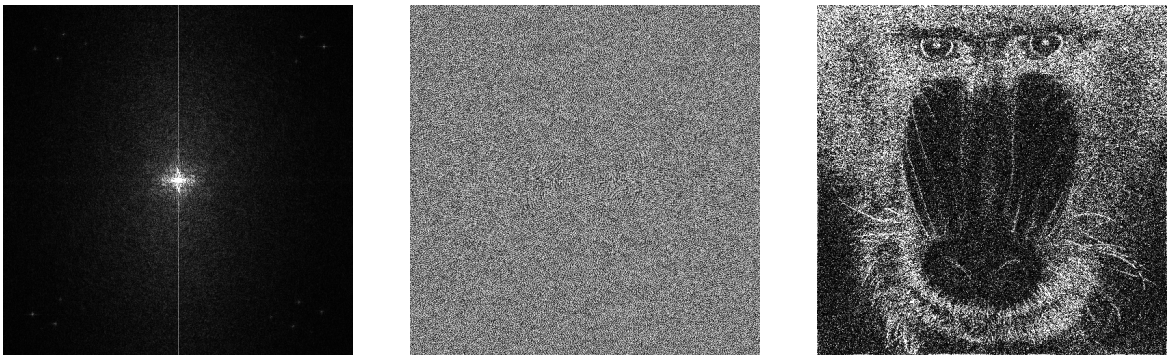
$A \leftarrow \mathscr{P}^{-1}[E]$

$H \leftarrow \angle A$

$R \leftarrow |\mathscr{P}[e^{jH}]|^2$

where $j = \sqrt{-1}$, the $\angle$ sign means phase unwrapping (i.e. taking arguments of the complex numbers element-wise in the matrix), and e^{jH} converts the angles back to complex numbers. All exponentials, modulus and square-root operators are carried out in an element-wise manner, so that the dimensions of T, A, H, R and E are all the same.

The Phase Unwrapping method (described in Algorithm 1) was then implemented in MATLAB and run on the sample target image shown in Fig. 2.11, with the distance set to infinity (i.e. in the Fraunhofer region using the propagation formula Eq. (2.31)). The results are shown in Fig. 2.12 below:



(a) Discarded amplitude ($|A|$) (b) Phase Hologram (H) (c) Reconstruction (R)

Fig. 2.12 Phase Unwrapping method output

After taking the inverse Fourier Transform, the amplitude and the phase of A are shown in Fig. 2.12a and Fig. 2.12b respectively. The next step then discards the amplitude component (Fig. 2.12a) and uses the phase component (in Fig. 2.12b) as the phase hologram H. Then the phase hologram went through a forward propagation and the resulting reconstruction intensity is shown in Fig. 2.12.

The reconstruction intensity is very far from the desired target image in Fig. 2.11. It shows that, discarding amplitude has introduced a significant loss of information. From a signal processing point of view, the peak around the centre in Fig. 2.12a corresponds to low spatial frequency signals, and discarding them causes the reconstruction in Fig. 2.12c to lose low frequency components and effectively becomes an edge detector. Another explanation of the poor reconstruction quality is that, this method is assuming a uniform phase profile for the target image, which is physically difficult to achieve. A simple improvement can be made by adding a random phase to the target, as shown in the pseudocode below:

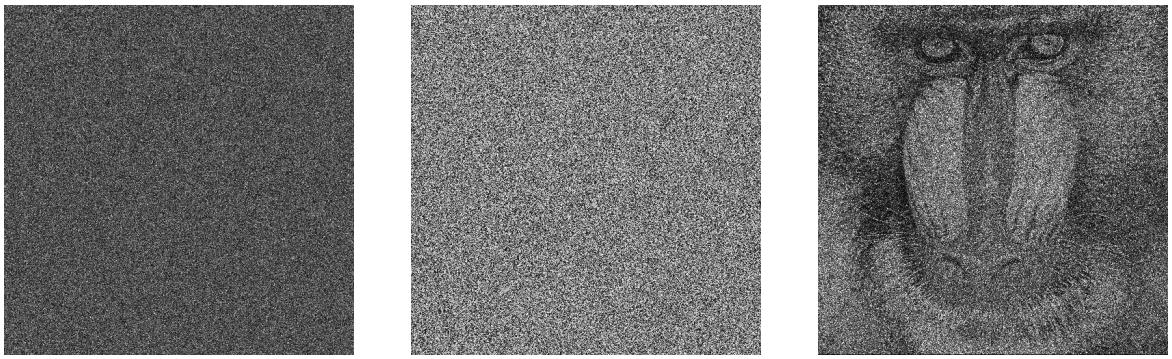
Algorithm 2 Improved Phase Unwrapping method with random phase added to the target field

Input: Target image T, Propagation function $\mathscr{P}$ (e.g. Fresnel or Fraunhofer propagation)

Output: Phase hologram H and its reconstruction intensity R

$E \leftarrow \sqrt{T} \times RandomPhase()$
$A \leftarrow \mathscr{P}^{-1}[E]$
$H \leftarrow \angle A$
$R \leftarrow |\mathscr{P}[e^{jH}]|^2$

The improved Phase Unwrapping method was implemented in MATLAB and produced the results in Fig. 2.13 below:



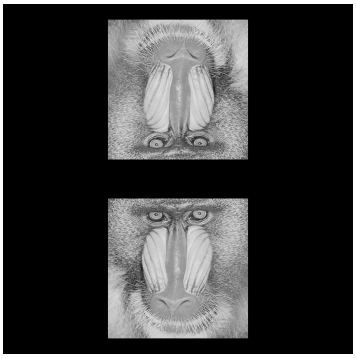
(a) Discarded amplitude ($|A|$) (b) Multi-level Phase Hologram (H) (c) Reconstruction (R)

Fig. 2.13 Output of the improved Phase Unwrapping method

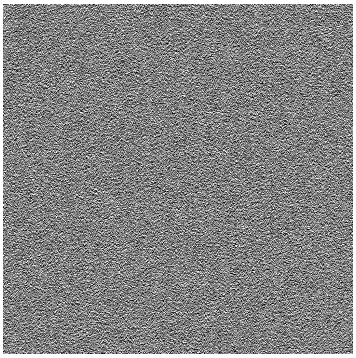
As shown in Fig. 2.13c, the reconstruction quality has been greatly improved, although still quite noisy. The amplitude of the hologram being discarded (shown in Fig. 2.13a) is a lot more uniformly distributed than the one in Fig. 2.12a, so that the loss of information

evenly spread across all spatial frequencies, leading to the much better reconstruction quality. However, the reconstruction quality is still quite noisy. The reconstruction in Fig. 2.13c has an NMSE of 1.0228×10^{-6} and an SSIM of 0.1603.

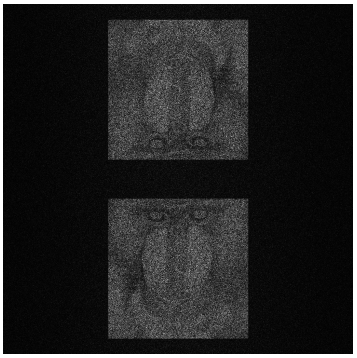
Moreover, additional error will be introduced during the quantisation step, which is necessary for the phase hologram to be displayed on SLMs with limited bit depth. As the SLM used in this thesis is a binary phase SLM, which has a rotational symmetry property as explained in Section 2.2.3, a target image is specifically designed as shown in Fig. 2.14a which is rotationally symmetrical.



(a) Target image for the binary-phase SLM



(b) Binary-phase Hologram



(c) Reconstruction

Fig. 2.14 Output of the improved Phase Unwrapping method with binary-phase quantisation

The binary phase hologram in Fig. 2.14b is generated by adding an additional binary quantisation step ($\mathscr{Q}$) on the phase hologram computed using Algorithm 2, which is simply rounding all phases to 0 and π radians, using Eq. (2.36)

$$\theta_{binary\ quantised} \begin{cases} 0 & -\frac{1}{2}\pi \leqslant \theta < \frac{1}{2}\pi \\ \pi & \theta < -\frac{1}{2}\pi \ or\ \theta \geqslant \frac{1}{2}\pi \end{cases} \tag{2.36}$$

where θ is the input phase value in range $[-\pi, \pi]$ and $\theta_{binary\ quantised}$ is the output quantised binary phase value. The reconstruction of the binary-phase hologram is shown in Fig. 2.14c, which has an NMSE of 4.5452×10^{-7} and an SSIM of 0.0603. To improve the reconstruction quality, better algorithms are needed. The following sections explores predecessors efforts in quality improvement.

2.3.2 Direct Binary Search (DBS) Algorithm

Direct Binary Search (DBS) algorithm [28] is an algorithm that generates the hologram by randomly flipping each pixel in the SLM between binary states (0 and π), one by one for many times in order to minimise the difference between its reconstruction intensity R and the target image T. The detailed algorithm is described in Algorithm 3 below:

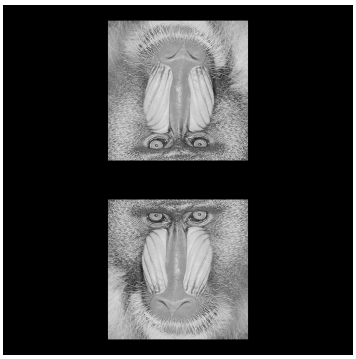
Algorithm 3 Direct Binary Search (DBS) algorithm

Input: Target image T, Propagation function $\mathscr{P}$, Loss function $\mathscr{L}$ (e.g. mean-squared error), Number of iterations N

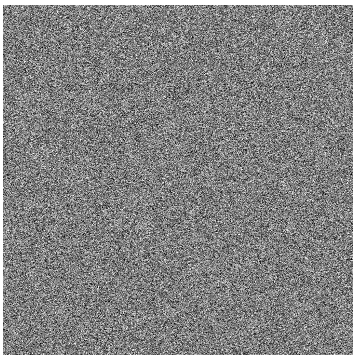
Output: Phase hologram H and its reconstruction intensity R

```
// Start with a random hologram with a size matching T
H ← Rand(Size(T))
R ← |𝒫[e^{jH}]|^2
L ← ℒ[R, T]
for n = 1 to N do
    // Flip a random pixel in the hologram
    H_n ← FlipRandomPixel(H)

    // Calculate the loss function for the new hologram
    R_n ← |𝒫[e^{jH_n}]|^2
    L_n ← ℒ[R_n, T]

    // Compare the new loss with the old one
    if L_n < L then
        // Accept the new hologram if loss is lower
        H ← H_n
        R ← R_n
        L ← L_n
    end if
end for
```

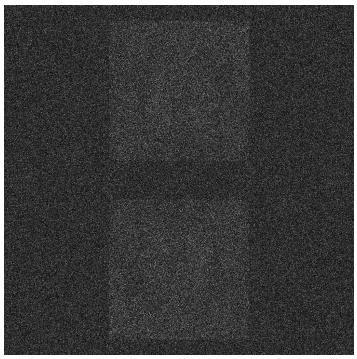
Although the DBS algorithm is specifically suited for generating binary phase holograms, it can also be adapted for generating multi-level phase holograms, by representing each level as binary numbers, at the cost of more computation.



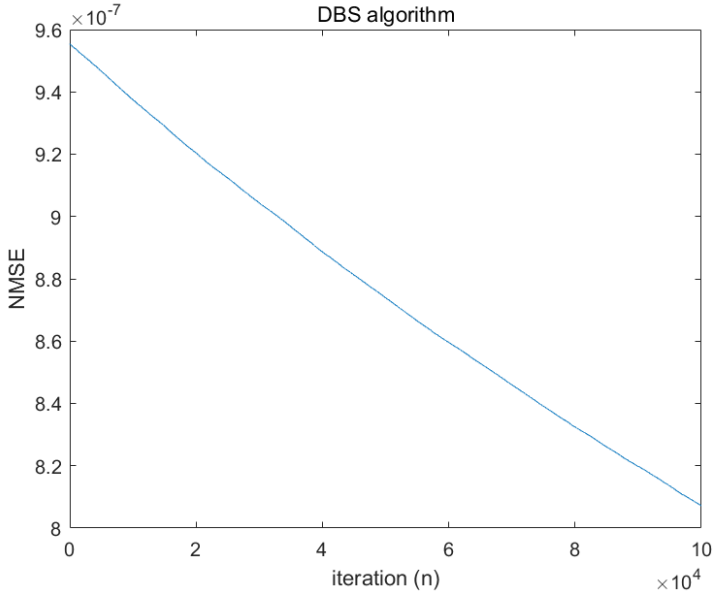
(a) Target image ($1024px \times 1024px$)



(b) Binary-phase Hologram



(c) Reconstruction



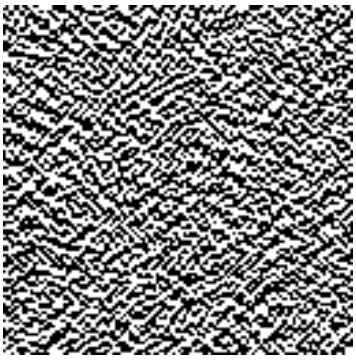
(d) NMSE v.s. iteration plot

Fig. 2.15 DBS algorithm running on the rotationally symmetrical mandrill target

DBS algorithm can sometimes find very accurate holograms if the run is lucky; however, it is extremely slow, because it takes numerous iterations (as shown in Fig. 2.15d, even 10^5 iterations has not reached a good convergence). The example run on the target image of resolution $1024px \times 1024px$ in Fig. 2.15a took more than one hour to finish the 10^5 iterations, which is still far from convergence. The binary-phase hologram produced is shown in Fig. 2.15b, and its corresponding reconstruction in Fig. 2.15c has an NMSE of 8.0717×10^{-7} which is 78% larger than the NMSE of the Phase Unwrapping method being 4.5452×10^{-7}. The reconstruction in Fig. 2.15c has an SSIM of only 0.0076, indicating

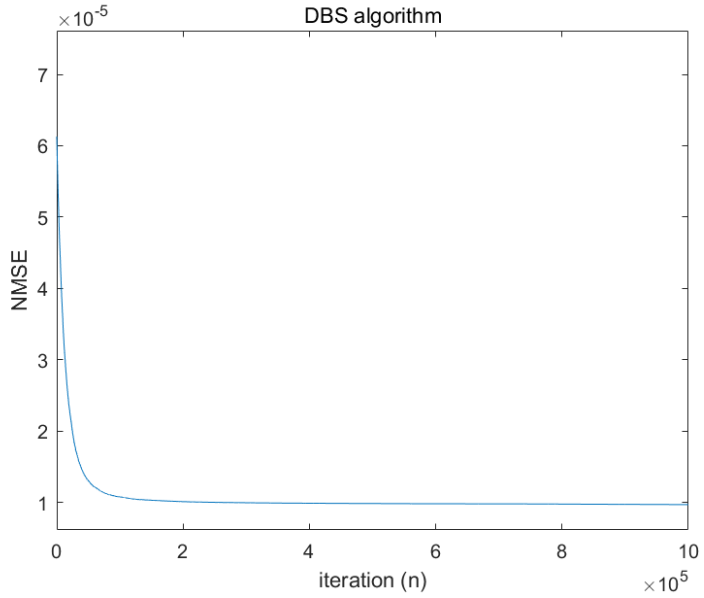
very poor structural similarity against the target image, and is much worse than the Phase Unwrapping method with SSIM being 0.0603 in Section 2.3.1.

(a) Target image ($128px \times 128px$)



(b) Binary-phase Hologram

(c) Reconstruction



(d) NMSE v.s. iteration plot

Fig. 2.16 DBS algorithm running on the low resolution target

As the DBS algorithm only flips one pixel per iteration, it naturally takes significantly longer to generate holograms with higher resolution. To test the programme on a smaller image for much quicker convergence, another target image has been designed as shown in Fig. 2.16a, which is also rotationally symmetrical as it is used for the binary-phase hologram generation. After 10^6 iterations, which took 10 minutes, the hologram generated is shown in Fig. 2.16b and its resulting reconstruction is shown in Fig. 2.16c, which has an NMSE of 9.6881×10^{-6} and an SSIM of 0.2871. The NMSE v.s. iteration plot in Fig. 2.16d shows that it reaches a

good convergence at around 2×10^5 iterations, corresponding to around 2 minutes. And the curve of NMSE is monotonically decreasing with iteration number, as only holograms with better results are accepted during the iterations.

In summary, the DBS algorithm is a slow but working algorithm for binary phase hologram generation. The programme running time scales up significantly when the target image's resolution gets higher. And also, as it only cares about local optimality at each iteration, it is a greedy algorithm that only follows the steepest descent route, which could easily get trapped in a local minimum where flipping any bit is not getting better reconstruction. Another consequence of the random nature is that the generated hologram will be different at each run, so the quality of the resulting reconstruction (R) will depend on how 'lucky' each run is. The Simulated Annealing (SA) algorithm [29] in the next session aims to resolve this issue.

2.3.3 Simulated Annealing (SA) Algorithm

The Simulated Annealing (SA) algorithm [29] is a variant of the DBS algorithm. It adopts a probabilistic approach to avoid the steepest gradient descent. Its name derives from the fact that it approximates the recrystallisation process during metal annealing and is particularly well-suited to avoiding the trap of local minima [59]. To implement this idea, we then need a function ($\mathscr{Z}$) to calculate the probability of the hologram (H), and a threshold p_t to decide whether the probability is high enough for the according hologram to be accepted. In this thesis, the probability is selected to be a random function and the threshold is chosen to be 0.9. The pseudocode for this algorithm is listed in Algorithm 4.

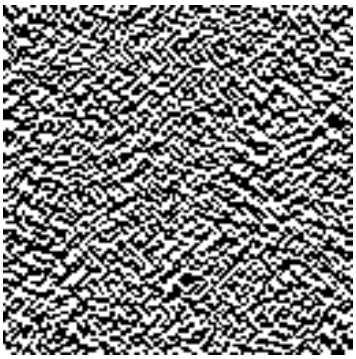
Algorithm 4 Simulated Annealing (SA) algorithm

Input: Target image T, Propagation function $\mathscr{P}$, Loss function $\mathscr{L}$, Number of iterations N, Probability function $\mathscr{Z}$, Probability threshold p_t

Output: Phase hologram H and its reconstruction intensity R

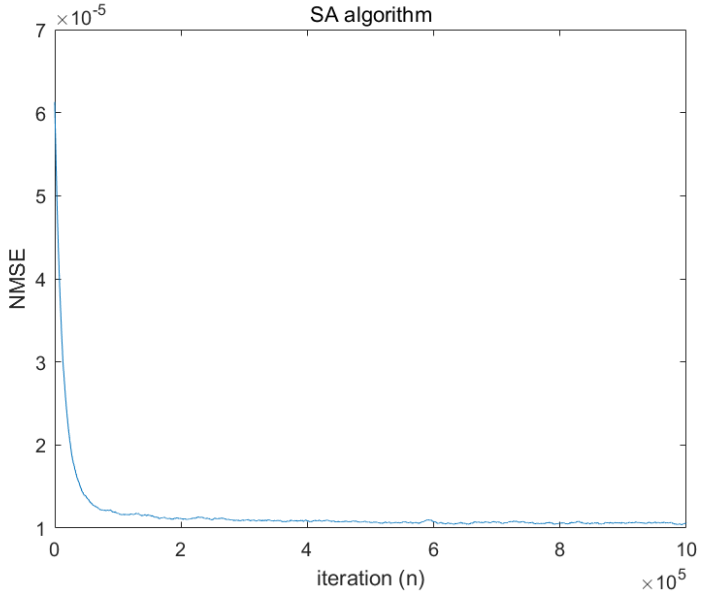
```
// Start with a random hologram with a size matching T
H ← Rand(Size(T))
R ← |𝒫[e^{jH}]|^2
L ← ℒ[R, T]
for n = 1 to N do
    // Flip a random pixel in the hologram
    H_n ← FlipRandomPixel(H)

    // Calculate the loss function for the new hologram
    R_n ← |𝒫[e^{jH_n}]|^2
    L_n ← ℒ[R_n, T]

    // Compare the new loss with the old one
    if L_n < L then
        // Accept the new hologram if loss is lower
        H ← H_n
        R ← R_n
        L ← L_n
    else
        // Calculate the probability of the hologram
        p_n ← 𝒵[H_n]
        if p_n > p_t then
            // Accept the new hologram if the probability exceeds the threshold
            H ← H_n
            R ← R_n
            L ← L_n
        end if
    end if
end for
```

(a) Target image ($128px \times 128px$)



(b) Binary-phase Hologram

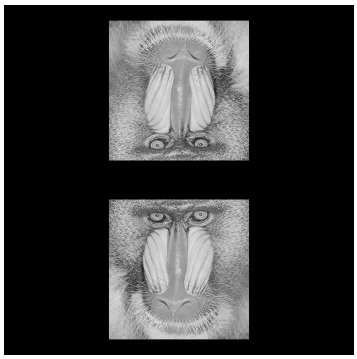
(c) Reconstruction



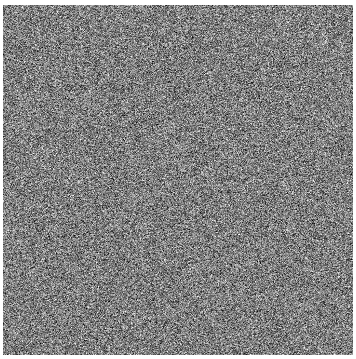
(d) NMSE v.s. iteration plot

Fig. 2.17 SA algorithm running on the low resolution target

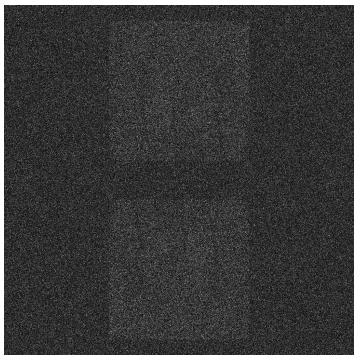
An implementation of SA algorithm with $p_t = 0.9$ was carried out on the low resolution target in Fig. 2.17a, and the resulting binary-phase hologram and its reconstruction are shown in shown in Fig. 2.17b and Fig. 2.17c respectively. From the NMSE v.s. iteration plot in Fig. 2.17d, it can be seen that, instead of the monotonic decrease observed in Fig. 2.16d for DBS algorithm, the SA algorithm has occasional rises in NMSE, which happens when the probability p_n exceeds the threshold p_t. The final NMSE was recorded to be 1.0542×10^{-5} and the final SSIM was 0.2750, which are both slightly worse than the DBS algorithm in this case. Due to the probabilistic nature of the SA algorithm, although it can avoid being trapped in local optimal points, the 'jump backs' can also cause delays in convergence.



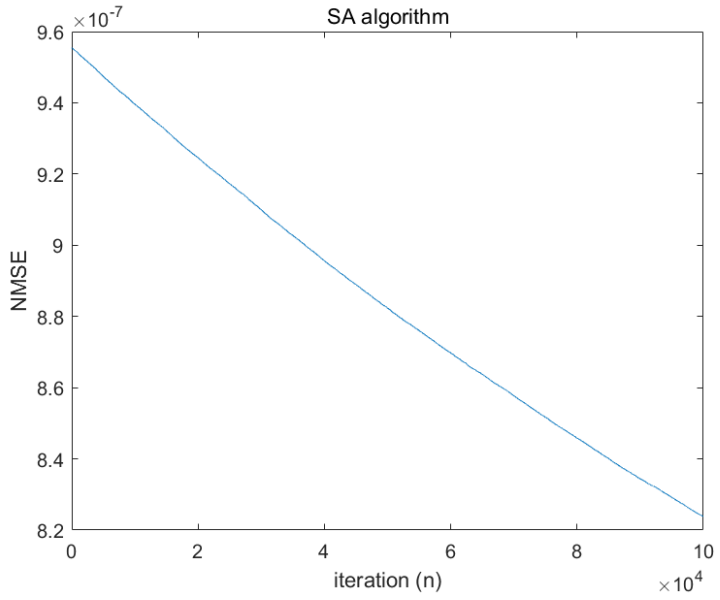
(a) Target image ($1024px \times 1024px$)



(b) Binary-phase Hologram



(c) Reconstruction



(d) NMSE v.s. iteration plot

Fig. 2.18 SA algorithm running on the rotationally symmetrical mandrill target

Then the SA algorithm was run for the mandrill target in Fig. 2.18a. The convergence plot in Fig. 2.18d shows that it did not converge within 10^5 iterations, which took around one hour. The binary phase hologram generated is shown in Fig. 2.18b and its corresponding reconstruction intensity is shown in Fig. 2.18c, which has an NMSE of 8.2054×10^{-7} and an SSIM of 0.0073, which are slightly worse than the DBS algorithm results in Fig. 2.15c with an NMSE of 8.0717×10^{-7} and an SSIM of 0.0076.

Both the DBS and the SA algorithms rely on flipping only a single pixel per iteration, which is very inefficient. A better algorithm should change the values of most pixels at every iteration for better efficiency, and to converge within much fewer iterations for lower computational

time. The Gerchberg-Saxton (GS) algorithm [30] is a classical example, which will be further explained in Section 2.3.4.

2.3.4 Gerchberg-Saxton (GS) Algorithm

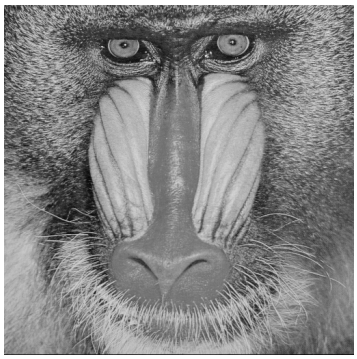
The Gerchberg-Saxton (GS) algorithm [30] is a revolutionary algorithm and is much better and more robust than the algorithms introduced in the previous sections. Although being more than 50 years old, the GS algorithm is still frequently used and has lots of variants [60–62]. It functions by iteratively determining the phase profile of the hologram required to reconstruct a target image, looping between the hologram and the reconstruction plane, and applying constraints to each plane accordingly during each iteration. GS algorithm is very easy to implement, its pseudocode is shown in Algorithm 5.

Algorithm 5 Gerchberg-Saxton (GS) Algorithm

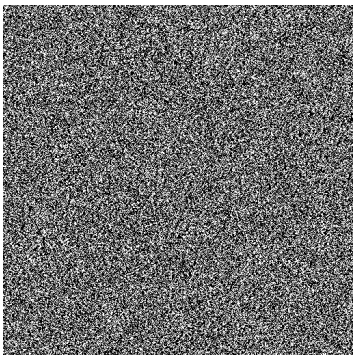
Input: Target image T, Propagation function $\mathscr{P}$, Number of iterations N, Initial phase Φ (e.g. random, zeros, or other patterns)

Output: Phase hologram H and its reconstruction intensity R

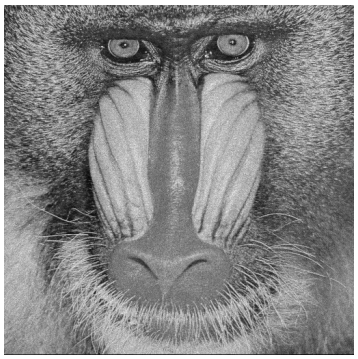
```
// Initiate E with amplitude √T and initial phase Φ
E ← √T × e^{jΦ}
for n = 1 to N do
    // Compute the hologram plane
    A ← 𝒫^{-1}[E]
    // Apply the phase-only constraint at the hologram plane
    A ← e^{j∠A}

    // Compute the propagation for the new hologram
    E ← 𝒫[A]
    // Apply the target field amplitude constraint at the reconstruction plane
    E ← √T × e^{j∠E}
end for
H ← ∠A
R ← |𝒫[A]|^2
```



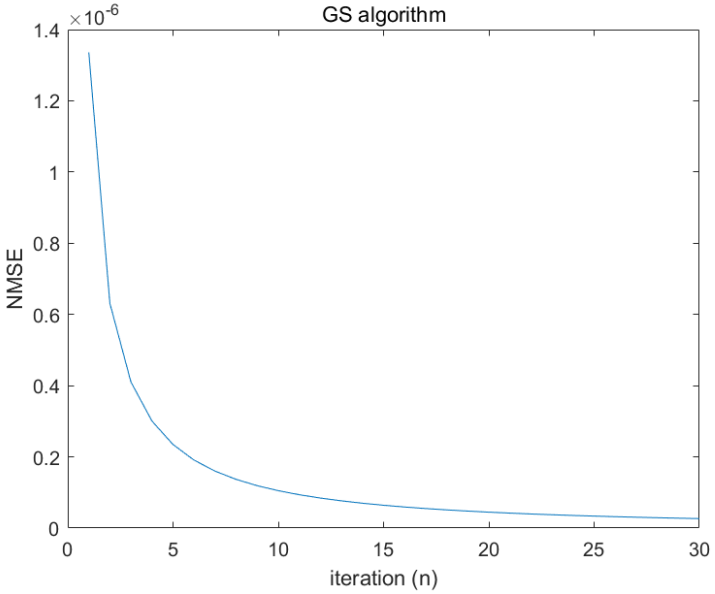
(a) Target image ($512px \times 512px$)



(b) Multi-level Phase Hologram



(c) Reconstruction

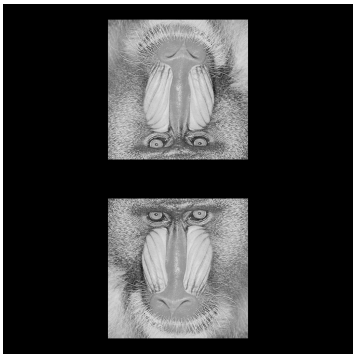


(d) NMSE v.s. iteration plot

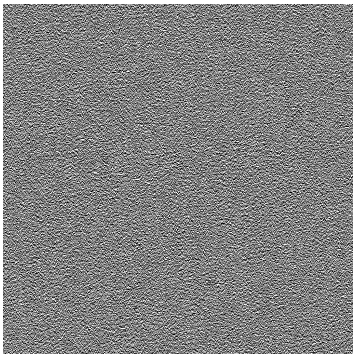
Fig. 2.19 GS algorithm output on the mandrill target

The GS algorithm described in Algorithm 5 was implemented in MATLAB and was first run on the mandrill target image in Fig. 2.11, and the output results are shown in Fig. 2.19. Fig. 2.19b is the multi-level hologram generated after 30 iterations, and its reconstruction is shown in Fig. 2.19c. The reconstruction in Fig. 2.19c has an NMSE of 2.6612×10^{-8} and an SSIM of 0.7940, which are both much better than the single-iteration Phase Unwrapping method's result in Fig. 2.13c (with an NMSE of 1.0228×10^{-6} and an SSIM of 0.1603). The NMSE v.s. iteration plot in Fig. 2.19d shows that the GS algorithm converged quickly, providing very good result in tens of iterations, which is much fewer than the DBS and SA algorithms. Although the GS algorithm is more computationally expensive at each iteration,

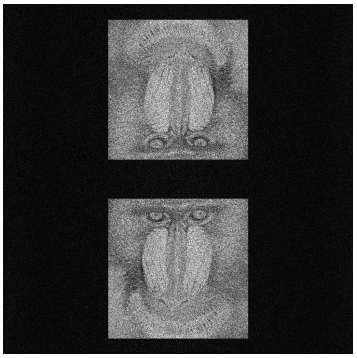
as it needs to compute both a forward and an backward propagation, leading to two Fourier transforms every iteration, the GS algorithm is still much faster and provides much better reconstruction quality than the DBS and SA algorithms.



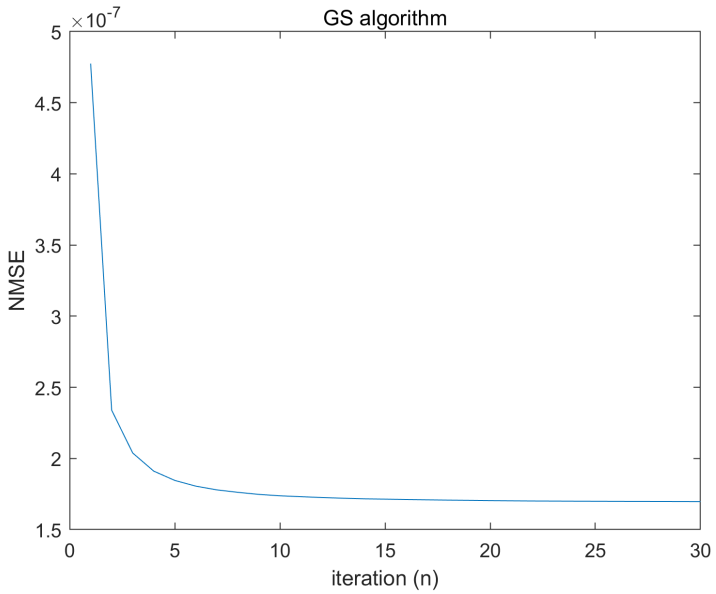
(a) Target image ($1024px \times 1024px$)



(b) Binary-phase Hologram



(c) Reconstruction



(d) NMSE v.s. iteration plot

Fig. 2.20 GS algorithm running on the rotationally symmetrical mandrill target

Then the GS algorithm was adapted to generate binary-phase holograms, for use on the binary-phase SLM in this thesis. The change was simply implemented by adding a quantisation function ($\mathscr{Q}$) when applying the phase-only constraint at the hologram plane (i.e. the line '$A \leftarrow e^{j\angle A}$' in Algorithm 5 is changed to '$A \leftarrow e^{j\mathscr{Q}[\angle A]}$'). The results are shown in Fig. 2.20. Fig. 2.20d shows good convergence within 20 iterations. The resulting binary-phase hologram is shown in Fig. 2.20b and its corresponding reconstruction in Fig. 2.20c has an NMSE

of 1.6968e-07 and an SSIM of 0.0619, which are both better than the Phase Unwrapping method's result in Fig. 2.14 (with an NMSE of 4.5452×10^{-7} and an SSIM of 0.0603). In summary, the GS algorithm is quick and robust. On my laptop computer of model ASUS ROG Zephyrus M16, which has a CPU of model i7-11800H and a GPU of model RTX3060, the 30 iterations took 1.5 seconds to complete. It reached convergence in tens of iterations. However, as it is still iterative, generating holograms in real-time is still a challenge, and the reconstruction still suffers from noise.

2.3.5 One-Step Phase Retrieval (OSPR) Algorithm

The OSPR algorithm was first demonstrated by Cable and Buckley [46]. It is a solution to high-quality hologram reconstruction that relies on time multiplexing of holograms, exploiting the response time of eye in order to reduce noise in the replay field [17]. The random noise is averaged by the eye, while the target image stays, so that the average noise can be reduced. The perceived noise is lessened by the temporal average detected by the eye, rather than computational optimisation of the hologram [17]. The pseudocode for OSPR is shown in Algorithm 6 below.

Algorithm 6 One-Step Phase Retrieval (OSPR) algorithm

Input: Target image T, Propagation function $\mathscr{P}$, Number of sub-frames S, Quantisation function $\mathscr{Q}$

Output: List of phase holograms $H[1 \dots S]$

```
// Compute a list of hologram sub-frames
for i = 1 to S do
    E ← √T · RandomPhase()
    A ← P^{-1}[E]
    H[i] ← Q[∠A]
end for

// Then display the sub-frames on the phase modulator sequentially
i ← 1
while True do
    Display(H[i])
    i ← i + 1
    if i > S then
        i ← 1
    end if
end while
```

When generating the list of holograms, it repetitively computes the inverse propagation of the target amplitude (which is the square-root of the target intensity T) multiplied by different random phases, for a total of S times to generate S hologram sub-frames ($H[1 \dots S]$). The computation of each hologram sub-frame is the same as the Phase Unwrapping method in Algorithm 2 discussed in Section 2.3.1. Then the S hologram sub-frames are displayed sequentially on a SLM having a refresh rate being so fast that the average reconstruction intensity is perceived by the human eyes. As currently available fast SLMs are binary-phase modulators, an example run on the rotationally symmetrical target previously used in Fig. 2.15a was carried out for 24 sub-frames ($S = 24$). The results are summarised in Fig. 2.21.

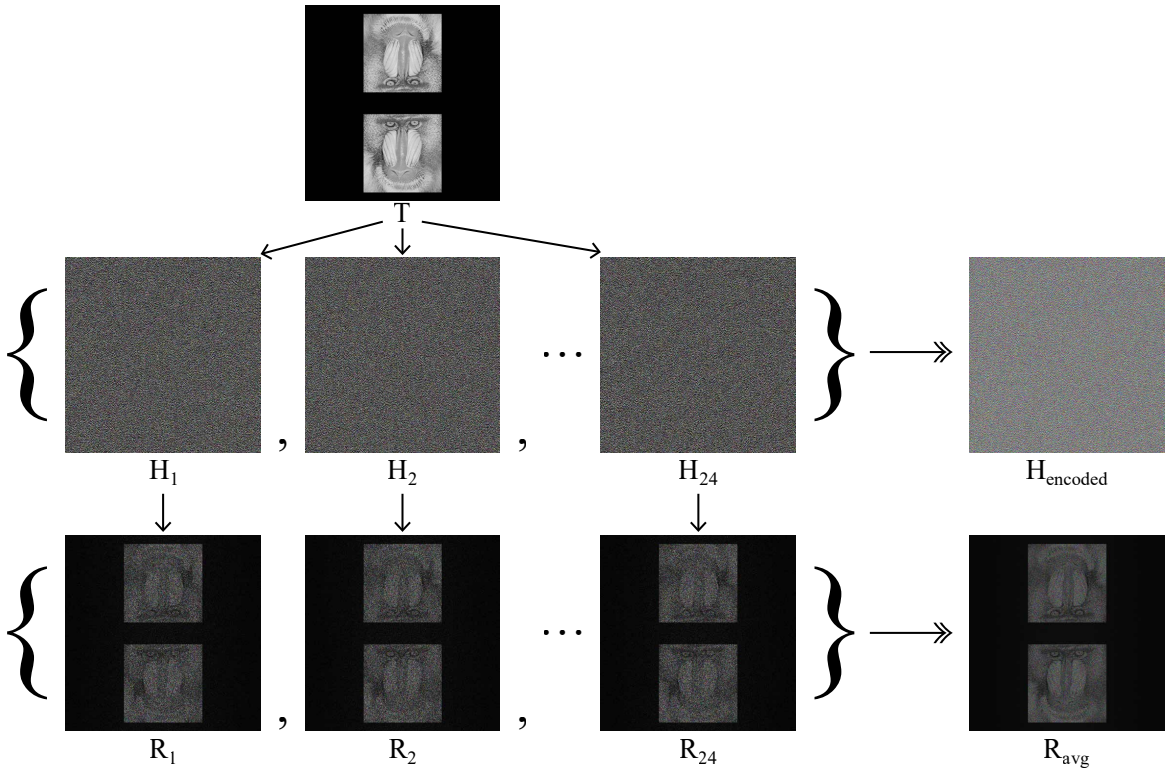


Fig. 2.21 OSPR algorithm running on the rotationally symmetrical mandrill target

In Fig. 2.21, a total of 24 binary-phase hologram sub-frames ($H_1, H_2, \ldots, H_{24}$) were generated for the target image T. For easier data transfer and to fit with the common display signal formats, the 24 binary-phase hologram sub-frames are encoded into a single file with 8 bit depth and RGB (red-green-blue) channels, so that each of the $(8 \times 3 =)24$ bit-planes corresponds to a single binary-phase hologram sub-frame. For a quantitative analysis, the reconstruction intensities of the hologram sub-frames are computed in $R_1, R_2, \ldots, R_{24}$ respectively, whose average is R_{avg} in Fig. 2.21. The average reconstruction intensity R_{avg} has an NMSE of 9.8632×10^{-8} and an SSIM of 0.1321, which are both significantly better than the GS algorithm (NMSE=1.6968×10^{-7}, SSIM=0.0619) and the Phase Unwrapping method (NMSE=4.5452×10^{-7}, SSIM=0.0603).

The major advantage of the OSPR algorithm is that it is fast. It is non-iterative and requires only one Fourier Transform per frame, taking less than a second to generate a 24-frame hologram set. Its non-iterative nature also allows it to be parallelised to further improve computation speed, which is crucial for Light Blue Optics who made a real-time holographic laser projector commercially available in 2010 [63], although the product was later discontinued for financial reasons. The downside of this algorithm is that the sub-frames are

independent from each other, and the final reconstruction output is still subject to some noise, as they are only more randomly distributed instead of being reduced. There is a variant of improvement on the OSPR algorithm, called Adaptive OSPR (AD-OSPR), to be introduced in Section 2.3.6.

2.3.6 Adaptive One-Step Phase Retrieval (AD-OSPR) Algorithm

The AD-OSPR algorithm [64] is a variant of the OSPR algorithm. It aims to improve the reconstruction quality without introducing a significant amount of additional computational cost. It functions in such a way that, when computing the second sub-frame onwards, it subtracts the average reconstruction from the target image to get the error, so that it can compensated in the next iteration. To help explain the process in detail, a pseudocode is written for the AD-OSPR algorithm, as shown in Algorithm 7.

Algorithm 7 Adaptive One-Step Phase Retrieval (AD-OSPR) algorithm

Input: Target image T, Propagation function $\mathscr{P}$, Number of sub-frames S, Quantisation function $\mathscr{Q}$

Output: List of phase holograms $H[1 \dots S]$

// Compute a list of hologram sub-frames
$T[1] \leftarrow T$
$R_{total} \leftarrow 0$
for i = 1 to S **do**
  $E \leftarrow \sqrt{T[i]} \cdot$ RandomPhase()
  $A \leftarrow \mathscr{P}^{-1}[E]$
  $H[i] \leftarrow \mathscr{Q}[\angle A]$
  // Compute the reconstruction intensity
  $R \leftarrow |\mathscr{P}[e^{jH[i]}]|^2$
  // Carry out energy conservation to match the total energy of the target image
  $R \leftarrow R \cdot \sqrt{\frac{sum(T^2)}{sum(R^2)}}$ // squares and sums are taken element wise
  // Compute the total reconstruction intensity so far
  $R_{total} \leftarrow R_{total} + R$
  // Compute the new target for the next iteration
  for $[x, y]$ = $[1, 1]$ to size(T) **do** // loop among all the pixels
    if $(i+1) \cdot T[x,y] > R_{total}[x,y]$ **then**
      $T[i+1][x,y] \leftarrow (i+1) \cdot T[x,y] - R_{total}[x,y]$
    else
      $T[i+1][x,y] \leftarrow 0$
    end if
  end for
end for
// Then display the sub-frames on the phase modulator sequentially
$i \leftarrow 1$
while True **do**
  Display($H[i]$)
  $i \leftarrow i+1$
  if $i > S$ **then**
    $i \leftarrow 1$
  end if
end while

In the pseudocode in Algorithm 7, the target intensity for the first iteration ($T[1]$) is initialised as the input target image T, and the total reconstruction intensity for the holograms generated up to each iteration (R_{total}) is initialised as 0. Then the **for** loop starts from the same routine as the OSPR algorithm, multiplying a random phase to the square root of target intensity, taking an inverse propagation, and applying the binary phase quantisation. Then, the total reconstruction is computed by forward propagating the quantised hologram, conserving the energy to match the target image, and adding to the total reconstruction of the last iteration. The total reconstruction is then used to compute the target intensity for the next iteration, by subtracting the total reconstruction from $(i+1)$ times the target image, with an **if-else** statement to avoid negative intensity values. To provide a visual illustration, an example run of the AD-OSPR algorithm was carried out on the mandrill target as shown in Fig. 2.22.

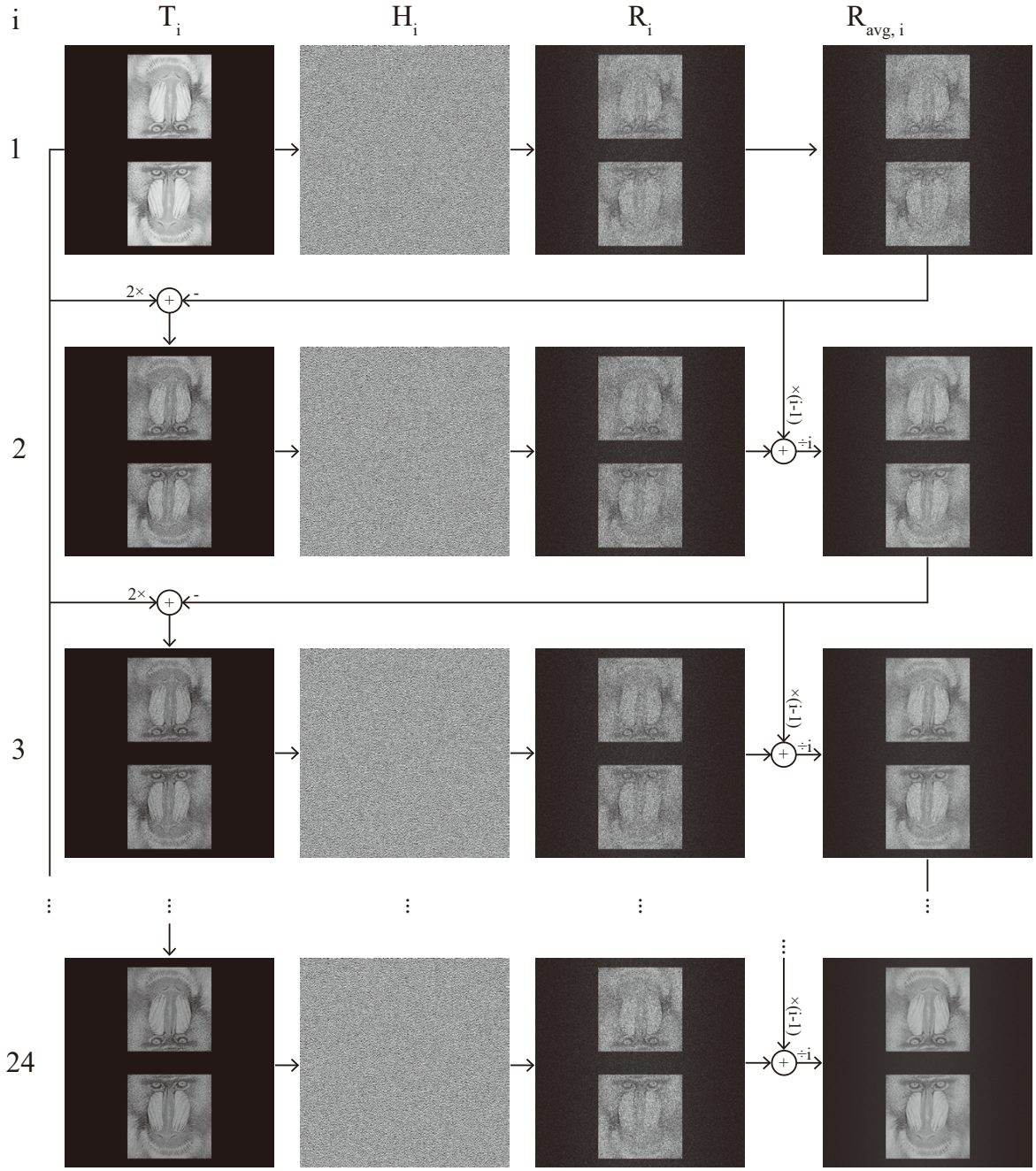


Fig. 2.22 AD-OSPR algorithm running on the rotationally symmetrical mandrill target

In Fig. 2.22, each row of images correspond to one iteration that is indexed in the 'i' column, where only iterations 1, 2, 3, 24 are shown here, while iterations 4-23 are omitted to save space. At each iteration, the binary-phase hologram H_i is computed from its target T_i, and the corresponding reconstruction is shown in R_i. Instead of showing the $R_{total,i}$, the $R_{avg,i} = \frac{R_{total,i}}{i}$ is shown here as $R_{total,i}$ has increasing dynamic range at each iteration.

For the first iteration, the target image of the same one as in Fig. 2.21 is used, and the first average reconstruction $R_{avg,1}$ is just the first reconstruction R_1. Then, from the second iteration onwards, the target intensity T_i is updated following the pseudocode in Algorithm 7. After 24 iterations, the 24 hologram sub-frames are computed and their average reconstruction ($R_{avg,24}$) had an NMSE of 9.1161×10^{-8} and a SSIM of 0.1992, which are both better than the OSPR algorithm in Section 2.3.5 who achieved an NMSE of 9.8632×10^{-8} and an SSIM of 0.1321.

The programme running time of the AD-OSPR algorithm is nearly the same as the OSPR algorithm, both being around 0.7 second. Therefore the AD-OSPR algorithm is quite an effective improvement on the original OSPR algorithm, achieving a 7.6% reduction in NMSE and 50.8% improvement in SSIM without adding much computation. The disadvantage is however, that it cannot be parallelised, unlike the OSPR algorithm, as it needs to have the total reconstruction intensity result from the previous iteration for use in the next iteration.

2.3.7 3D CGH

Section 2.3.1 - Section 2.3.6 described several algorithms to generate a phase hologram for a single target image. However, the major benefit of holography is that it can produce a true 3D light field, much more than just a single slice 2D image. Then the problem arises as how to generate a hologram for 3D targets to make full use of the major benefit of holography. The simplest method is to slice the 3D target into a set of 2D layers, like computed tomography (CT) scanning, and then generate a hologram so that its Fresnel propagation (in Eq. (2.29)) at each depth (z) matches the according layer. This subsection therefore reviews how the current methods in Section 2.3.1 - Section 2.3.6 can be adapted to multi-depth hologram generation.

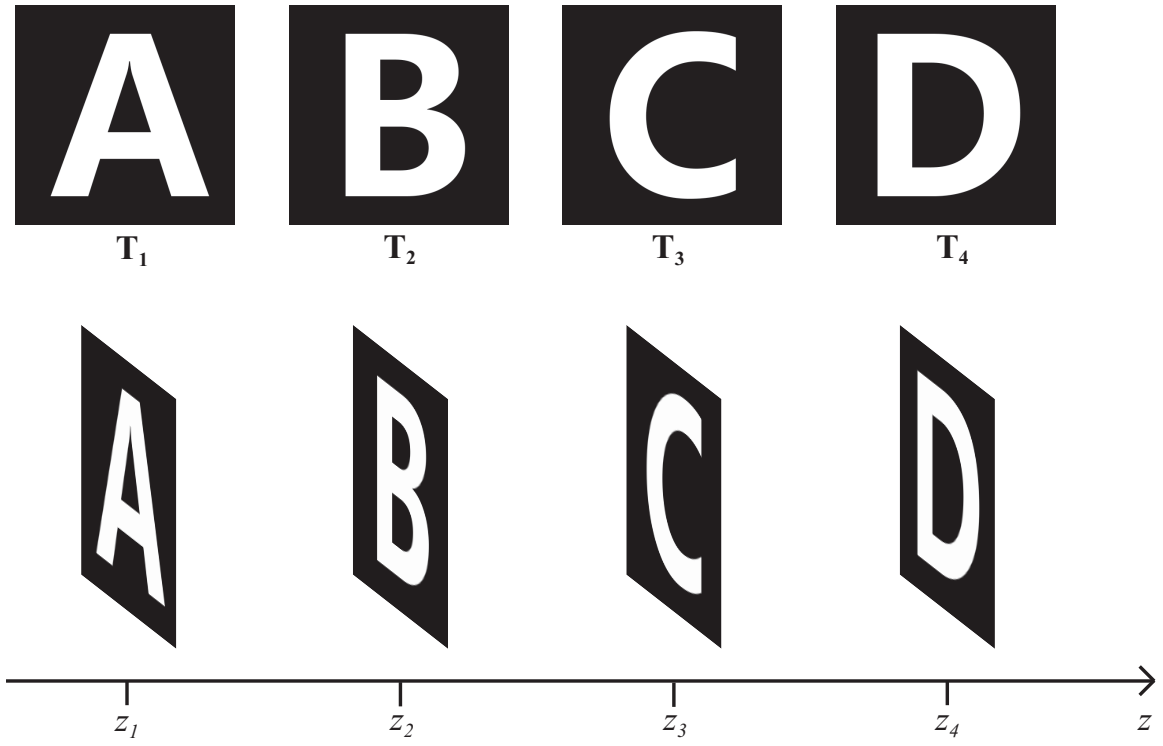


Fig. 2.23 Multi-slice target consisted of 4 different characters at different distances

For fair tests on algorithms, an example 3D target field consisted of 4 slices has been created using alphabets, as shown in Fig. 2.23. T_1 to T_4 have the same resolution of $512px \times 512px$, and the distances z_1 to z_4 are set to $1, 2, 3, 4cm$ respectively.

Phase Unwrapping with Superposition

To adapt the Phase Unwrapping method in Section 2.3.1 to compute multi-depth hologram, firstly, a set of holograms are generated corresponding to each layer of the target field respectively, using the inverse of the Fresnel propagation function in Eq. (2.29). Then, based on the principle of superposition, the set of holograms are added up to form the final hologram, whose phase is directly extracted to be the phase-only hologram while the amplitude component is discarded.

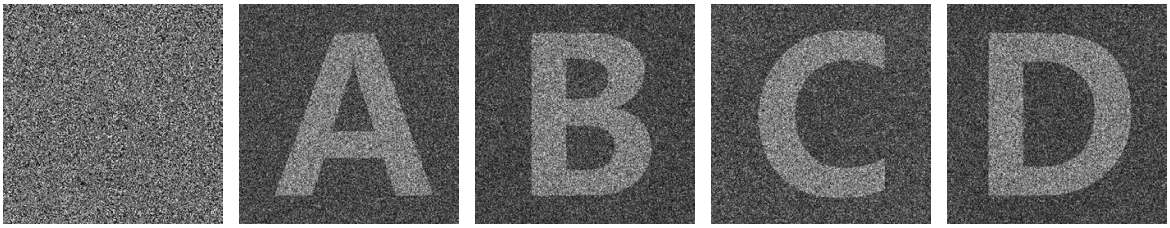


Fig. 2.24 Phase Unwrapping with Superposition method's result on the 4-slice target

The simulation results shown in Fig. 2.24 demonstrates the effectiveness of the Phase Unwrapping method. The reconstructions at each depth are legible, despite the presence of some noise. However, after applying a binary-phase quantisation on the hologram, the reconstruction quality deteriorates significantly, as shown in Fig. 2.25.

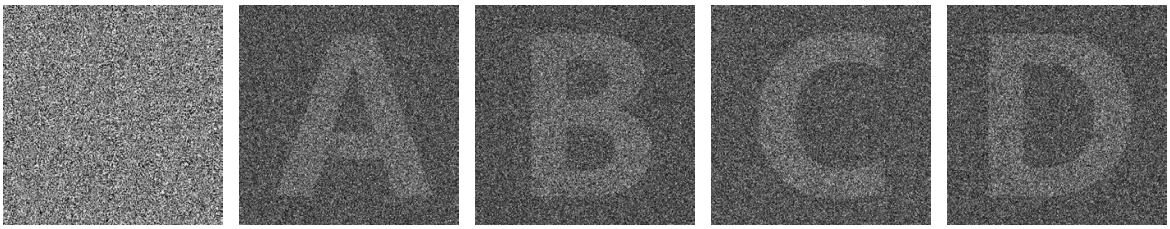


Fig. 2.25 Phase Unwrapping method's result on the 4-slice target after binary quantisation

To improve the reconstruction quality, the time-multiplexed OSPR algorithm (in Section 2.3.5) is adapted to produce multi-frame holograms for 3D targets.

OSPR algorithm adaptation

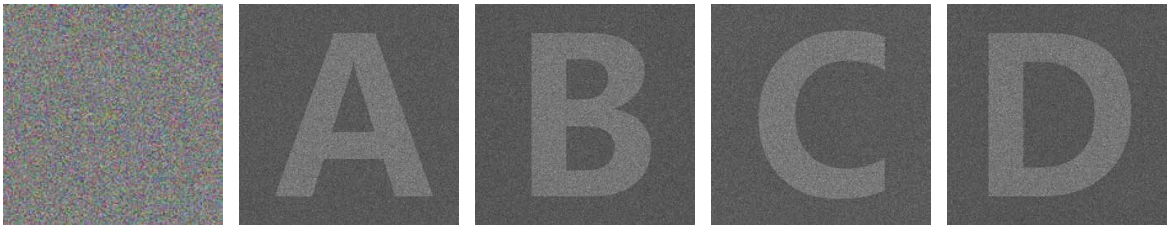


Fig. 2.26 OSPR algorithm's result on the 4-slice target

The OSPR algorithm in Section 2.3.5 was adapted to generate multi-depth targets by doing the Phase Unwrapping method in the previous paragraph for a total of S times to form S sub-frames. The results of the example run with $S = 24$ is shown in Fig. 2.26. The average reconstruction has much less noise than the single frame one in Fig. 2.25. However, the contrast is not high enough. Therefore, one of the major objective of this thesis is to search for time-multiplexed binary-phase holograms with better reconstruction quality than the existing methods, which will be further explored in Chapter 5.

GS algorithm adaptations

The GS algorithm in Section 2.3.4 can also be adapted to generate phase-only holograms for multi-depth 3D targets. Three different adaptations are reviewed in the following paragraphs.

GS algorithm adaptation 1 - Superposition: Similar to the OSPR adaptation, the first method of adapting the GS algorithm for 3D CGH is by superposition. Firstly, a set of holograms are generated individually using the GS algorithm on each slice of the 3D target, and then those holograms are superposed into a total hologram whose phase is then unwrapped to be the final phase hologram.

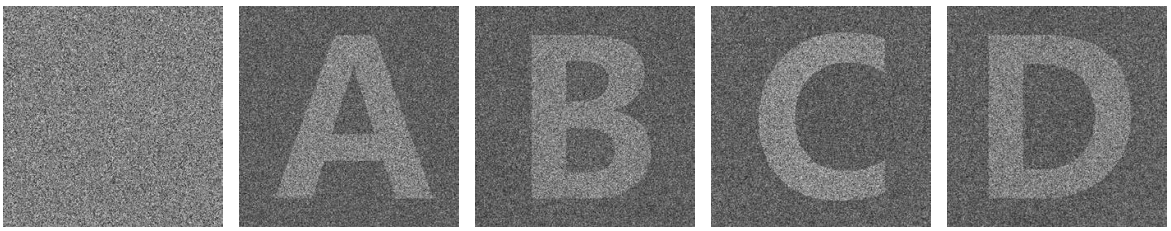


Fig. 2.27 GS with superposition method's result on the 4-slice target

The GS with superposition method was run on the target field in Fig. 2.23 and produced the result in Fig. 2.27. Similar to the observations with the OSPR adaptation, the superposition leads to defocusing between each slice, giving rise to the background noise. The advantage of this method is its simplicity of implementation, while the disadvantages of this method are its poor reconstruction quality and its high number of iterations, which grows linearly with the number of slices.

GS algorithm adaptation 2 - GS with Sequential Slicing (GS-SS): The second adaptation is called GS-SS, Instead of propagating to a fixed target image at each iteration, the GS algorithm is modified to propagate to a different distance at each iteration, and the target amplitude constraint of the according distance is applied (i.e. line $E \leftarrow \sqrt{T} \times e^{j\angle E}$ in Algorithm 5 becomes $E \leftarrow \sqrt{T_{i\%n}} \times e^{j\angle E}$, where i is the iteration number, n is the total number of slices and the % sign takes the remainder of the division). Such method is named sequential slicing, where the algorithm sweeps through the slices one by one sequentially during its iterations.

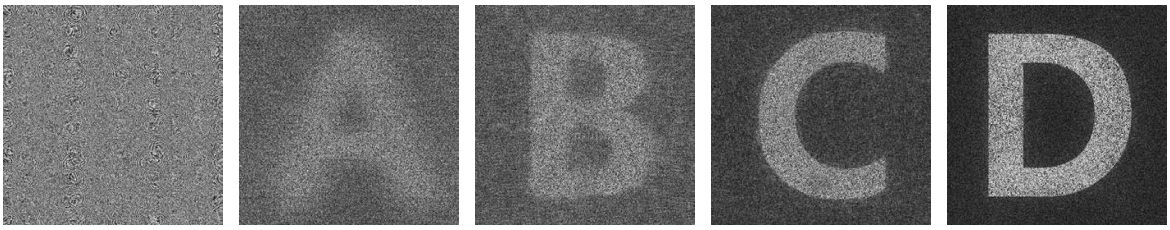


Fig. 2.28 GS with sequential slicing algorithm's result on the 4-slice target

The GS with sequential slicing (SS) algorithm was run on the target field in Fig. 2.23 for 100 iterations, and the result is shown in Fig. 2.28. The interesting phenomena is that, as the 100 iterations terminated at the 4^{th} layer, the 4^{th} reconstruction (R_4) has the best quality among all slices, and the reconstruction quality deteriorates as the slice index reduces.

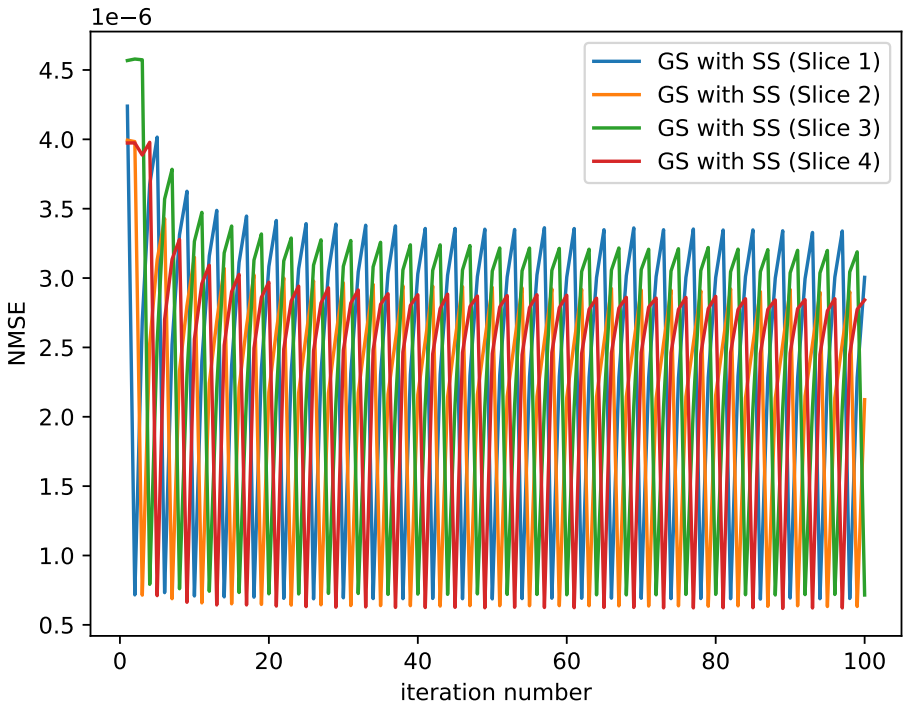


Fig. 2.29 GS with SS algorithm's NMSE v.s. iteration number plot

As a further investigation, the NMSE of each slice is plotted against iteration number in Fig. 2.29. It shows that the iterations did not converge. The NMSE curves are oscillating severely. When the NMSE of one slice decreases, the NMSE of all other slices increases. Applying the amplitude constraint at a single depth worsen all the other slices. There is a solution to this issue in the literature, called Dynamic Compensatory GS (DCGS) [62], as introduced in the following paragraph.

GS algorithm adaptation 3 - Dynamic Compensatory GS (DCGS): The DCGS algorithm 'softens' the target field amplitude constraint [62]. Instead of forcing the amplitude of the reconstructed field to the target amplitude directly, it allows a fraction α of the original amplitude to be retained (i.e. $E \leftarrow \sqrt{T_{i\%n}} \times e^{j\angle E}$ becomes $E \leftarrow [\alpha \times |E| + (1-\alpha) \times \sqrt{T_{i\%n}}] \times e^{j\angle E}$), where α is adjusted dynamically at each iteration. The DCGS algorithm was implemented and run on the 4-slice target in Fig. 2.23. The result is shown in Fig. 2.30.

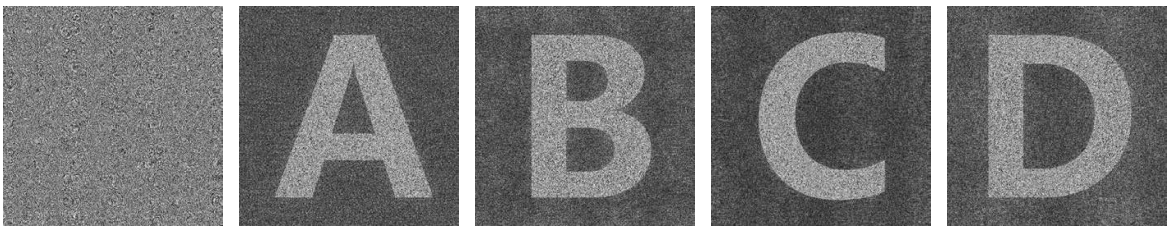


Fig. 2.30 DCGS algorithm's result on the 4-slice target

Fig. 2.30 shows the effectiveness of the DCGS algorithm, where all slices have good quality instead of the huge inter-slice quality imbalance observed in Fig. 2.28. The NMSE of each slice is plotted against the iteration number in Fig. 2.31.

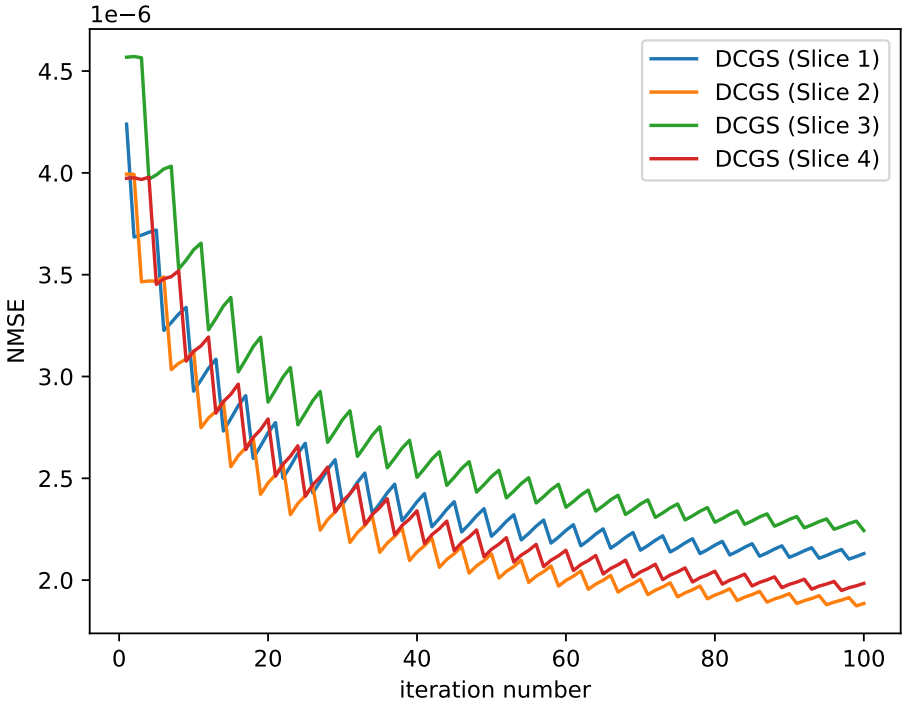


Fig. 2.31 DCGS algorithm's NMSE v.s. iteration number plot

Fig. 2.31 shows a much better convergence than Fig. 2.29. Although the bounce-backs still present (i.e. when the NMSE of one slice decreases, those of other slices increase), the overall trend of NMSE is decreasing as the iterations continue.

Chapter 3

Digital Pre-Distorted One-Step Phase Retrieval (DPD-OSPR) Algorithm

Note: The work in this chapter has been published in Ref. [5]

In a computer-generated holographic projection system, the image is reconstructed via the diffraction of light from a spatial light modulator. In this process, several factors could contribute to non-linearities between the reconstruction and the target image. This chapter evaluates the non-linearity of the overall holographic projection system experimentally, using binary phase holograms computed by the OSPR algorithm introduced in Section 2.3.5, and then applies a digital pre-distortion (DPD) method to correct for the non-linearity. Both a notable increase in reconstruction quality and a significant reduction in mean squared error were observed, proving the effectiveness of the proposed DPD-OSPR algorithm.

3.1 Introduction

Chapter 2 reviewed the real-time CGH method called OSPR (in Section 2.3.5), which was fast enough for real-time holography, but the reconstruction quality has still got potential for improvement. Hence, a computationally inexpensive method is needed to improve the reconstruction quality whilst maintaining the real-time property of the OSPR algorithm.

There are several factors contributing to the non-linearities between the reconstruction of hologram and the target image, including the calculation and quantisation of the hologram, the modulation of the light and the imperfections in the optical setup. This chapter proposes the digital pre-distorted one-step phase retrieval (DPD-OSPR) algorithm. The digital pre-distortion (DPD) is carried out on the holographic projection system by measuring the non-linearity experimentally and applying the according pre-distortion curve on target images. DPD can be done via a one-to-one correction curve or a look-up table (LUT) which allows the relationship between the input and output to be adjusted without any heavy computation.

The intuition of the proposed DPD-OSPR algorithm for CGH comes from the gamma correction method for conventional displays, such as cathode-ray tube (CRT) monitor [65], plasma display panel television (PDP-TV) [66] and thin-film-transistor liquid-crystal display (TFT LCD) [67, 68]. Gamma correction for conventional displays were originally developed to mimic the perceptual response of human vision [69]. The work presented here is a logical continuation of this approach applied to holographic displays.

3.2 Experimental setup

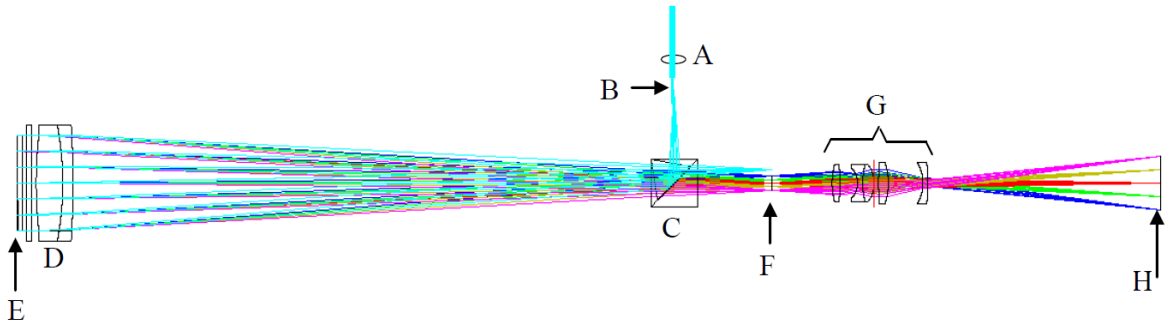


Fig. 3.1 Optical setup of the holographic projection system [16]

The holographic projector used in this experiment is a Fourier projection system developed by Freeman [16], as shown in Fig. 3.1. The design is consisted of a diode-pumped solid-state (DPSS) 532 nm 50mW laser source, focussed down by an aspheric singlet (A), the focus of which becomes the diffraction limited point source (B) for the projector. The beam then passes through a polarizing beam splitter cube (C) to a collimating lens (D), which illuminates the SLM (E). The SLM is a binary phase SXGA-R2 ForthDD ferroelectric Liquid crystal on silicon (LCOS) micro-display with a refresh rate of 1440Hz, a pixel pitch of 13.6 µm and a resolution of $1280px \times 1024px$. An aperture at point (F) spatially filters out the other orders,

leaving only one first order, which is then magnified up by a finite conjugate lens group (G) to produce an image, of the required size, on the screen (H). [16]

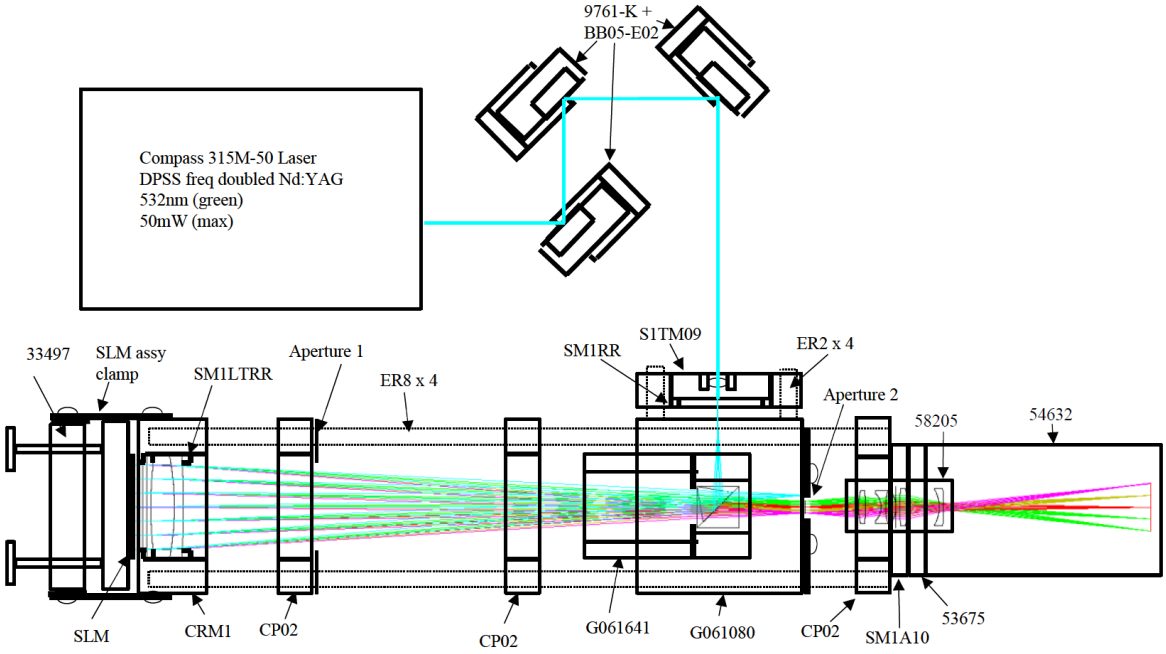


Fig. 3.2 Mechanical components with part numbers of the holographic projection system [16]

The mechanical components are listed in Fig. 3.2 with part numbers. The holograms displayed on the SLM are generated using the OSPR algorithm [46], and as the SLM is a binary-phase modulator, each hologram sub-frame needs to be binary quantised. Then each group of the 24 binary-phase hologram sub-frames are encoded as the 8-bit red, green, blue (RGB) channels of a 24-bit image to interface with the SLM driver electronics. The SLM displays each bit plane sequentially, with ones and zeros mapping to opposing phase modulations at each pixel. The reconstructions were captured using a Canon 550D camera with an EFS 18-55 mm lens. To ensure fair comparison, the camera was set to the same manual setting when comparing each pair of replay fields before and after DPD. It takes $24/1440 = 1/60$s to display all 24 sub-frames on a 1440Hz SLM, so the camera shutter speed was set to $1/30$s to capture all frames twice. The images captured are raw format in RGB colour, which are subsequently converted to grey-scale (using the rgb2gray function in Matlab[70]) when calculating NMSE against the target images, using the equation $NMSE = \frac{\frac{1}{n}*\sum(x-\hat{x})^2}{\sum(x)^2}$, where x is the target, $\hat{x}$ is the measured output and n is the dimension of x and $\hat{x}$.

3.3 Determining the DPD curve

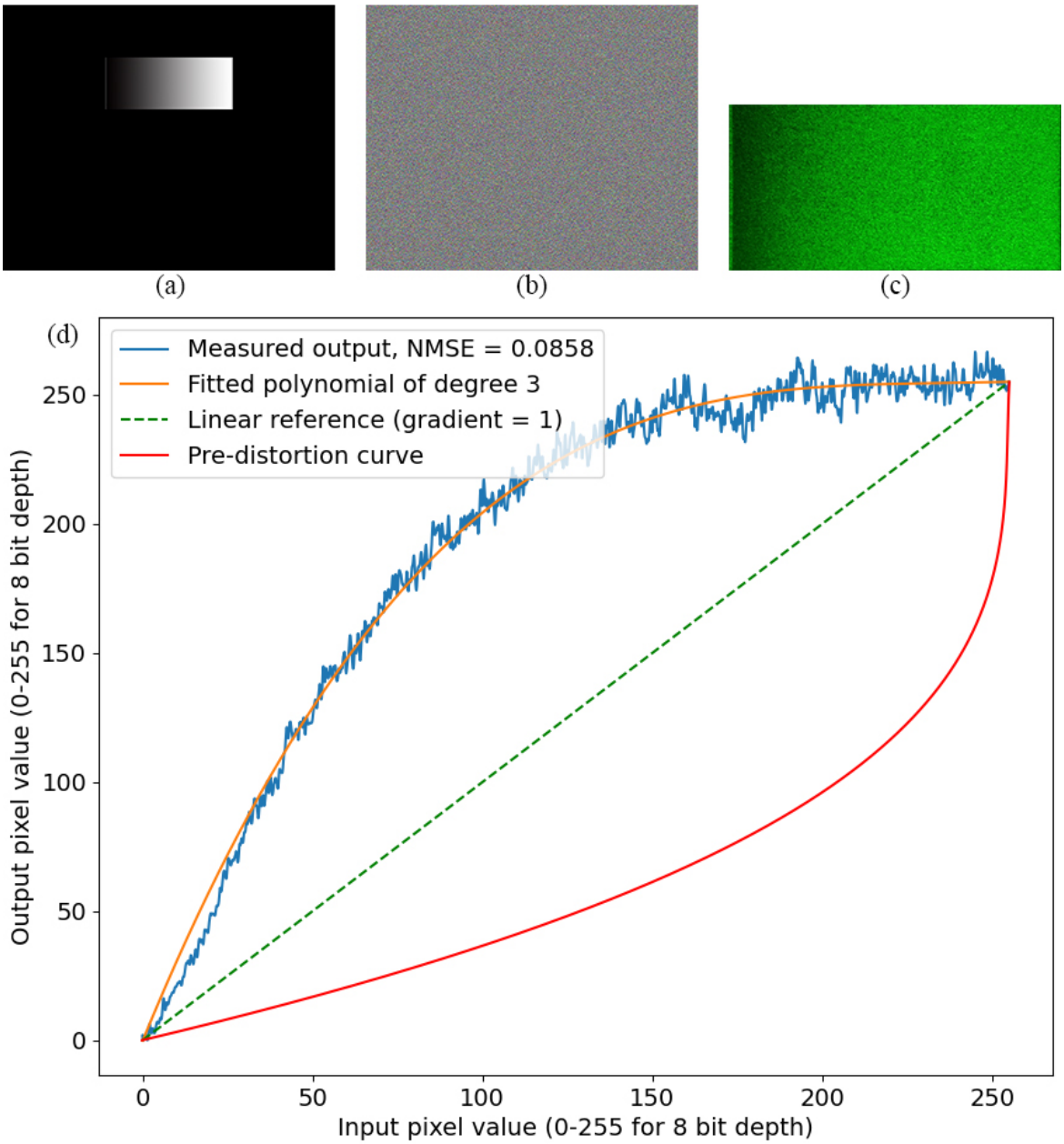


Fig. 3.3 Determining the DPD curve. (a) Input linear grey-scale ramp. (b) Corresponding CGH of (a) with 24-subframe binary phase encoding. (c) Holographic projection replay field of (b). (d) Plot of non-linearity measurement and according pre-distortion curve.

To determine the DPD curve of the holographic projection system, the non-linearity needs to be measured first. The hologram in Fig. 3.3(b) was first generated using OSPR algorithm for the linear grey-scale ramp of pixel value increasing linearly from 0 to 255, as shown in

Fig. 3.3(a), along with a single pixel white (255) strip at the left end as a fiducial marker to demonstrate the beginning of the grey-scale region [17].

The projection output of the linear grey-scale ramp was then captured and cropped as shown in Fig. 3.3(c), from which the non-linearity curve was determined, by averaging each column of pixels in the image and discarding the fiducial marker, forming the blue line in Fig. 3.3(d). A third-order polynomial fit was applied, generating a smoothed non-linearity curve (yellow line in Fig. 3.3(d)).

There exhibits a high degree of non-linearity. By taking the mean of the square of the error between the measured output (blue line) and the linear reference (green dashed line), the normalized mean squared error (NMSE) of the measured output was calculated to be 0.0858. To correct for the non-linearity, the DPD curve (red line) was formed by inverting the smoothed non-linearity curve (yellow line) in Fig. 3.3(d).

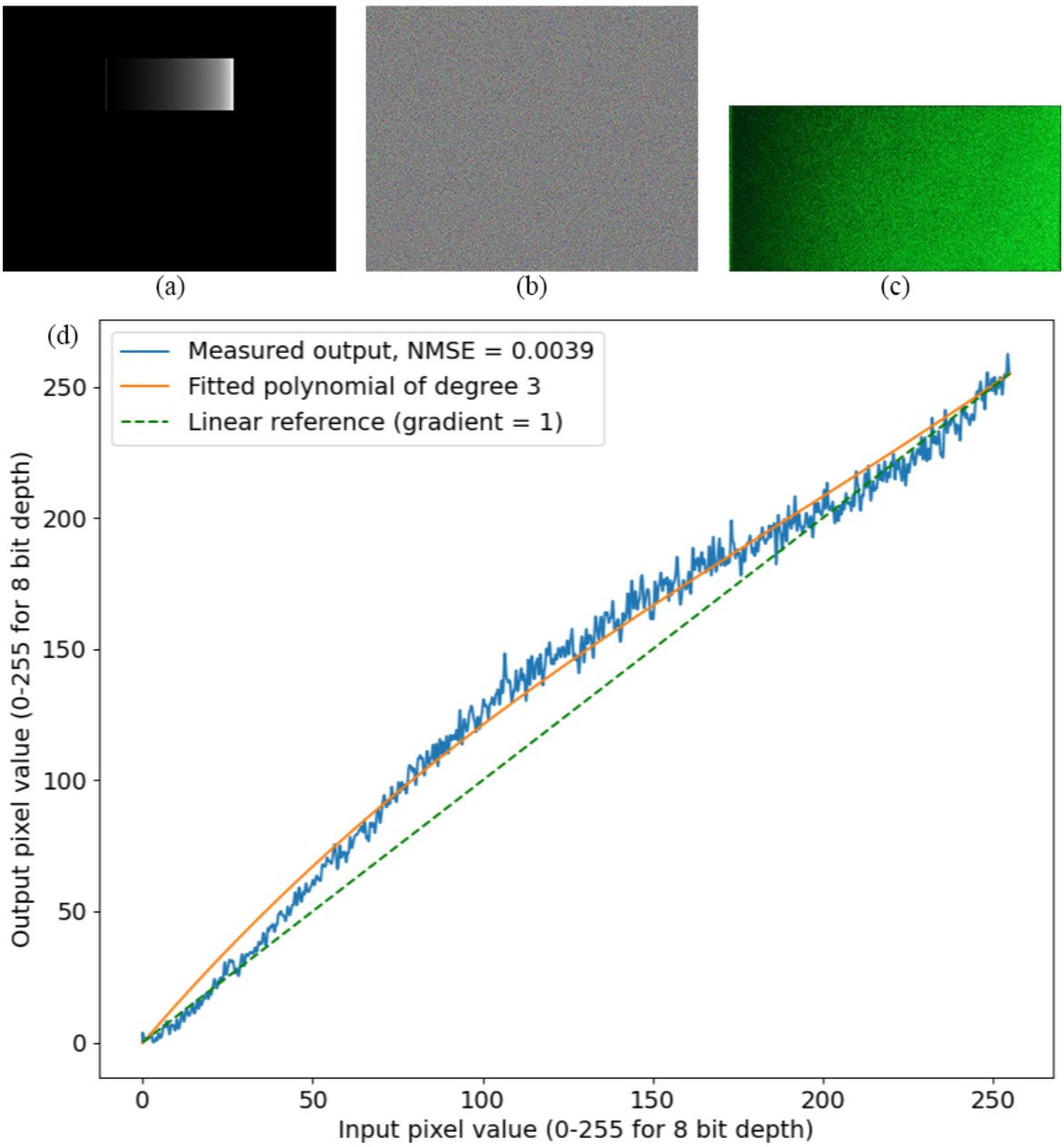


Fig. 3.4 Validation of DPD curve on the grey-scale ramp. (a) Pre-distorted ramp. (b) Corresponding CGH of (a) with 24-subframe binary phase encoding. (c) Holographic projection replay field of (b). (d) Non-linearity measurement after DPD.

Subsequently, the DPD curve (red line in Fig. 3.3(d)) was used to adjust the grey-scale ramp, achieving the pre-distorted grey-scale ramp as shown in Fig. 3.4(a). The according projection output was then captured as shown in Fig. 3.4(c). By using the same method of averaging columns of pixels, the measured output was plotted in Fig. 3.4(d). It can be seen that the

corrected non-linearity was much closer to linear comparing to the original non-linearity, and the NMSE was calculated to be 0.0039.

	NMSE	Percentage
Before DPD	0.0858	100%
After DPD	0.0039	4.55%

Table 3.1 Non-linearity results before and after DPD

Hence, as demonstrated in Table 3.1, DPD achieved a 95.45% reduction in NMSE, which was a significant improvement in non-linearity, therefore the DPD curve measured is validated.

3.4 Applying the DPD Curve

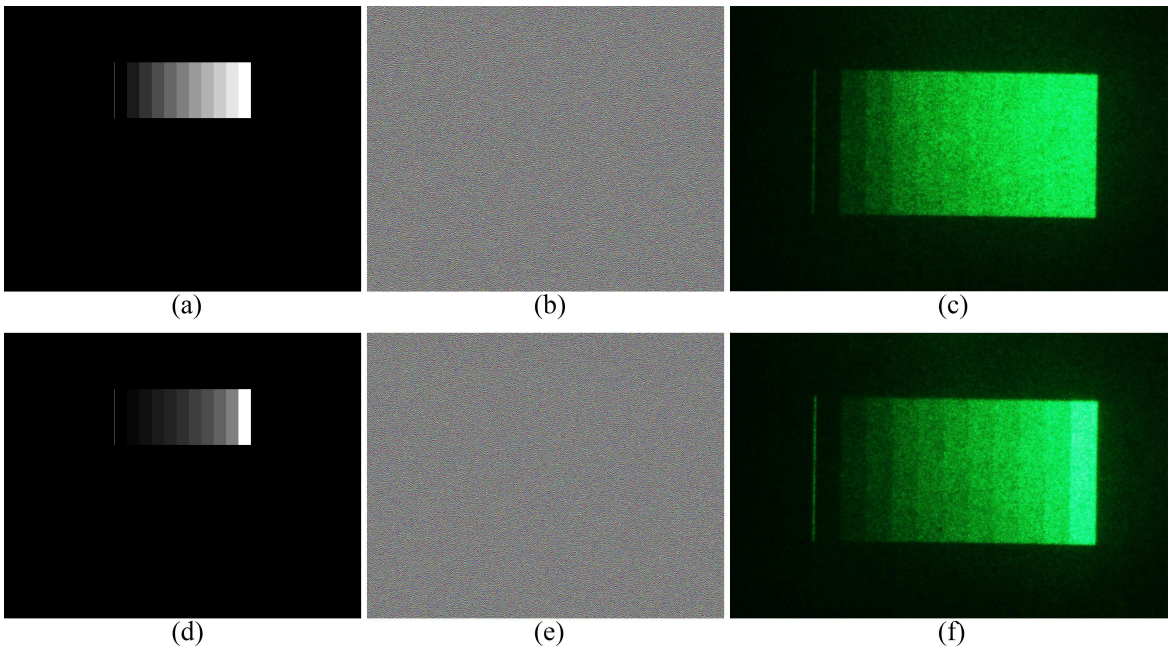


Fig. 3.5 Application of DPD on the 10-step strips. (a) 10 strips with equal step of pixel value. (b) CGH of (a). (c) Holographic projection replay field of (b). (d) After DPD of (a). (e) CGH of (d). (f) Holographic projection replay field of (e).

To qualitatively demonstrate the effectiveness of our approach, we project a simple test pattern of a graduated ramp test pattern consisted of 10-step strips in Fig. 3.5(a), which is commonly employed in gamma-correction calibration of many display systems. As shown

in the projection replay field captured in Fig. 3.5(c), before DPD, the right few strips are barely distinguishable. In comparison, after carrying out DPD, it can be seen that each pair of adjacent strips in Fig. 3.5(f) are much more distinguishable, qualitatively showing the effectiveness of the DPD method.

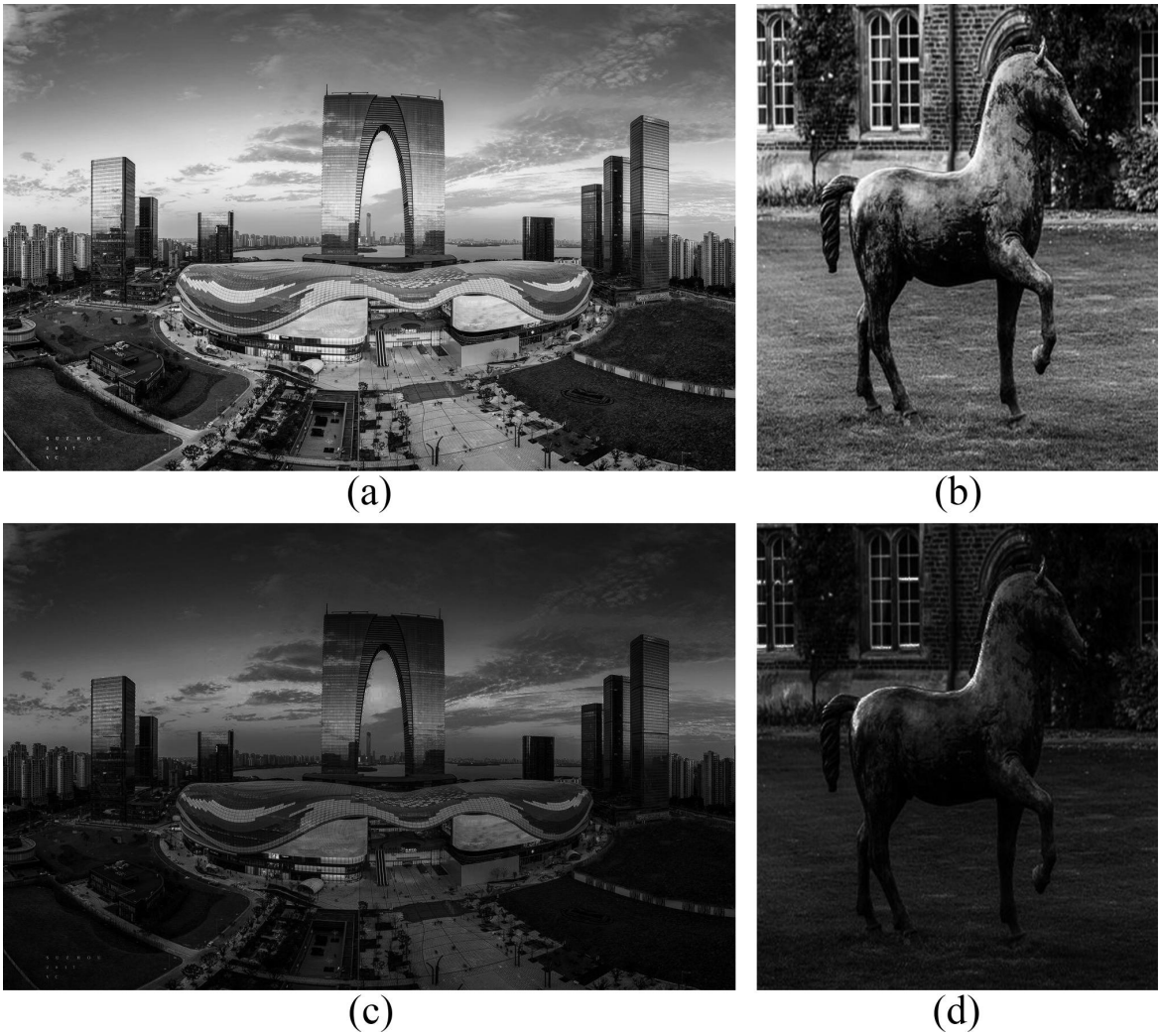


Fig. 3.6 Application of DPD on two sample real-word images. (a) Sample image 1: City Scene [19]. (b) Sample image 2: Horse. (c) Sample image 1 after DPD. (d) Sample image 2 after DPD.

Then the DPD curve was applied to the two sample images as shown in Fig. 3.6 (a) and (b), producing pre-distorted images in Fig. 3.6 (c) and (d). Holograms were generated for each image using the OSPR algorithm and loaded onto the SLM respectively. The according replay fields were captured as shown in Fig. 3.7.

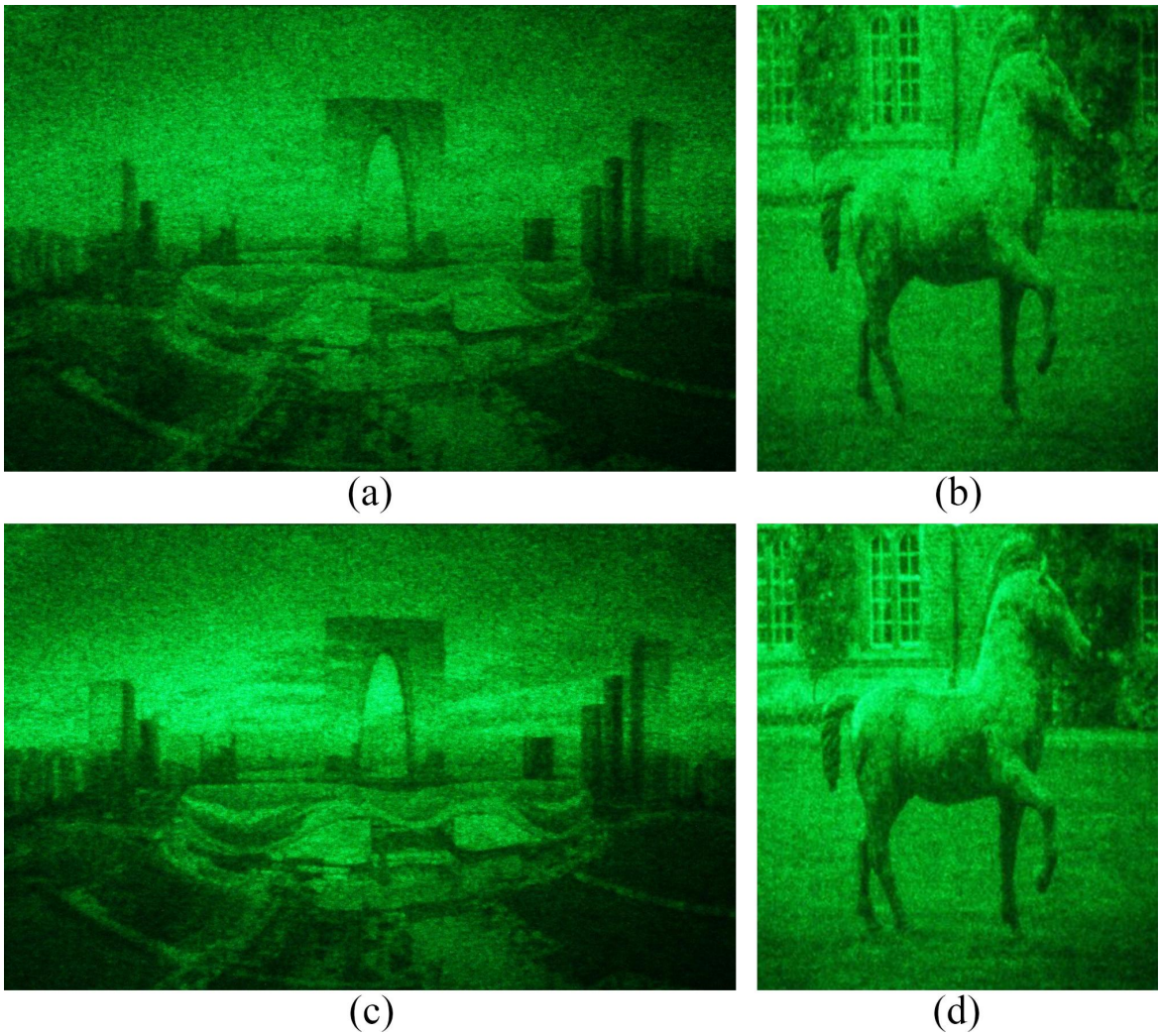


Fig. 3.7 Projection output of the two sample images before and after DPD. (a) Replay field of Sample image 1 before DPD (NMSE=0.06139). (b) Replay field of Sample image 2 before DPD (NMSE=0.04309). (c) Replay field of Sample image 1 after DPD (NMSE=0.04920). (d) Replay field of Sample image 2 after DPD (NMSE=0.03635).

The replay fields of the holographic projection of original images are shown in Fig. 3.7 (a) and (b), and the replay fields of the holographic projection of images after DPD are shown in Fig. 3.7 (c) and (d).

As shown in Fig. 3.7(a), it can be seen that, before DPD, the edges between the buildings and the sky were quite ambiguous, with most detail of the sky being lost. In comparison, after DPD, the replay field in Fig. 3.7(c) provided not only sharper edges between buildings and

the sky, but also more detail of clouds in the sky. The NMSE of the replay field for sample image 1 decreased from 0.06139 to 0.04920, which was a 19.86% reduction.

In Fig. 3.7(b), before DPD, the horse was difficult to distinguish from the background, especially around the horse's back area. But after DPD, as shown in Fig. 3.7(d), contrast has been significantly boosted and the fine detail around this part of the horse is more evident. The NMSE of the replay field for sample image 2 decreased from 0.04309 to 0.03635, which was a 15.64% reduction.

Sample image 1	NMSE	Percentage
Before DPD	0.06139	100%
After DPD	0.04920	80.15%
Sample image 2	NMSE	Percentage
Before DPD	0.04309	100%
After DPD	0.03635	84.36%

Table 3.2 DPD results for sample images

Hence, as summarised in Table 3.2, DPD achieved a 19.86% reduction in NMSE for sample image 1 and a 15.64% reduction in NMSE for sample image 2, quantitatively proving the effectiveness of DPD method for CGH of real-world test images using OSPR algorithm.

Lastly, as the DPD is a one-to-one mapping, the computation time is negligible. In practice, the computational overhead is too small to be measured against randomness between subsequent runs. DPD can also be further accelerated in hardware using a hardware LUT, so that the DPD can be carried out instantly. This approach is widely adopted in gamma correction for displays.

3.5 Summary

The non-linearity between target image and reconstructed image was measured for the overall holographic projection system by projecting a linear grey-scale ramp. Then the DPD curve was applied to the grey-scale ramp and successfully reduced the NMSE by 95.45%. To examine its effectiveness on real world images, the DPD method was applied on two sample images, it was observed that more details were shown in the replay field after DPD, and the NMSE's of the two example images were reduced by 19.86% and 15.64%. As the

DPD is a one-to-one mapping, the extra computation required is negligible. Hence, we have demonstrated the effectiveness of the proposed DPD-OSPR method to improve reconstruction quality on the existing OSPR algorithm while still keeping its ability for real-time holography. The proposed DPD method only needs to be applied once for every holographic projector; however, compared to the original OSPR method, the proposed DPD-OSPR method requires an additional device of a camera, whose accuracy and linearity can affect the effectiveness of the DPD process.

Chapter 4

L-BFGS Optimisation of 2D and 3D CGH

Note: The work in this chapter have been published in Ref. [2, 4]

As previously introduced in Chapter 2, currently available spatial light modulators (SLMs) can only modulate either phase or amplitude, so algorithms are needed to compute amplitude-only or phase-only holograms, among which the phase-only holograms are usually preferred due to their higher energy efficiency, leading to the emergence of the classical phase retrieval algorithms reviewed in Section 2.3. With the developments in modern numerical optimisation methods and computational power, advances in CGH algorithms can be made. This chapter therefore implements the use of numerical optimisation methods for CGHs, and then proposes a novel target image phase optimisation (TIPO) algorithm and also examines the use of sequential slicing (SS) techniques in optimisation algorithms.

4.1 Numerical Optimisation Methods

4.1.1 Optimisation framework

Numerical optimisation methods aim to find an optimal solution which minimise an objective function numerically. They begin with an initial guess of the optimal solution ($\mathbf{x}_0$) and then, after iterations, generate a sequence of gradually improved estimates until they reach a solution [71]. If we have $\mathbf{x}$ as the vector of variables, and denote $f(\mathbf{x})$ as the objective function, which is a function of x we want to minimise, any unconstrained optimisation

problem can be written as:

$$\underset{\mathbf{x}\in R^n}{\text{minimise}} \quad f(\mathbf{x}) \tag{4.1}$$

Numerical optimisation then calculate the optimal solution $\mathbf{x}^*$ iteratively, the iteration is given by:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k \tag{4.2}$$

where the positive scalar α_k is called step length, or sometimes may be referred as 'learning rate' in some contexts, and the vector $\mathbf{p}_k$ is the search direction, which usually takes the form of

$$\mathbf{p}_k = -\mathbf{B}_k^{-1} \nabla f_k \tag{4.3}$$

where $\mathbf{B}_k$ is a nonsingular matrix that varies for different optimisation methods, and ∇f_k is the gradient, which, if unable to evaluate directly, can be approximated by:

$$\nabla f_k \approx \frac{f_{k+1} - f_k}{\mathbf{x}_{k+1} - \mathbf{x}_k}$$
$$\text{where } f_k \text{ denotes } f(\mathbf{x}_k) \tag{4.4}$$

The strategy used to determine $\mathbf{p}_k$ distinguishes one algorithm from another. Most methods make use of the values of f, ∇f and $\nabla^2 f$, and some methods even make use of the accumulated historical values of those derivatives, which are further discussed in Section 4.1.2 - Section 4.1.5.

4.1.2 Gradient Descent

Gradient descent (GD) is a first-order optimisation method, it finds a local minimum by following the negative of the gradient (i.e. the steepest descent direction). The $\mathbf{B}_k$ (in Eq. (4.3)) for gradient descent simply takes the value of $\mathbf{I}$, which is the identity matrix. And the search direction becomes:

$$\mathbf{p}_k = -\nabla f_k \tag{4.5}$$

The steepest descent method is very intuitive: among all possible directions to move away from $\mathbf{x}_k$, the steepest gradient direction is the one which f decreases most rapidly. The advantage of this method is that it requires few computations and memory resources, because it only requires a computation of the first derivative, and it does not require any accumulation

of historical gradients. However, it is a greedy method that only considers the current iteration without any global consideration, so it can be extremely slow on complicated problems. [71]

To work around the disadvantage, a few variants have emerged, such as AdaGrad [72], RMSProp [73] and Adam [74] which combines the advantages of AdaGrad and RMSProp. The Adam method is an iconic variant of GD, often referred to as 'gradient descent with momentum', as the name Adam is derived from 'adaptive moment estimation'. Adam algorithm is based on adaptive estimates of lower-order moments [74]. It computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradients [74].

4.1.3 Newton's Method

Newton's method is a second-order optimisation method. Its search direction is derived from the second-order Taylor series approximation to $f(\mathbf{x}_k + \mathbf{p})$, which is

$$f(\mathbf{x}_k + \mathbf{p}) \approx f_k + \mathbf{p}^T \nabla f_k + \frac{1}{2}\mathbf{p}^T \nabla^2 f_k \mathbf{p} \stackrel{\text{def}}{=} m_k(\mathbf{p}) \tag{4.6}$$

The Newton direction can then be obtained by finding the vector $\mathbf{p}$ that minimises $m_k(\mathbf{p})$. By setting the derivative of $m_k(\mathbf{p})$ to zero, $\mathbf{p}$ can be obtained as:

$$\mathbf{p}_k = -\nabla^2 f_k^{-1} \nabla f_k \tag{4.7}$$

By comparing Eq. (4.7) to Eq. (4.3), it can be seen that the Newton's method has a $\mathbf{B}_k$ of $\nabla^2 f_k$. Unlike the gradient descent method, there is a "natural" step length of 1 associated with the Newton direction, so $\alpha_k = 1$ by default and is only adjusted when it does not produce a satisfactory reduction in the value of f.

The Newton direction is reliable when the difference between the true function $f(\mathbf{x}_k + \mathbf{p})$ and its quadratic model $m_k(\mathbf{p})$ is not too large. Methods that use the Newton direction have a fast rate of local convergence, typically quadratic. After a neighbourhood of the solution is reached, convergence to high accuracy often occurs in just a few iterations. The main drawback of the Newton direction is the need for the Hessian $\nabla^2 f_k$. Explicit computation of this matrix of second derivatives can sometimes be a cumbersome, error-prone, and expensive process. [71]

4.1.4 Quasi-Newton Method

Quasi-Newton method provides an attractive alternative to Newton's method, in that it does not require computation of the Hessian and yet still attains a linear rate of convergence. In place of the true Hessian $\nabla^2 f_k$, they use an approximation $\mathscr{H}_k \stackrel{\text{def}}{=} \mathbf{B}_k^{-1}$, which is updated after each step to take account of the additional knowledge gained during the step. The updates make use of the fact that changes in the gradient provide information about the second derivative of f along the search direction. The most popular quasi-Newton algorithm is the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method, named for its discoverers Broyden, Fletcher, Goldfarb, and Shanno. [71]

The process of the BFGS method is shown below:

$$denote \begin{cases} \mathscr{H}_k & = \mathbf{B}_k^{-1} \\ \mathbf{p}_k & = -\mathscr{H}_k \nabla f_k \end{cases} \tag{4.8}$$

$$Initiate\ \mathscr{H}_0 \leftarrow \frac{\mathbf{y}_k^T \mathbf{s}_k}{\mathbf{y}_k^T \mathbf{y}_k} \mathbf{I} \tag{4.9}$$

$$update\ \mathscr{H}_{k+1} = (\mathbf{I} - \rho_k \mathbf{s}_k \mathbf{y}_k^T)\mathscr{H}_k(\mathbf{I} - \rho_k \mathbf{y}_k \mathbf{s}_k^T) + \rho_k \mathbf{s}_k \mathbf{s}_k^T \tag{4.10}$$

$$where \begin{cases} \mathbf{s}_k & = \mathbf{x}_{k+1} - \mathbf{x}_k \\ \mathbf{y}_k & = \nabla f_{k+1} - \nabla f_k \\ \rho_k & = \frac{1}{\mathbf{y}_k^T \mathbf{s}_k} \end{cases} \tag{4.11}$$

The algorithm is robust, and its rate of convergence is linear, which is fast enough for most practical purposes. Even though Newton's method converges more rapidly (that is, quadratically), its cost per iteration usually is higher, because of its need for second derivatives and solution of a linear system. The drawback is that, it is not directly applicable to large optimisation problems because $\mathscr{H}_k$'s are usually dense, requiring large storage and computational requirements. [71]

4.1.5 Large Scale Quasi-Newton Method: Limited Memory BFGS (L-BFGS)

L-BFGS algorithm [75] modifies the technique described in Section 4.1.4 to obtain Hessian approximations that can be stored compactly in just a few vectors of length n, where n is the number of unknowns in the problem. The main idea of this method is to use the curvature information from only the most recent iterations to construct the Hessian approximation.

Curvature information from earlier iterations, which is less likely to be relevant to the actual behaviour of the Hessian at the current iteration, is discarded in the interest of saving storage. [71]

Denoting $\mathbf{V}_k = \mathbf{I} - \rho_k \mathbf{y}_k \mathbf{s}_k^T$, Eq. (4.10) can be written as:

$$\mathscr{H}_{k+1} = \mathbf{V}_k^T \mathscr{H}_k \mathbf{V}_k + \rho_k \mathbf{s}_k \mathbf{s}_k^T \tag{4.12}$$

The inverse Hessian approximation $\mathscr{H}_k$ will generally be dense, so that the cost of storing and manipulating it is prohibitive when the number of variables is large. To circumvent this problem, we store a modified version of $\mathscr{H}_k$ implicitly, by storing a certain number (say, m) of the vector pairs $\{\mathbf{s}_i, \mathbf{y}_i\}$ used in the Eq. (4.10) and Eq. (4.11). The product $\mathscr{H}_k \nabla f_k$ can be obtained by performing a sequence of inner products and vector summations involving ∇f_k and the pairs $\{\mathbf{s}_i, \mathbf{y}_i\}$. After the new iterate is computed, the oldest vector pair in the set of pairs $\{\mathbf{s}_i, \mathbf{y}_i\}$ is replaced by the new pair $\{\mathbf{s}_k, \mathbf{y}_k\}$ obtained from the current step (Eq. (4.11)). In this way, the set of vector pairs includes curvature information from the m most recent iterations. Practical experience has shown that modest values of m (between 3 and 20, say) often produce satisfactory results. We now describe the updating process in a little more detail. At iteration k, the current iterate is $\mathbf{x}_k$ and the set of vector pairs is given by $\{\mathbf{s}_i, \mathbf{y}_i\}$ for $i = k-m, \ldots, k-1$. We first choose some initial Hessian approximation $\mathscr{H}_k^0$ (in contrast to the standard BFGS iteration, this initial approximation is allowed to vary from iteration to iteration) and find by repeated application of Eq. (4.10) that the L-BFGS approximation $\mathscr{H}_k$ satisfies the following formula: [71]

$$\begin{aligned}
\mathscr{H}_k = {} & (\mathbf{V}_{k-1}^T \cdots \mathbf{V}_{k-m}^T) \mathscr{H}_k^0 (\mathbf{V}_{k-m} \cdots \mathbf{V}_{k-1}) \\
& + \rho_{k-m} (\mathbf{V}_{k-1}^T \cdots \mathbf{V}_{k-m+1}^T) \mathbf{s}_{k-m} \mathbf{s}_{k-m}^T (\mathbf{V}_{k-m+1} \cdots \mathbf{V}_{k-1}) \\
& + \rho_{k-m+1} (\mathbf{V}_{k-1}^T \cdots \mathbf{V}_{k-m+2}^T) \mathbf{s}_{k-m+1} \mathbf{s}_{k-m+1}^T (\mathbf{V}_{k-m+2} \cdots \mathbf{V}_{k-1}) \\
& + \cdots \\
& + \rho_{k-1} \mathbf{s}_{k-1} \mathbf{s}_{k-1}^T
\end{aligned} \tag{4.13}$$

From this expression we can derive a recursive procedure (Algorithm 8) to compute the product $\mathscr{H}_k \nabla f_k$ efficiently.

Algorithm 8 L-BFGS two-loop recursion [71]

$\mathbf{q} \leftarrow \nabla f_k$
for $i = k-1, k-2, ..., k-m$ **do**
  $\alpha_i \leftarrow \rho_i \mathbf{s}_i^T \mathbf{q}$
  $\mathbf{q} \leftarrow \mathbf{q} - \alpha_i \mathbf{y}_i$
end for
$\mathbf{r} \leftarrow \mathscr{H}_k^0 \mathbf{q}$
for $i = k-m, k-m+1, ..., k-1$ **do**
  $\beta \leftarrow \rho_i \mathbf{y}_i^T \mathbf{r}$
  $\mathbf{r} \leftarrow \mathbf{r} + \mathbf{s}_i(\alpha_i - \beta)$
end for
Step with $\mathbf{p}_k \leftarrow -\mathscr{H}_k \nabla f_k = -\mathbf{r}$

Apart from being inexpensive, L-BFGS has the advantage that the multiplication by the initial matrix $\mathscr{H}_k^0$ is isolated from the rest of the computations, allowing this matrix to be chosen freely and to vary between iterations. A method for choosing $\mathscr{H}_k^0$ that has proved effective in practice is to use the same as BFGS as stated in Eq. (4.9). [71]

4.2 Phase-only Hologram Optimisation

To implement the optimisation methods listed in Section 4.1 on the CGH, firstly the optimisation framework needs to be adapted. The objective of CGH is to find the phase hologram (**H**) that has the optimal reconstruction (**R**) matching the target image (**T**), which can be formulated as minimising the difference between **R** and **T** by varying **H**, leading to the mathematical expression below:

$$\arg\min_{\mathbf{H}} Loss(\mathbf{T}, \mathbf{R}) \tag{4.14}$$

where '*Loss*' denotes an error function quantifying the difference between **T** and **R**, and 'arg' returns the argument (**H**) instead of the error value. For the *Loss* function, the classic error function mean-squared error (MSE) [57] is selected, with its definition as shown below:

$$MSE(\mathbf{T}, \mathbf{R}) = \frac{1}{X \times Y} \sum_{x=1}^{X} \sum_{y=1}^{Y} (\mathbf{T}_{x,y} - \mathbf{R}_{x,y})^2 \tag{4.15}$$

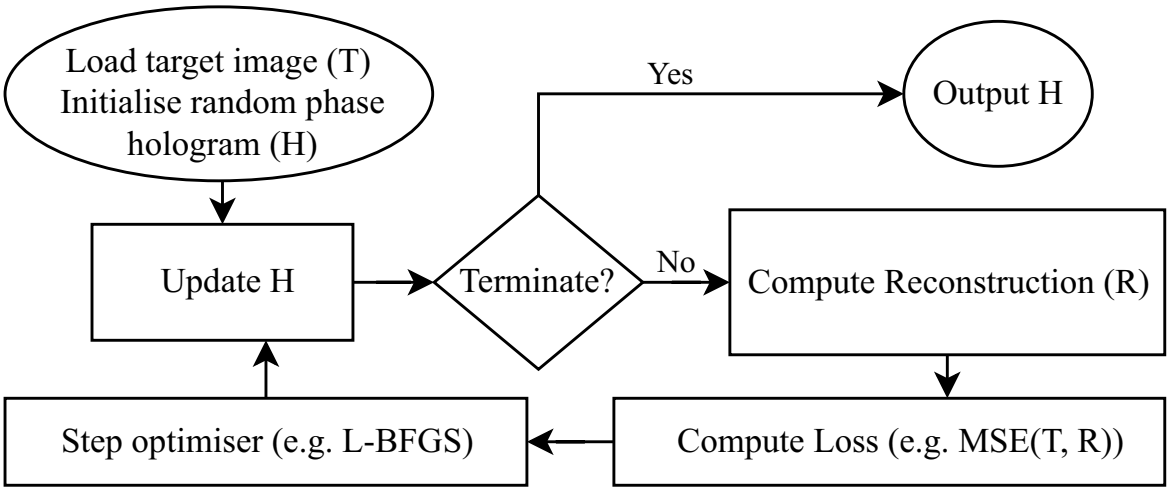


Fig. 4.1 Flowchart of the optimisation process

The optimisation process for generating a phase hologram to reconstruct a target image is depicted in Fig. 4.1. The process begins by loading the target image (**T**) and initializing a random phase hologram (**H**). Using this initial phase hologram, a reconstruction (**R**) of the target image is computed via either the Fraunhofer or Fresnel propagation equation introduced in Section 2.2.2 based on the distance needed. The difference between the target image (**T**) and the reconstruction (**R**) is quantified by computing the loss, such as the MSE in Eq. (4.15). An optimiser (such as the GD, Adam or L-BFGS algorithms mentioned in Section 4.1) then updates the phase hologram based on the computed loss. This iterative process of reconstruction, loss calculation, and hologram update continues until a termination condition is met, usually a fixed total number of iterations, or when optimisation has reached convergence. The final optimised phase hologram (**H**) is then outputted.

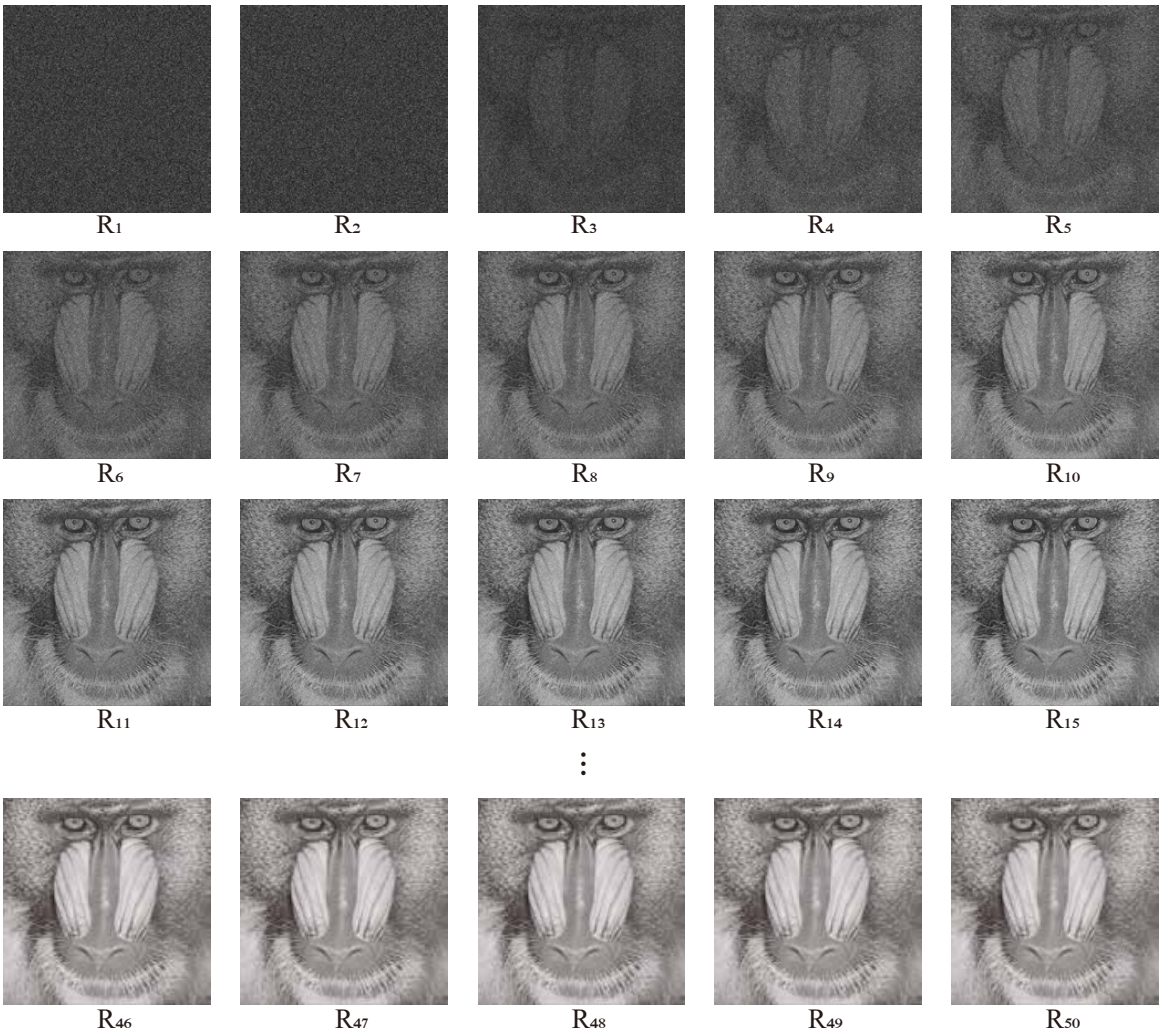


Fig. 4.2 Reconstructions at each iteration of the L-BFGS optimisation

The optimisation flowchart in Fig. 4.1 is run on the example target image in Fig. 2.11 for a total of 50 iterations. The reconstruction ($\mathbf{R}$) at each iteration is listed in Fig. 4.2, with iterations 16 to 45 omitted to save space. The list of reconstructions demonstrate visually how the optimisation converges to the resulting reconstruction ($\mathbf{R}_{50}$) which has a very good quality. It shows that the proposed method using the L-BFGS algorithm to generate phase hologram is effective.

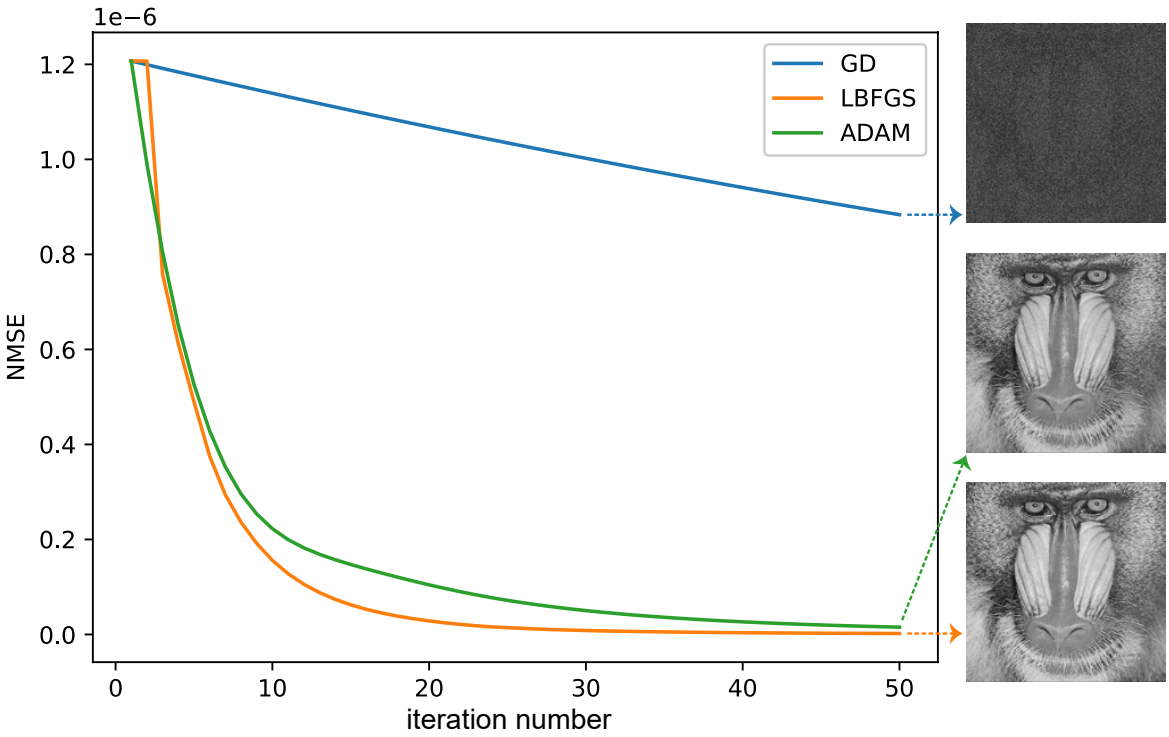


Fig. 4.3 Convergence plot for comparison between the GD, Adam and L-BFGS optimisations

Then the proposed method of using L-BFGS algorithm to generate phase hologram is quantitatively compared among the existing methods using GD and Adam algorithms, by plotting the normalised mean-squared error (NMSE) defined in Eq. (2.34) against the iteration number.

The proposed method using L-BFGS algorithm (the orange line in Fig. 4.3) stagnates for the first two iterations as it needs to estimate the Hessian as explained in Section 4.1.5. After the brief stagnation, the L-BFGS converges quickly, surpassing the existing methods in the literature using GD and Adam optimiser, corresponding to the blue line and the green line in Fig. 4.3 respectively. The final reconstructions are shown aside the NMSE plots, from which it can be seen that the L-BFGS algorithm produces a reconstruction of comparable quality to the Adam algorithm, and both of them are much better than the GD algorithm. Such observation matches the final NMSE values. The NMSE value of the L-BFGS algorithm (the orange line in Fig. 4.3) stays lower than the Adam optimiser (the blue line in Fig. 4.3) after the 3^{rd} iteration, and reaches convergence earlier at the 30^{th} iteration. In summary, the L-BFGS algorithm was proven to be able to optimise a phase-only hologram for a target

image, and is much better than the GD optimiser and converges quicker than the Adam optimisation algorithm.

4.3 Target Image Phase Optimisation (TIPO)

This section proposes a novel method of optimising the phase of the target image instead of optimising the phase of the hologram as previously introduced in Section 4.2, under the same 'phase only' constraint of the hologram. The objective is to find an optimal phase profile to be attached to the target image, so that its inverse propagation to the hologram plane produces a hologram whose phase has a reconstruction that best matches the target image. As it optimises the phase of the target image instead of the traditional way optimising the phase of the hologram, this method is named as Target Image Phase Optimisation (TIPO). A flowchart is drawn in Fig. 4.4 to clarify the detailed operation of the proposed TIPO algorithm.

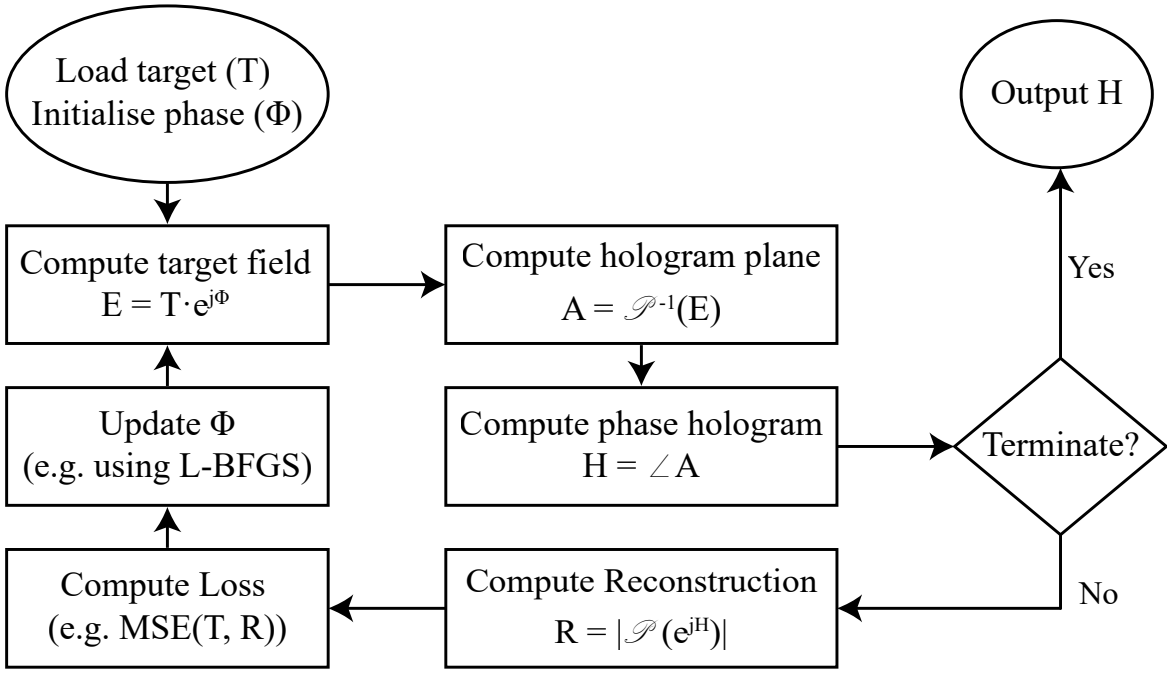


Fig. 4.4 TIPO flowchart

The flowchart in Fig. 4.4 outlines the TIPO algorithm process for generating a phase-only hologram (**H**) to reconstruct a target image (**T**). The procedure starts with loading the target image and initializing the phase (Φ). The complex target field (**E**) is then computed from the target image and the target image phase ($\mathbf{E} = \mathbf{T} \cdot e^{j\Phi}$). Next, the hologram aperture (**A**) is computed by applying the inverse propagation ($\mathscr{P}^{-1}$) to the target field, where $\mathscr{P}$ is chosen

to be the Fraunhofer diffraction equations in Section 2.2.2. The phase-only hologram (**H**) is then derived from the angle of the complex hologram aperture ($\mathbf{H} = \angle\mathbf{A}$). Subsequently, a reconstruction (**R**) is computed using the forward propagation ($\mathbf{R} = |\mathscr{P}(e^{jH})|$). The loss, such as the Mean Squared Error (MSE), is calculated between the target image (**T**) and the reconstruction (**R**). An optimiser (e.g. SGD or L-BFGS) then updates the phase (Φ) based on the computed loss. This iterative process of computing the target field, hologram plane, phase hologram, reconstruction, and loss calculation continues, followed by phase updates, until a termination condition is met, which is usually set for a fixed number of iterations. Upon convergence, the final optimised phase hologram (**H**) is produced.

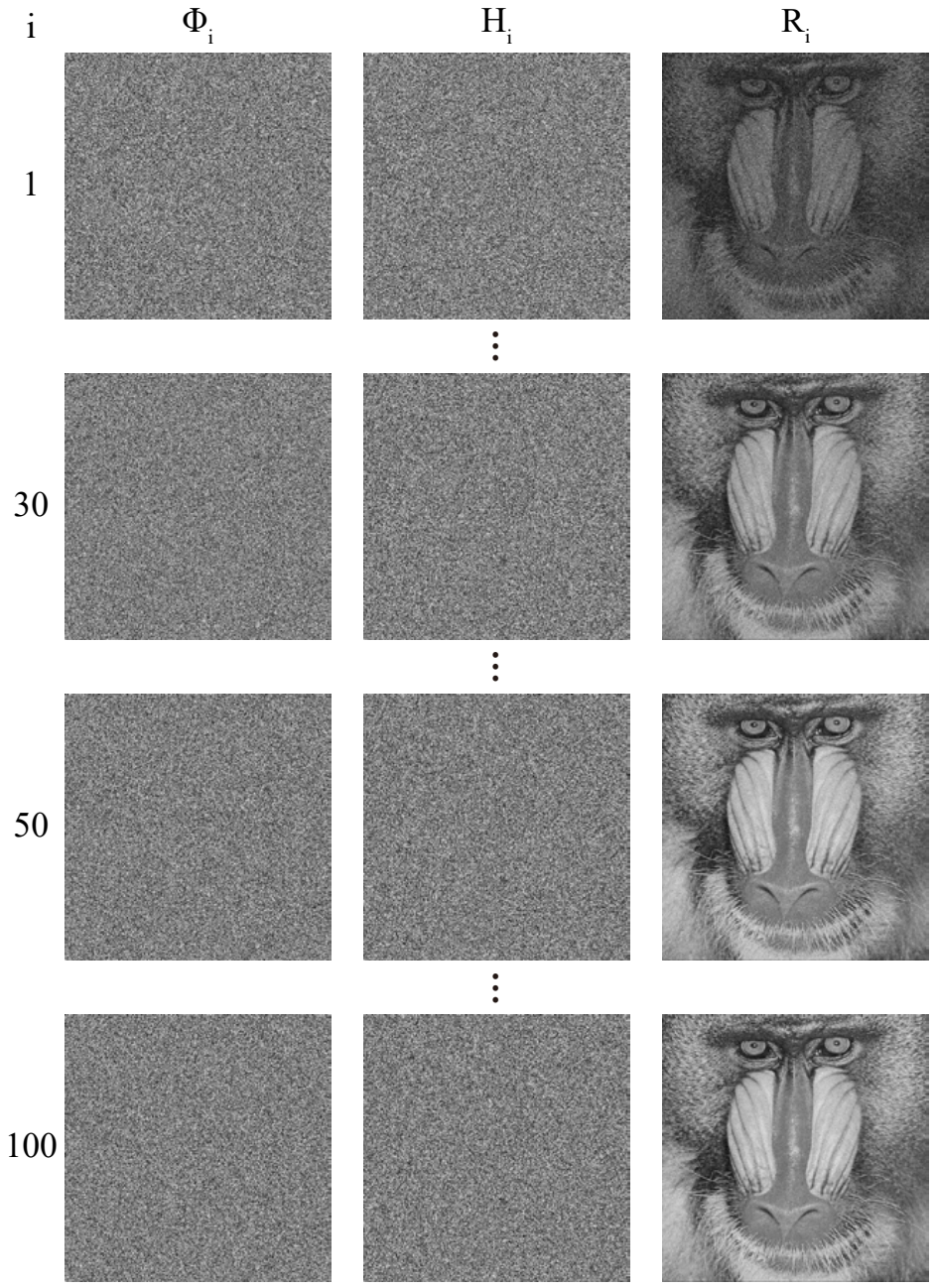


Fig. 4.5 TIPO iterations on the mandrill target

The results for an example run of the TIPO algorithm on the mandrill target (in Fig. 2.11) with the total number of iterations set to 100 are shown in Fig. 4.5 (only iterations number 1, 30, 50 and 100 are shown for space saving). As the first hologram is computed by the inverse propagation ($\mathscr{P}^{-1}$) of the target image with an random phase, the reconstruction ($\mathbf{R}_1$) is

already showing the target image. Then as the iterations continue, the reconstruction quality gets better, proving the effectiveness of the proposed TIPO algorithm.

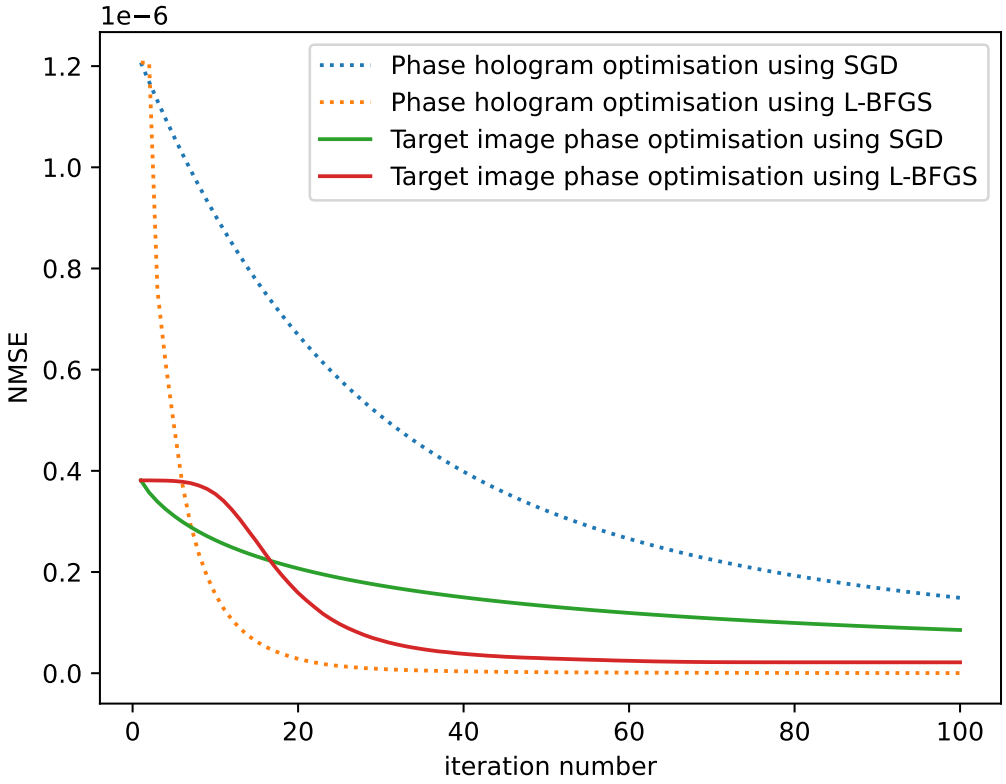


Fig. 4.6 TIPO convergence plot

To quantitatively analyse the results, the convergence of the TIPO algorithm is plotted in Fig. 4.6, where the NMSE between the reconstruction and the target image are plotted against the number of iterations. Both SGD and L-BFGS optimisers are run for the TIPO algorithm (corresponding to the green and red line in Fig. 4.6 accordingly), and are compared against the regular phase hologram optimisation algorithm in Section 4.2 (corresponding to the dotted blue line and the dotted orange line in Fig. 4.6 respectively). The TIPO algorithms using both SGD and L-BFGS optimisers start with a lower NMSE of the reconstruction as their holograms in the first iteration are extracted from the inverse propagation of the target image, instead of pure random holograms as done in the regular phase hologram optimisation. However, the regular phase hologram optimisation using L-BFGS optimiser quickly surpassed the TIPO algorithm within 10 iterations, and reached the lowest reconstruction error. For the SGD optimiser, the TIPO method (green line) has an significant improvement than the

regular phase hologram optimisation method (dotted blue line). The two TIPO methods (in solid lines) lie between the two regular phase hologram optimisation methods (in dotted lines), proving that although not being the best, the TIPO method is still an effective method for phase hologram generation.

4.4 Multi-Depth Phase-Only Hologram Optimisation

A search in the literature has found some recent work that compute CGH for 3D targets using numerical optimisation methods [31, 32, 34, 33], but speed and quality are still the major challenges. They either evaluate the error of reconstructions against the entire 3D target, which is time-consuming, or evaluate the hologram for each plane and then sum the holograms, which introduces quality degradation.

This section reviews the existing multi-depth hologram optimisation methods, and proposes the novel use of sequential slicing (SS) techniques during the optimisation process, which only evaluates the loss for a single slice at each iteration. The proposed technique aims for a quicker hologram generation with proper overall quality and low quality imbalance across the multiple depths.

4.4.1 Methods

The optimiser used for CGH in this section is the L-BFGS optimiser as introduced in Section 4.1.5, with the GD optimiser in Section 4.1.2 also implemented as a reference. The phase-only constraint of CGH is applied by fixing a constant amplitude of the hologram, while keeping its phase varying and being the argument of optimisation ($\mathbf{x}$ in Eq. (4.1)).

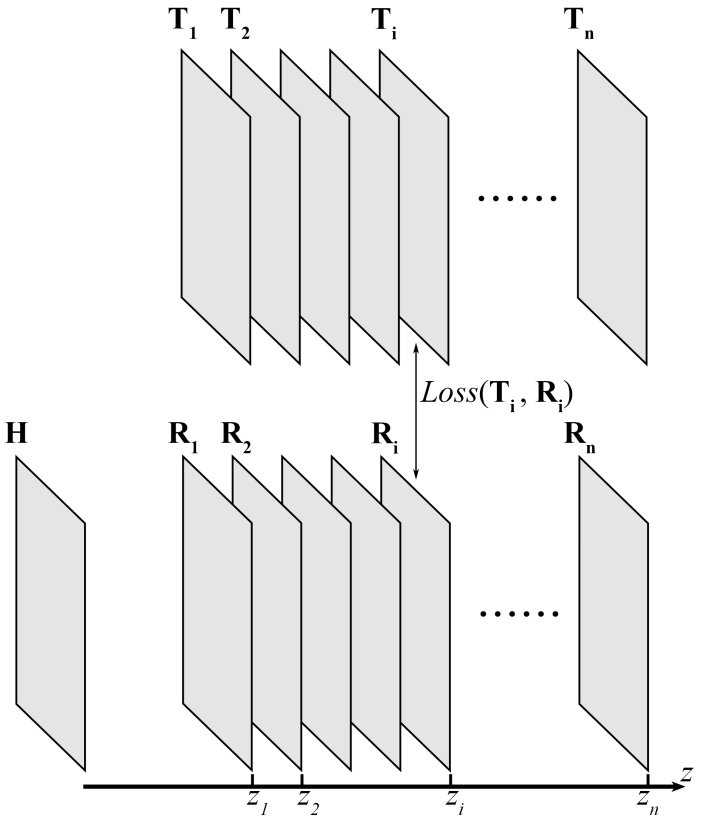


Fig. 4.7 Loss between the multi-depth targets ($\mathbf{T}_1$ to $\mathbf{T}_n$) and the reconstructions ($\mathbf{R}_1$ to $\mathbf{R}_n$) of hologram $\mathbf{H}$

As shown in Fig. 4.7, the multi-depth target is set up as a collection of n slices ($\mathbf{T}_1$ to $\mathbf{T}_n$), each slice $\mathbf{T}_i$ is at a distance z_i to the hologram plane. And for the hologram $\mathbf{H}$, its reconstruction at each distance z_i is computed using Fresnel diffraction formula in Eq. (2.29), which is labelled as the propagation function $\mathscr{P}(\mathbf{H}, z_i)$.

Loss Functions

To formulate an objective function $f(\mathbf{x})$ in Eq. (4.1), we need to quantify the difference between each target slice ($\mathbf{T}_i$) and the respective reconstruction ($\mathbf{R}_i$) numerically, which is denoted as $Loss(\mathbf{T}_i, \mathbf{R}_i)$ in Fig. 4.7. In addition to the mean squared error (MSE) [57] previously used in Section 4.2, the cross entropy (CE) [76] and relative entropy (RE) [77] are also implemented. To adapt the loss functions for two-dimensional (2D) target image $\mathbf{T}_i$ and reconstructed image $\mathbf{R}_i$ of dimension $X \times Y$, the loss functions are adapted as shown in Eq. (4.16) to Eq. (4.18).

$$MSE(\mathbf{T}_i, \mathbf{R}_i) = \frac{1}{X \times Y} \sum_{x=1}^{X} \sum_{y=1}^{Y} (\mathbf{T}_{i;x,y} - \mathbf{R}_{i;x,y})^2 \tag{4.16}$$

$$CE(\mathbf{T}_i, \mathbf{R}_i) = -\sum_{x=1}^{X} \sum_{y=1}^{Y} \mathbf{T}_{i;x,y} \log(\mathbf{R}_{i;x,y}) \tag{4.17}$$

$$RE(\mathbf{T}_i, \mathbf{R}_i) = -\sum_{x=1}^{X} \sum_{y=1}^{Y} \mathbf{T}_{i;x,y} \log\left(\frac{\mathbf{R}_{i;x,y}}{\mathbf{T}_{i;x,y}}\right) \tag{4.18}$$

MSE is a traditional metric averaging the squared error between the target and observed values. CE, adapted as shown in Eq. (4.17), is usually used in classification problems, such as language modelling [78]. RE, also called Kullback-Leibler divergence (usually denoted as $D_{KL}(P||Q)$, but is denoted as RE here for uniformity), is adapted as shown in Eq. (4.18). RE is usually used to measure how much a probability distribution P is different from another probability distribution Q. Both CE and RE are usually computed between the true probabilistic distribution and the predicted probabilistic distribution. While the images are not probability distributions, the pixel values can be normalized to decimal numbers in the range from 0 to 1 so that CE and RE can be applied.

Sum-of-Loss (SoL) technique

The conventional technique to compute 3D CGH is to sum the losses for each slice at each iteration during optimisation, which is called the Sum-of-Loss (SoL) method here. At every iteration, it computes the full 3D reconstructions ($\mathbf{R}_1, \cdots, \mathbf{R}_n$) of the hologram $\mathbf{H}$ at every distance z_i, and then sum the losses between each pair of reconstruction $\mathbf{R}_i$ and target image $\mathbf{T}_i$. The mathematical expression of the SoL technique's optimisation objective is shown below:

$$\text{SoL:} \quad \arg\min_{\mathbf{H}} \sum_{i=1}^{n} Loss(\mathbf{T}_i, \mathbf{R}_i) \tag{4.19}$$

The SoL method requires a total of n Fourier Transforms to fully evaluate the hologram at each step, making it computationally heavy.

Sum-of-Hologram (SoH) technique

Another technique for 3D CGH is to compute the complex sum of the sub-holograms $\mathbf{H}_i$ generated for each target slice $\mathbf{T}_i$ to form a total hologram based on the principle of super-

position, which is called the Sum-of-Hologram (SoH) method here. The SoH technique's mathematical expression is shown in Eq. (4.20) below:

$$\text{SoH:} \quad \angle \sum_{i=1}^{n} e^{j \cdot \arg\min_{\mathbf{H}_i} Loss(\mathbf{T}_i, \mathbf{R}_i)} \tag{4.20}$$

The sub-hologram $\mathbf{H}_i$ for each slice number i is computed independently with the optimisation objective of $\min_{\mathbf{H}_i} Loss(\mathbf{T}_i, \mathbf{R}_i)$, the arguments of which are then summed in a complex manner, as each phase hologram is the angle, the exponential operator is needed to turn angles into complex numbers. Lastly, the angle of the complex sum is taken so that it can meet the 'phase-only' constraint.

If using a fixed number of iterations per sub-hologram, the total computation scales up linearly with the number of slices n. The SoH method's advantage is its ease of implementation, that any existing single slice CGH algorithm can be quickly converted to multi-depth 3D CGH algorithm. Its major disadvantage for phase-only hologram generation is that, the final summing of sub-holograms will result in a non-uniform amplitude hologram, and taking the phase of which will result in discarding the amplitude information of the summed hologram, leading to deprecations in reconstructions quality. And also, the SoH method suffers from the defocusing effect between the each slice to another, causing additional noise.

Sequential Slicing (SS) technique

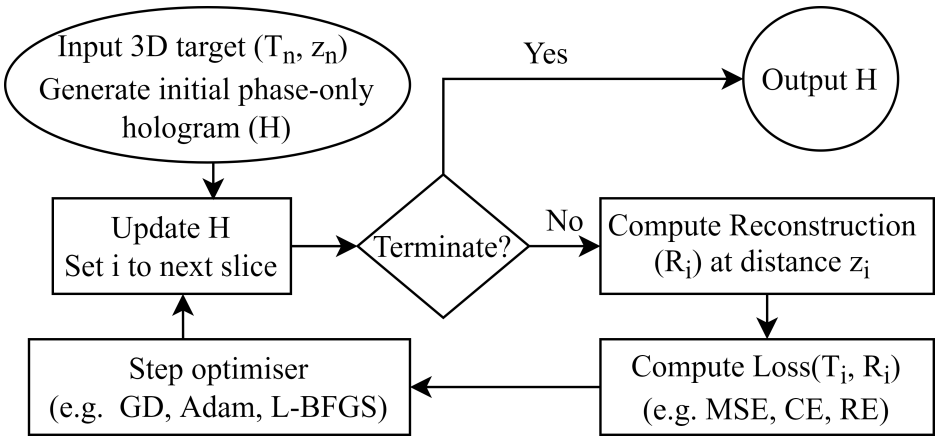


Fig. 4.8 Optimisation of CGH with sequential slicing (SS) flowchart

This section proposes the novel CGH optimisation with sequential slicing (SS) technique, as shown in the flowchart in Fig. 4.8, that only computes the loss for a single slice at each

iteration (between a reconstruction $\mathbf{R}_i$ at a single distance z_i and its according target slice $\mathbf{T}_i$), where i sweeps through the multi-layer 3D target sequentially when the algorithm iterates. When the final slice is reached ($i = n$), it goes back to the first slice ($i \leftarrow 1$). The proposed method only needs to carry out one Fourier Transform at each iteration, and the number of iterations does not need to scale up with n. So it is expected to be much quicker than SoL and SoH techniques while producing a proper resulting hologram.

4.4.2 Results

CGH for a 4-slice target consisted of letters 'A, B, C, D'

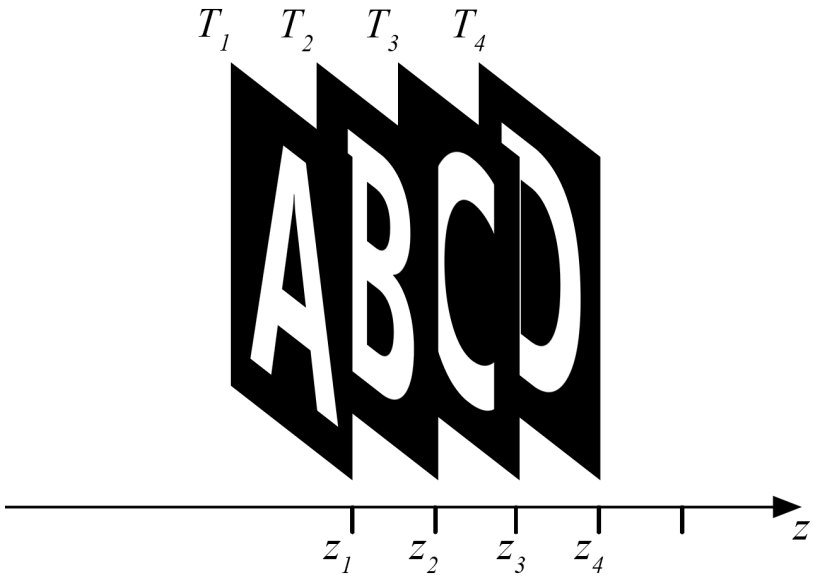


Fig. 4.9 Layout of the 4-slice target ($z_1 = 1$ *cm*, $z_2 = 2$ *cm*, $z_3 = 3$ *cm*, $z_4 = 4$ *cm*)

The first example 3D target used consisted of 4 slices made from letters 'A', 'B', 'C', 'D', each having 512×512 pixels. The positions of the four slices range from 1 *cm* to 4 *cm* with 1 *cm* gap between each other (i.e. $z_i = i$ *cm*). The overall layout is shown in Fig. 4.9.

As there are two optimisers (GD and L-BFGS), three techniques (SoL, SoH and SS) and three loss functions (MSE, CE, RE) in consideration, they can form a total of 18 combinations. In order to control the number of variables, all 18 combinations were set to start from the same initial random hologram and run for the same amount of 100 iterations for the optimisation of each hologram, on the same laptop of model ASUS ROG Zephyrus M16, which has a CPU of model i7-11800H and a GPU of model RTX3060. For the L-BFGS algorithm, the gradient history of size 10 ($m = 10$ in Algorithm 8) was used for all techniques and loss

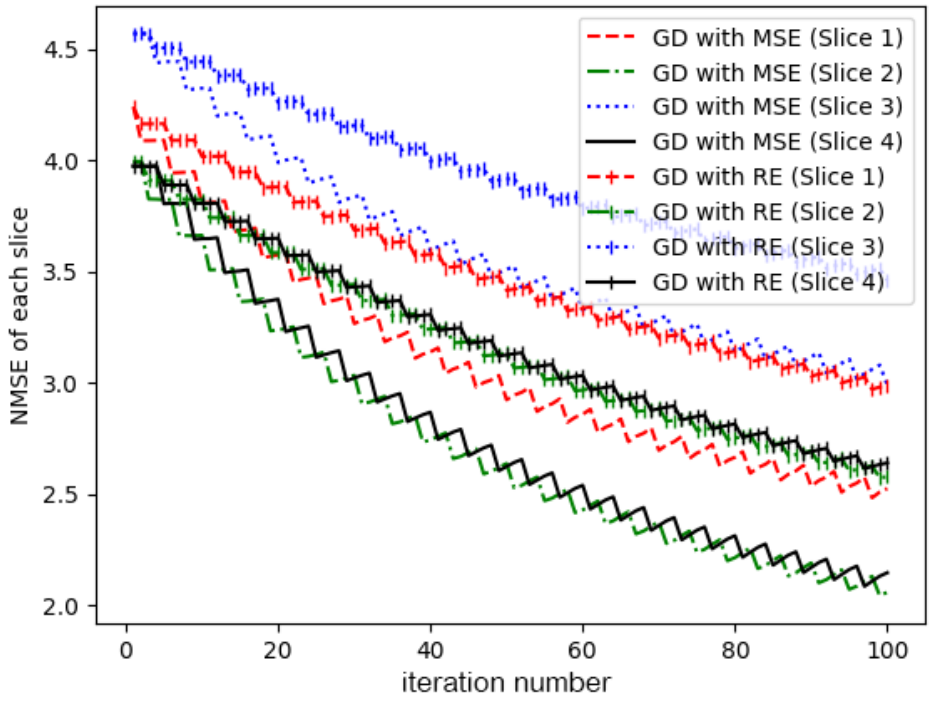
functions. And to ensure a sensible comparison, although three different loss functions are used for optimisation of CGH, the metric used to assess the final quality of multi-depth reconstructions of the hologram is the normalized mean squared error (NMSE). As there are a total of 4 slices in this example, the final NMSE of each are computed separately and the total optimisation run time is recorded. The final results are gathered in Fig. 4.10. As each slice has a different final error, their mean and standard deviation (SD) are also computed for investigations. Three columns are colour coded where green indicates better a result while red indicates a worse result.

		Loss	Final NMSE (x10^-6)						Time (s)
			Slice 1	Slice 2	Slice 3	Slice 4	Mean	SD	
L-BFGS	SoL	MSE	1.62	1.23	1.43	1.19	1.37	0.17	1.04
		CE	1.76	1.29	1.52	1.24	1.45	0.21	2.16
		RE	1.62	1.23	1.43	1.19	1.37	0.17	1.02
	SoH	MSE	3.09	2.87	3.31	2.91	3.05	0.17	1.76
		CE	3.05	2.83	3.39	2.87	3.03	0.22	2.25
		RE	3.09	2.87	3.31	2.91	3.05	0.17	1.81
	SS	MSE	2.57	2.52	2.50	2.48	2.52	0.03	0.50
		CE	2.76	2.63	2.63	2.64	2.67	0.05	0.74
		RE	2.57	2.52	2.50	2.48	2.52	0.03	0.47
GD	SoL	MSE	1.94	1.65	1.96	1.66	1.80	0.14	0.86
		CE	2.20	1.94	2.30	1.97	2.10	0.15	1.92
		RE	2.23	1.98	2.35	2.01	2.14	0.15	0.85
	SoH	MSE	3.33	3.06	3.67	3.09	3.29	0.25	0.63
		CE	3.58	3.29	3.96	3.30	3.53	0.27	1.14
		RE	3.59	3.30	3.97	3.31	3.54	0.27	0.59
	SS	MSE	2.53	2.06	2.99	2.15	2.43	0.37	0.28
		CE	2.99	2.57	3.46	2.64	2.92	0.35	0.54
		RE	2.99	2.58	3.46	2.64	2.92	0.35	0.28

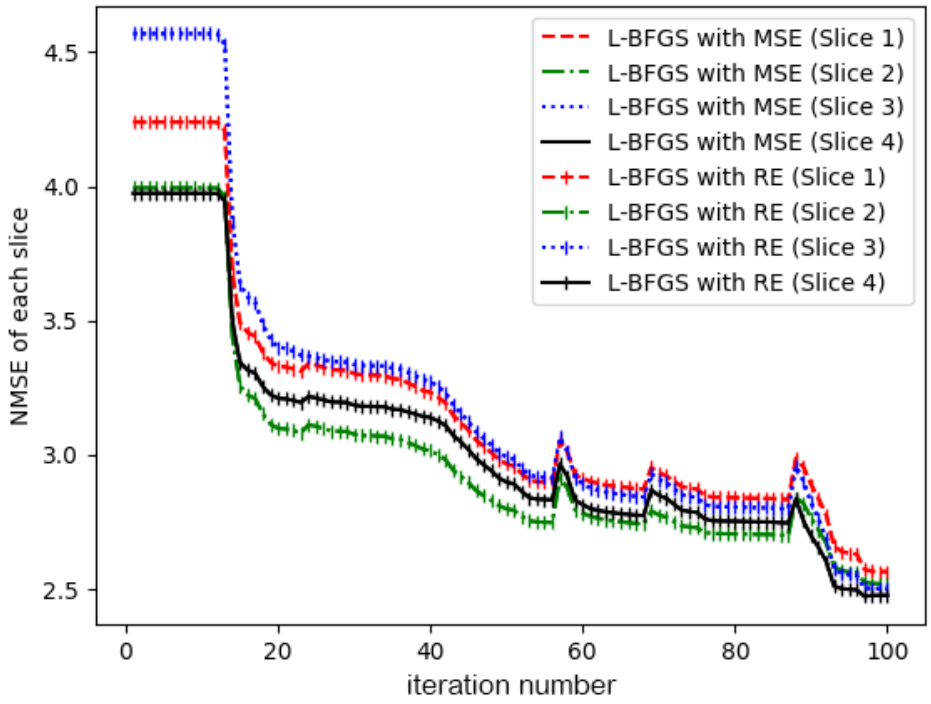
Fig. 4.10 Final NMSE and run time comparison across the three techniques

Comparing the mean of final NMSE and the run time of the proposed sequential slicing (SS) technique to those of the sum-of-loss (SoL) and sum-of-hologram (SoH) techniques in Fig. 4.10, it can be concluded that, for all combinations of optimisers and loss functions attempted, the proposed SS technique runs much quicker than the existing SoL and SoH techniques, while still providing a proper result, sitting between the SoL and SoH techniques. The SS technique is both quicker and has better reconstruction quality than the SoH technique, demonstrating an absolute advantage. Meanwhile, the runs with CE as loss function are

much slower and has not demonstrated any advantage in the final NMSE, demonstrating an absolute disadvantage, so the results of runs using CE loss or SoH method will not be shown in the per-iteration plots later on in this section.

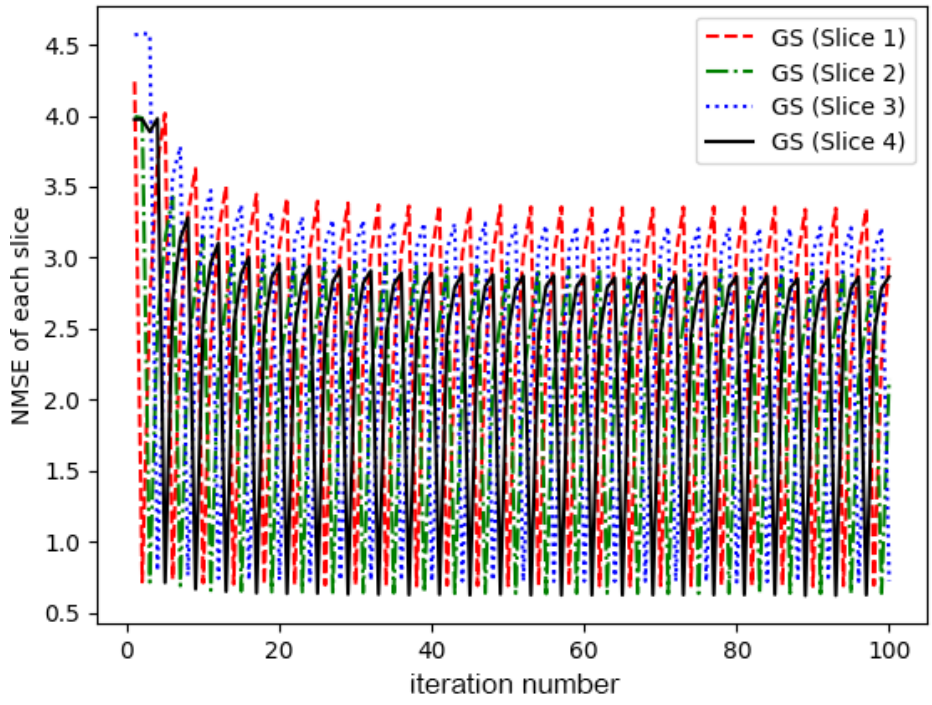


(a)

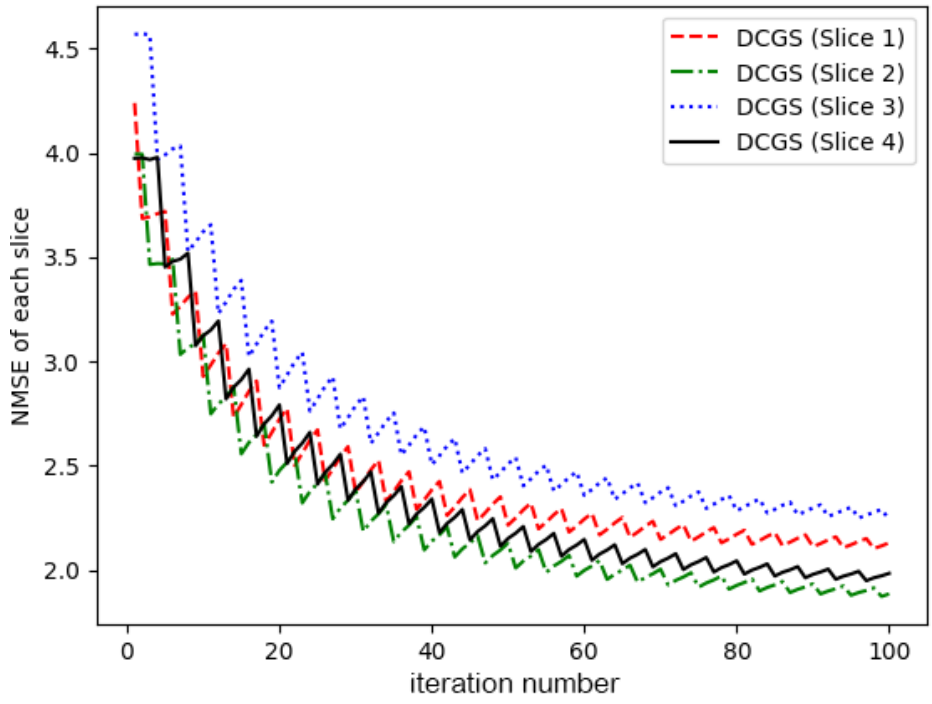


(b)

Fig. 4.11 NMSE of each slice plotted against the iteration number for the optimisation based SS technique implemented with MSE and RE loss functions and (a) GD optimiser, (b) L-BFGS optimiser.



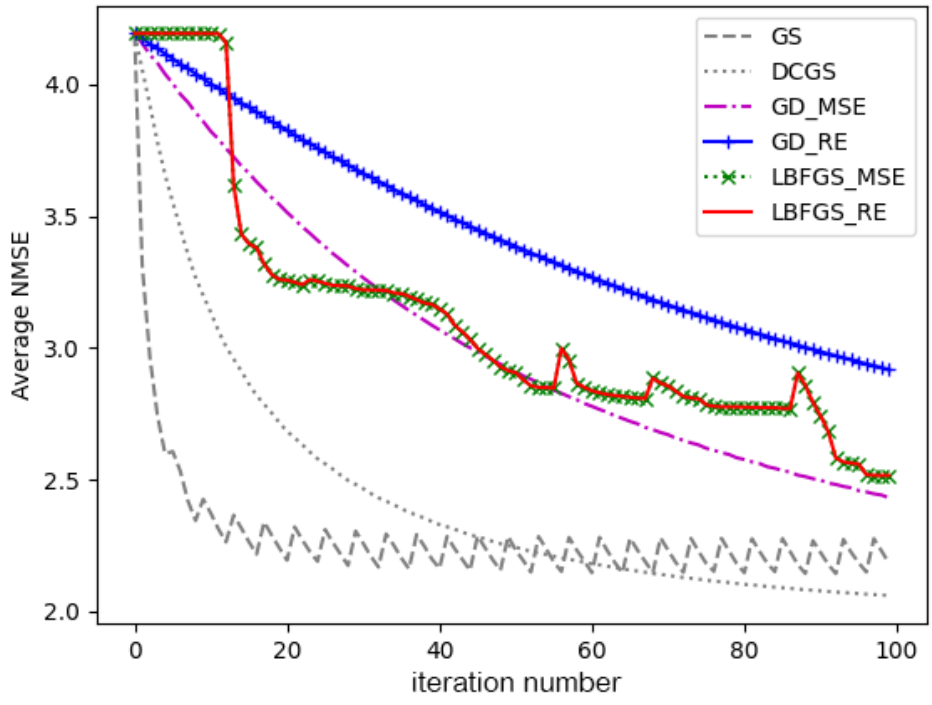
(a)



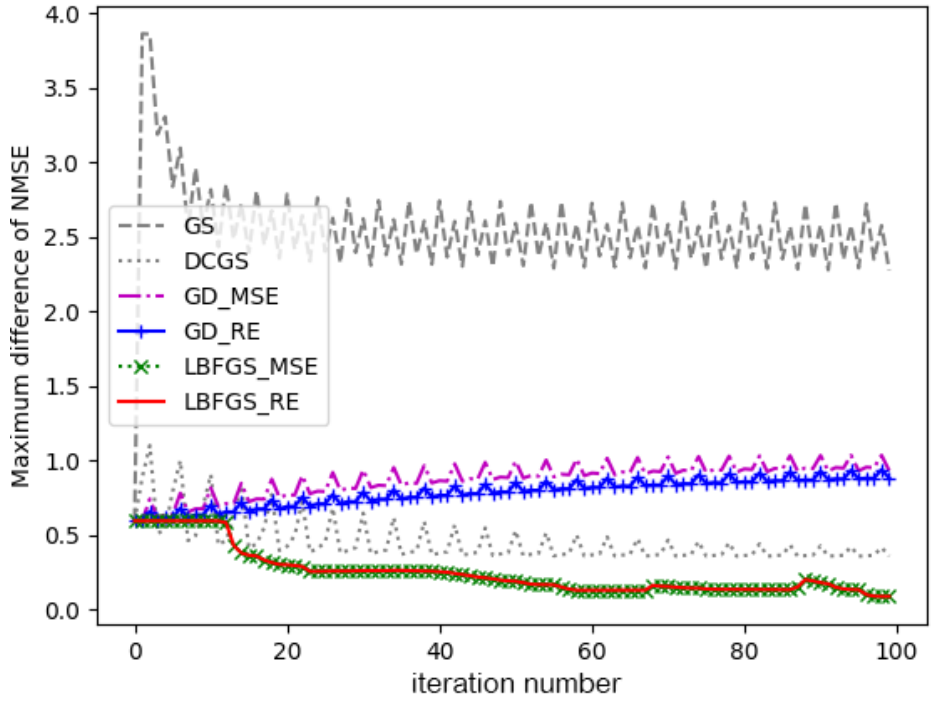
(b)

Fig. 4.12 NMSE of each slice plotted against the iteration number for the (a) GS with SS algorithm, (b) DCGS algorithm.

For comparison among the runs using the sequential slicing (SS) techniques, the NMSE for each slice are plotted for the GD and LBFGS optimisers with MSE and RE loss functions in Fig. 4.11, and the GS-based sequential GS and DCGS [62] methods (mentioned in Section 2.3.7) are also plotted in Fig. 4.12 for reference. Looking at the plots of NMSE of each slice against iteration numbers (in Fig. 4.11 and Fig. 4.12), apart from the L-BFGS algorithm (Fig. 4.11b), all the other algorithms (in Fig. 4.11a Fig. 4.12a Fig. 4.12b) are showing a staircase-like trend, where a decrease in error on one slice results in an increase in error on all the other slices, and the final NMSE of each slice distinguishes a lot from another. The sequential GS algorithm suffers the most from the quality imbalance between each slice, and the sequential GD algorithm follows. The DCGS algorithm benefits from its modification of the inclusion of a weighting factor consisting historical amplitude, therefore managed to converge. The average and maximum difference of the NMSE values across the 4 slices are then computed and summarised in Fig. 4.13.



(a)



(b)

Fig. 4.13 (a) Average NMSE among all slices, (b) Maximum difference of NMSE across all slices, plotted against the iteration number for the 6 sequential slicing runs using different techniques

Fig. 4.13 plots the average and maximum difference values of NMSE across the 4 slices against the iteration number, for all the six runs (sequential GS, DCGS, GD optimiser with MSE as loss function, GD optimiser with RE as loss function, L-BFGS optimiser with MSE as loss function, and L-BFGS optimiser with RE as loss function). From the average NMSE plot (Fig. 4.13a), the proposed sequential L-BFGS method does not have the lowest average NMSE, but it has the lowest quality imbalance across the slices as shown in the maximum difference plot (Fig. 4.13b). The L-BFGS algorithm mainly benefits from its inclusion of curvature information during optimisation, so that the update of hologram $\mathbf{H}$ at each iteration takes into account not only the loss for the current slice, but also all the historical iterations up to the set history size (m in Algorithm 8). So for the L-BFGS algorithm, at each iteration, the NMSE of all slices behave in the same way, ensuring each slice to have the similar quality (i.e. lower difference of NMSE across slices). Hence, the proposed SS technique with L-BFGS optimiser is shown to have the lowest quality imbalance across all slices, although the average quality across the 4 slices is worth than the GS based algorithms.

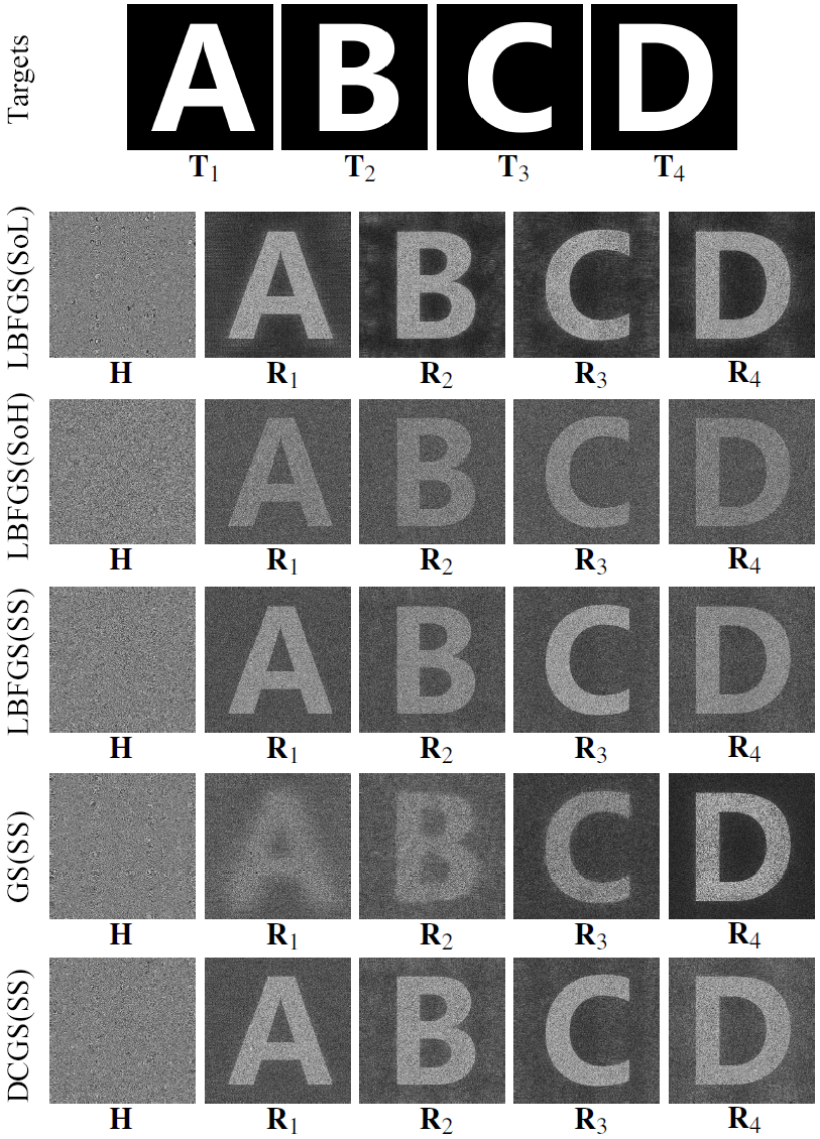


Fig. 4.14 Comparison of final holograms and reconstructions on the 4-slice target consisted of letters ‘A’, ‘B’, ‘C’ and ‘D’

The final holograms and their reconstructions for L-BFGS algorithm with SoL, SoH and SS techniques are shown in Fig. 4.14, with GS with SS technique and DCGS also shown as reference. The reconstructed images confirm the SS technique having a quality between SoH and SoL method (for the same amount of iterations), and has a much better quality imbalance than sequential GS, which has a very clear reconstruction at the fourth slice (letter ‘D’) because the iteration stopped at the fourth slice but much worse reconstruction at other slices. Admittedly the proposed L-BFGS with SS method cannot surpass the GS-based DCGS algorithm yet, but among the optimisation-based methods, the proposed L-BFGS with SS

combination has shown better reconstruction quality than the SoH method inFig. 4.14, and takes fewer time to run than the SoH and the SoL methods as previously shown in Fig. 4.10.

CGH for a 4-slice target consisted of letters and real-life images

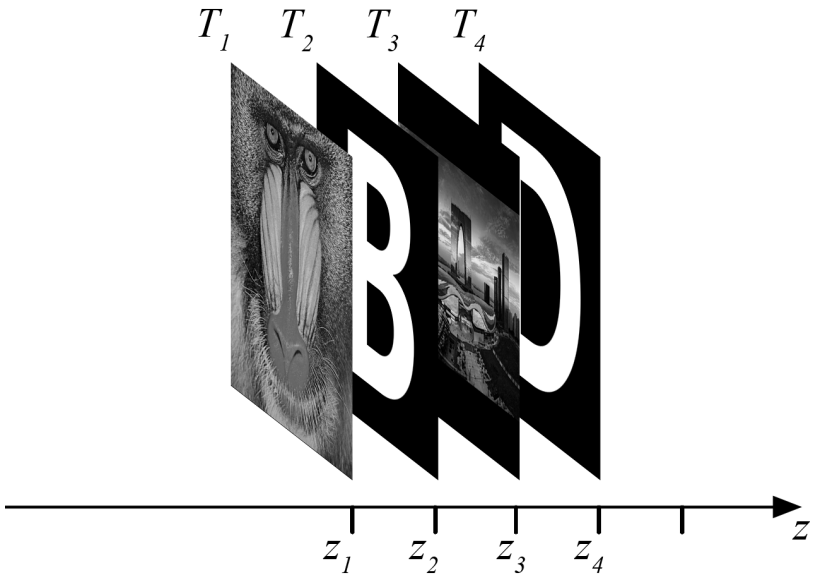


Fig. 4.15 Layout of the non-binary 4-slice target

To prove that the proposed method also works for non-binary valued target images, another example of a 4-slice 3D target is attempted, as shown in Fig. 4.15, where two of the slices are replaced by an image of the mandrill [18] and an image of the city scene [31] respectively.

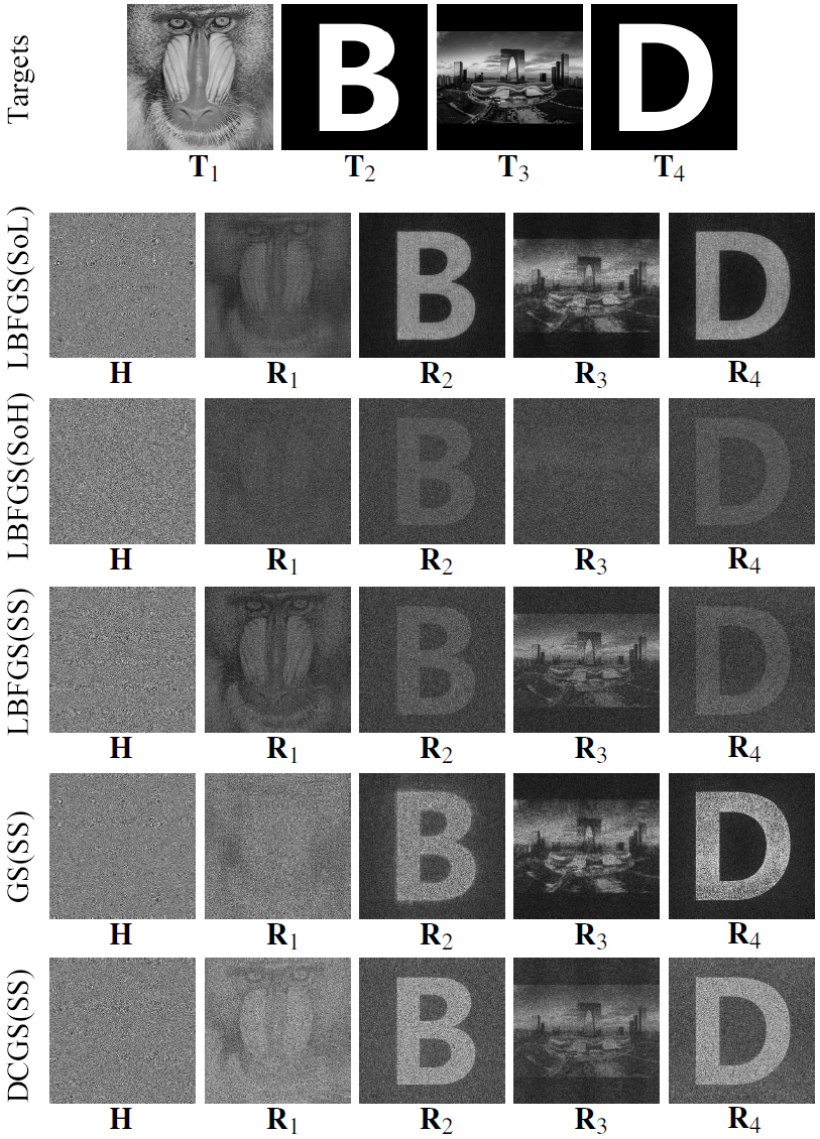


Fig. 4.16 Comparison of final holograms and reconstructions for non-binary target

As shown in the final holograms and reconstructions in Fig. 4.16, the proposed LBFGS with SS technique still managed to converge, with final reconstruction quality sitting between the SoL and the SoH method, and also having a good quality balance across all slices. As the SS technique is faster than both the SoL and the SoH method, its quality performance is impressive.

4.5 Summary

This chapter started from the background knowledge of numerical optimisation including the L-BFGS optimiser, then introduced and carried out the optimisation of phase hologram. Then the novel TIPO algorithm was proposed, which optimises for the phase of the target image instead of the phase of the hologram. The TIPO showed its effectiveness on CGH but did not show any advantage compared to the regular phase hologram optimisation methods. And lastly, the method of using L-BFGS optimiser with SS technique was proposed for multi-depth CGH. The L-BFGS with SS technique has demonstrated a good suppression on the quality imbalance across the multi-depth slices, benefiting from the nature of L-BFGS being a second order optimiser, which implicitly records the historical gradients by other slices for the determination of the descent direction. For both GD and L-BFGS optimisation algorithms, the SS technique runs faster and produces better reconstruction quality than the simple SoH technique, and it is much quicker than the SoL technique. Therefore, the proposed L-BFGS optimisation with SS method has demonstrated great ability of time-limited optimisation of multi-depth 3D CGH.

However, all the CGH methods investigated in this chapter produce single-frame multi-level phase hologram, which cannot be used directly on the binary-phase SLM in the optical setup described in Section 3.2. So the next chapter will propose a multi-frame binary-phase holograms batched optimisation (MFHBO) method, to optimise multi-frame binary-phase holograms which are the most suitable for the optical setup in Section 3.2.

Chapter 5

Multi-Frame Binary-Phase Holograms Batched Optimisation

Note: The work in this chapter has been published in Ref. [9]

This chapter builds up on the idea of time-averaging multiple hologram frames, and proposes a technique called Multi-Frame Holograms Batched Optimisation (MFHBO), which uses the L-BFGS optimisation algorithm to simultaneously generate a batch of phase-only holograms which result in an average reconstructed image of improved fidelity and fast algorithmic convergence, both in the Fraunhofer and the Fresnel regions.

5.1 Introduction

Chapter 4 implemented optimisation algorithms to generate single-frame holograms. This chapter proposes the novel MFHBO method which optimise a batch of multi-frame holograms for time multiplexing. Time multiplexing seeks to improve a time-averaged response by displaying different hologram sub-frames at a high refresh rate [79]. Such approach can exploit the finite response time of human vision, where human eyes average out the unwanted noise while the wanted signal remains. A few time-multiplexed multi-frame holograms generation methods have been explored in the literature, including the OSPR algorithm in Section 2.3.5 and the AD-OSPR algorithm in Section 2.3.6; however, both OSPR and

AD-OSPR are still subject to noise and defects in reconstruction quality. The objective of the proposed MFHBO algorithm is therefore to produce a batch of holograms having better reconstruction quality than the existing OSPR and AD-OSPR methods.

5.2 Methods

The use of the L-BFGS algorithm for single-frame hologram optimisation has been introduced in Chapter 4. To implement it onto multi-frame holograms generation, the argument to optimise becomes the set of holograms with n sub-frames ($\{\mathbf{H}_1, \mathbf{H}_2, ..., \mathbf{H}_n\}$), each having a resolution of $X \times Y$ pixels matching the resolution of the target image, and the objective function to minimise is therefore the difference between the average reconstruction amplitude ($\mathbf{R}_{avg} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{R}_i$) and the target image ($\mathbf{T}$), which is denoted as $Loss(\mathbf{T}, \mathbf{R}_{avg})$, where n is the total number of frames, $\mathbf{R}_i$'s are reconstructions from individual hologram sub-frames $\mathbf{H}_i$'s for $i \in [1, n]$. To compute each $\mathbf{R}_i$ from the corresponding $\mathbf{H}_i$, the Fresnel diffraction formula given in Eq. (2.29) is used. As we are generating holograms for phase-only SLM's, the hologram aperture $\mathbf{A}$ in Eq. (2.29) is then comprised of a uniform amplitude with phase $\mathbf{H}$, giving $\mathbf{A} = e^{j\mathbf{H}}$, where the exponential is taken element-wise, and the replay field($\mathbf{E}$)'s amplitude is the reconstruction (i.e.$\mathbf{R} = |\mathbf{E}|$).

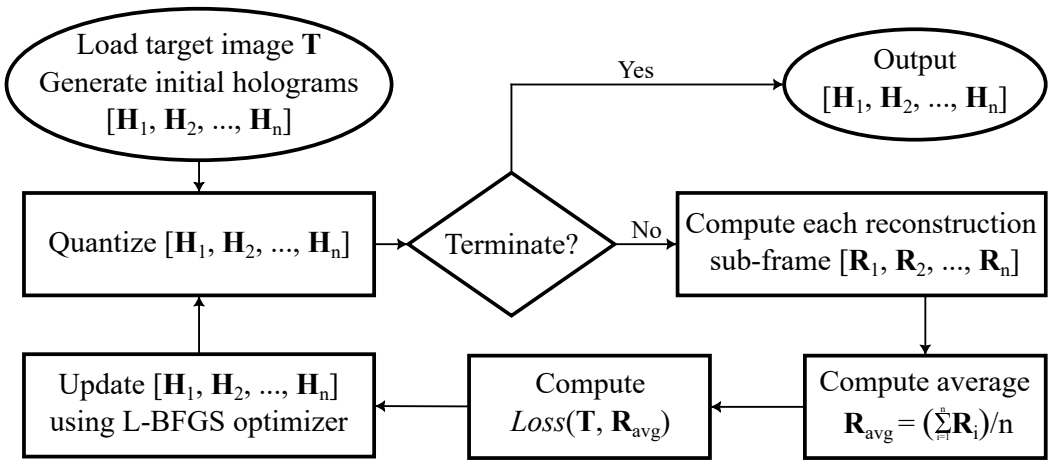


Fig. 5.1 MFHBO flowchart

To help explain the optimisation process, a flow chart is drawn in Fig. 5.1. As shown in the flowchart, the target image $\mathbf{T}$ is first loaded, with a set of n hologram sub-frames ($\{\mathbf{H}_1, \mathbf{H}_2, ..., \mathbf{H}_n\}$) generated randomly. Then at every iteration, each hologram sub-frame $\mathbf{H}_i$ is quantised to the bit-depth constraint of the SLM, and propagated to the reconstruction

plane $\mathbf{R}_i$, and the average of the amplitudes of all reconstructions $\mathbf{R}_{avg}$ is computed and compared against the target image $\mathbf{T}$ using a loss function $Loss(\mathbf{T}, \mathbf{R}_{avg})$, after which the search direction is computed using the L-BFGS optimiser and the hologram sub-frames are updated accordingly. Here the loss function selected is the relative entropy[77] given in Eq. (4.18).

Since fast SLM's available in the lab are binary-phase devices, the quantisation step in the flowchart in Fig. 5.1 is carried out with bit-depth limit of 1, hence producing binary-phase holograms. However, the optimisation algorithm does not converge with a straight binary quantisation as integers are discrete, therefore a Sigmoid function [80] is used for a smoother and differentiable quantisation, as defined in Eq. (5.1). The output of the Sigmoid function is then scaled by π so that the binary phase levels are 0 and π.

$$\text{sigmoid}(x) = \frac{1}{1+e^{-x}} \tag{5.1}$$

And finally, when displaying the multi-frame holograms, each of the n frames generated are then rounded to binary phase values and displayed on the binary phase SLM sequentially. And when the first round finished, the second round starts with the first frame again (i.e. after frame n, the next frame displayed is frame 1), and such infinite loop doesn't stop until another set of holograms are uploaded.

5.3 Results

5.3.1 Simulation results

To test the proposed MFHBO method, a target image $\mathbf{T}$ as shown in Fig. 5.2 was used. It was designed from the widely used mandrill image [18] in Fig. 2.11. A rotational symmetry was introduced to match the rotational symmetry of the far field projection from binary phase holograms, as explained in Section 2.2.3. It was then zero padded to a resolution of $1024px \times 1024px$ and subsequently interpolated (using the '$torch.nn.functional.interpolate$' function in the PyTorch module[81]) to a resolution of $1280px \times 1024px$ to match the resolution of the SLM in our lab. Note that the target image was zero padded to a square aspect ratio and then stretched to the non-square aspect ratio because more pixels in the horizontal axis only means higher sampling rate as part of the features of the FFT, the replay field is continuous

and is not pixelated and the simulated reconstruction of $1280px \times 1024px$ resolution is the sampled result, which will be illustrated visually in Fig. 5.4 later.

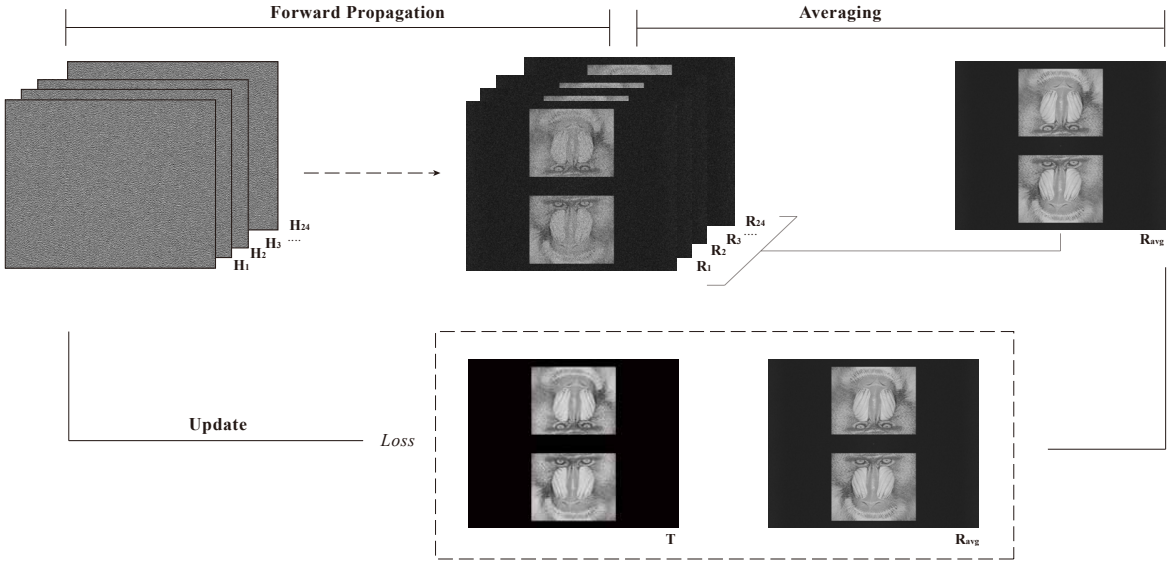
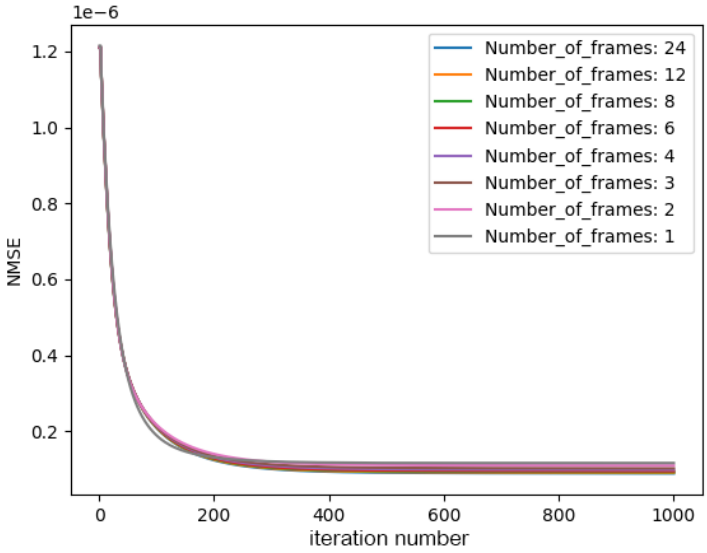
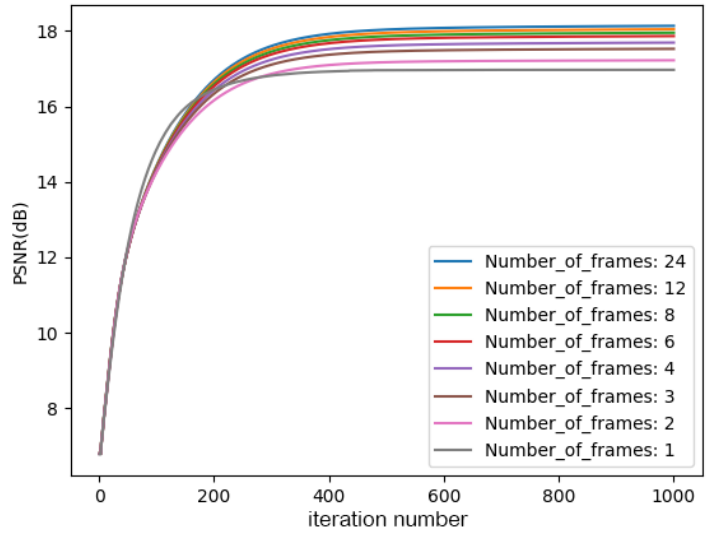


Fig. 5.2 An example iteration in the optimisation process

To further explain the optimisation process described in Fig. 5.1, an example iteration with $n = 24$ is shown in Fig. 5.2. At each iteration, every hologram is quantised and propagated to the reconstruction plane, forming $\{\mathbf{R}_1, \mathbf{R}_2, ..., \mathbf{R}_{24}\}$. The average reconstruction amplitude $\mathbf{R}_{avg}$ is then compared against the target image $\mathbf{T}$, using the loss function in Eq. (4.18). The holograms $\{\mathbf{H}_1, \mathbf{H}_2, ..., \mathbf{H}_{24}\}$ are then updated according to the search direction calculated using the L-BFGS optimiser. After setting the optimisation to terminate when the number of iterations reach 1000, the same algorithm was run on the same target for different number of frames (n), the normalised mean squared error (NMSE) and the peak signal-to-noise ratio (PSNR) between the average reconstruction $\mathbf{R}_{avg}$ and the target image $\mathbf{T}$ were calculated at every iteration and plotted in Fig. 5.3a and Fig. 5.3b respectively.



(a) NMSE v.s. Iteration Number



(b) PSNR v.s. Iteration Number

Fig. 5.3 Convergence of the MFHBO algorithm on the rotationally symmetrical Mandrill target

The plots in Fig. 5.3 show that the proposed MFHBO method has achieved good convergence within 400 iterations, for the various number of frame settings n in $\{1,2,3,4,6,8,12,24\}$. The final NMSE values in Fig. 5.3a are difficult to distinguish in the plot, therefore it will be further compared in the bar chart in Fig. 5.4. The number of frames are chosen to be integer

factors of 24, which is determined by our experimental setup, further explained in the next subsection.

n	200 iterations	400 iterations	600 iterations	800 iterations	1000 iterations
1	1.79	3.59	5.31	7.00	8.72
2	2.82	5.59	8.34	11.11	13.88
3	3.84	7.67	11.45	15.21	19.00
4	4.93	9.83	14.70	19.58	24.47
6	6.95	13.87	20.76	27.58	34.50
8	8.87	17.67	26.54	35.60	44.56
12	12.95	25.79	38.63	51.47	64.30
24	51.81	101.43	151.09	201.08	251.15

Table 5.1 MFHBO runtime (s)

The programme runtime of the proposed MFHBO method has been measured on a laptop computer of model ASUS ROG Zephyrus M16 (GU603H) with a CPU of model i7-11800H and a GPU of model NVIDIA RTX3060 and the results for different combinations of number of frames and number of iterations are listed in Table 5.1. It can be concluded that the application of the proposed method is for pre-computed high-quality holograms, instead of real-time holographic projection.

5.3.2 Optical Experiment results

The holographic projection system used in this experiment is the same as the one described in Section 3.2, with the optical setup shown in Fig. 3.2. Since the SLM has a refresh rate of 1440Hz and modern computer monitors have refresh rate of at least 60Hz, the maximum number of frames was chosen to be $1440/60 = 24$, so that each set of 24 frames will take a total of $1/60$ seconds to display, therefore giving an equivalent refresh rate of 60Hz. Then the integer factors of 24 were chosen so that the equivalent refresh rate becomes integer multiples of 60Hz. The number of frames starts from 1 to help illustrate how the increase in number of frames positively affect the reconstruction quality.

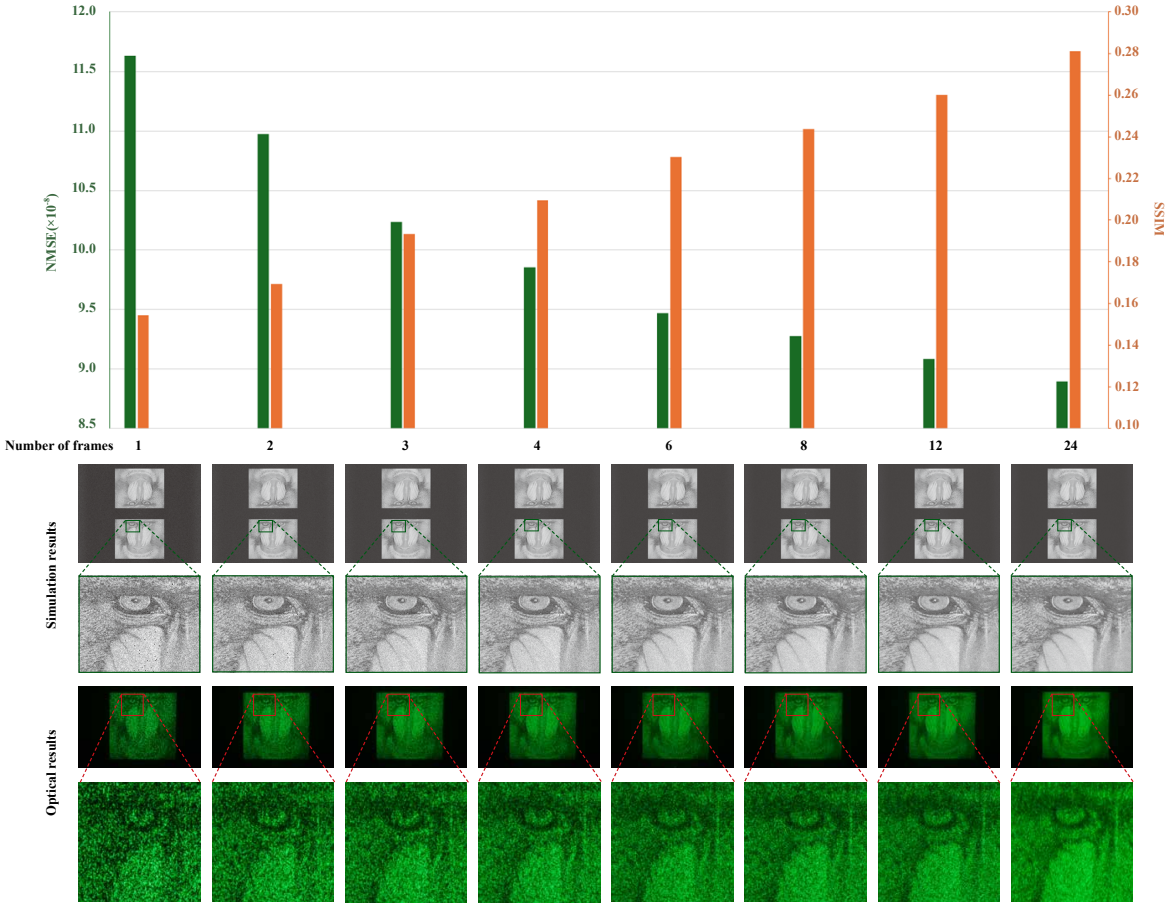


Fig. 5.4 Simulation and optical reconstruction results for different number of frames

The results in Fig. 5.4 further compares the final results for different number of frames. The histogram in Fig. 5.4 shows that, as the number of frames increases, the NMSE between the average reconstructions $\mathbf{R}_{avg}$ and the target image $\mathbf{T}$ decreases and the structural similarity index (SSIM)[58] increases, showing a trend of better reconstruction quality with higher number of frames. Such trend is expected as more frames provide higher information capacity, which agrees with the information capacity research in Chapter 6 where holograms with higher bit depth were found to achieve better reconstruction quality. The trend is also shown visually via the simulation results and their detail enlargements. The corresponding multi-frame holograms are then loaded onto the SLM, and the reconstructed field is captured using a camera of model Canon EOS 1000D. Only the bottom halves of the reconstructed field were captured as the symmetrical conjugates were unwanted feature of far field projections from binary-phase SLM's. The raw data including multi-frame binary-phase holograms, simulated reconstruction and optical results captured are accessible in the database [82].

Fig. 5.5 Sample target image - ‘holography’ ambigram

Then another target image was tested, which is the holography ambigram shown as shown in Fig. 5.5.[1] The term ambigram is used to refer to (often typographical) designs that are invariant under a reflection, rotation or other symmetry. The ‘holography’ design contains 180-degree rotational symmetry, which makes it especially well suited to binary Fourier-holographic projection, where this symmetry is unavoidable. Multi-frame holograms were then generated using the proposed MFHBO method and the existing OSPR and AD-OSPR methods, for the same number of frames $n = 24$. And the optical results are shown in Fig. 5.6.

[1] Adapted, with colours reversed, from *holography* - Benjamin Wetherfield, 2022.

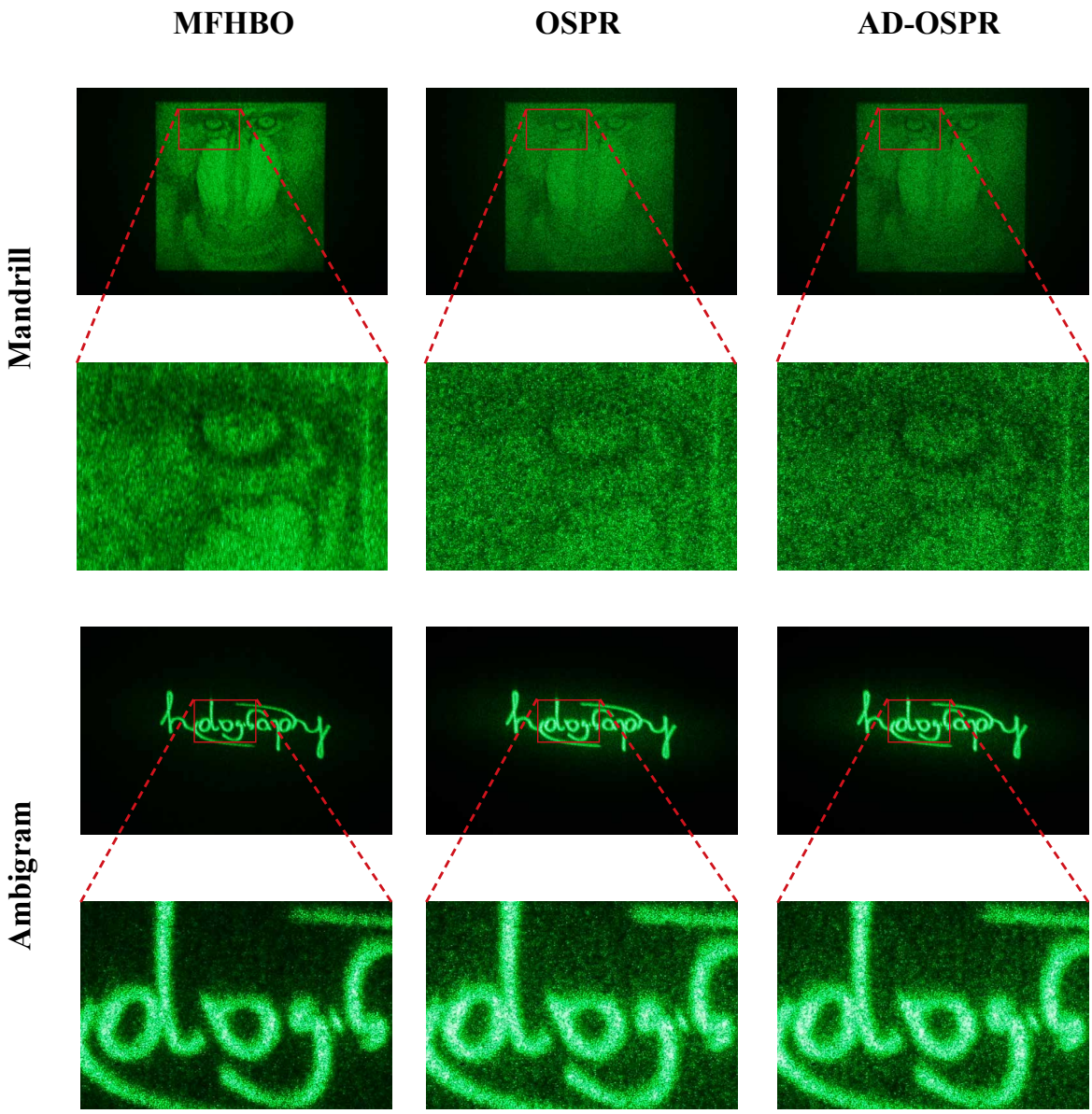


Fig. 5.6 Optical results comparison of the proposed MFHBO method against the existing OSPR and AD-OSPR methods

As shown in Fig. 5.6, for the Mandrill target image, it can be seen that the proposed MFHBO method achieved a much better optical reconstruction quality than the existing OSPR and AD-OSPR methods, with clearer details and better contrasts; for the 'holography' ambigram target image, the proposed MFHBO method is shown to have a much lower background noise around the centre, than the existing OSPR and AD-OSPR methods. The intended black regions are represented much more cleanly, with an elimination of speckle-like artefacts in the zero-valued space around the lettering, and an overall increase in discernible contrast.

Image	Metric	MFHBO	OSPR	AD-OSPR
Mandrill	NMSE ($\times 10^{-4}$)	0.84	1.00	1.12
	SSIM	0.124	0.076	0.078
Ambigram	NMSE ($\times 10^{-5}$)	2.29	3.31	3.23
	SSIM	0.795	0.826	0.827

Table 5.2 Quantitative analysis of the optical results in Fig. 5.6

A quantitative analysis was then conducted on the optical results in Fig. 5.6, the NMSE and SSIM between the captured reconstructions and there corresponding targets are computed and listed in Table 5.2. The NMSE results of the proposed MFHBO method are lower than those of the existing OSPR and AD-OSPR methods, with a 25% reduction on average among both target images. On the other hand, the SSIM results have shown a 62% increase using MFHBO than OSPR and AD-OSPR for the mandrill target image, but a slight decrease of 3.7% for the 'holography' ambigram target image, which is negligible as it is less than 5% and the SSIM metric is not originally designed for binary-valued non-greyscale images.

3D Holography

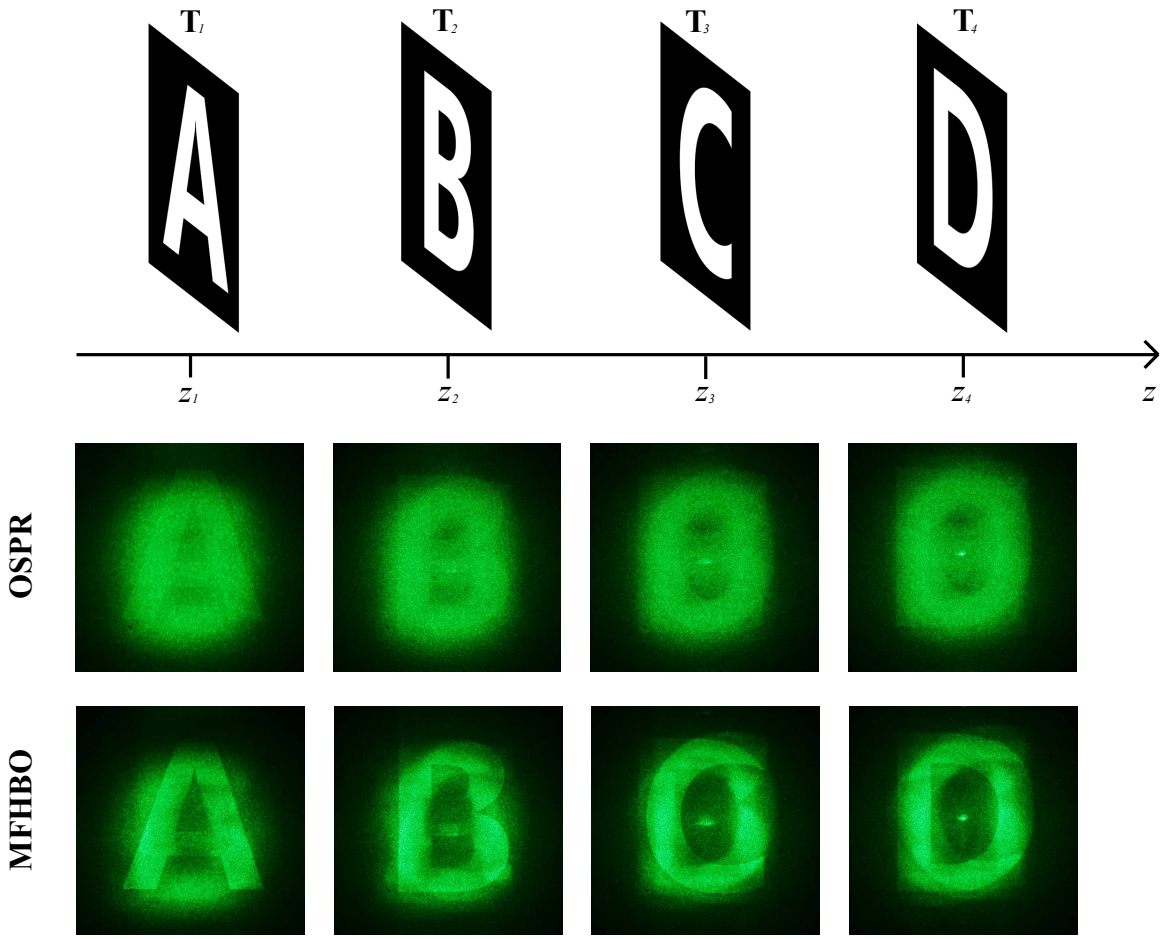


Fig. 5.7 4-slice target and according reconstruction results

The proposed MFHBO method was extended to multi-slice targets, by computing the loss between all 4 slices of reconstructions and target images (the Sum-of-Loss method in [4]). An example 4-slice target made from letters 'A, B, C, D' is shown in Fig. 5.7. The z values, corresponded to the z variable in Eq. (2.29), were chosen to be $1.1m, 1.9m, 3.5m, 7.7m$ for the 4 slices respectively (as there's no correlation between each slice, larger separation was chosen for fewer cross-talk across different planes). It can be seen that the proposed MFHBO method has produced sharper edges in reconstruction than the existing OSPR method. (The AD-OSPR method was not attempted here as its application to multi-slice targets was not defined).

Method	Metric	Slice 1	Slice 2	Slice 3	Slice 4	Average
OSPR	NMSE($\times 10^{-4}$)	4.980	4.484	5.644	4.846	4.988
	SSIM	0.072	0.061	0.048	0.060	0.060
MFHO	NMSE($\times 10^{-4}$)	4.484	4.230	4.990	4.289	4.498
	SSIM	0.063	0.067	0.058	0.072	0.065

Table 5.3 Quantitative analysis of the optical results in Fig. 5.7

Then a quantitative analysis was carried out, with NMSE and SSIM values measured and shown in Table 5.3. The proposed MFHBO method has shown a 10% reduction in NMSE and a 8% improvement in SSIM on average than the existing OSPR method, demonstrating the effectiveness of the proposed method.

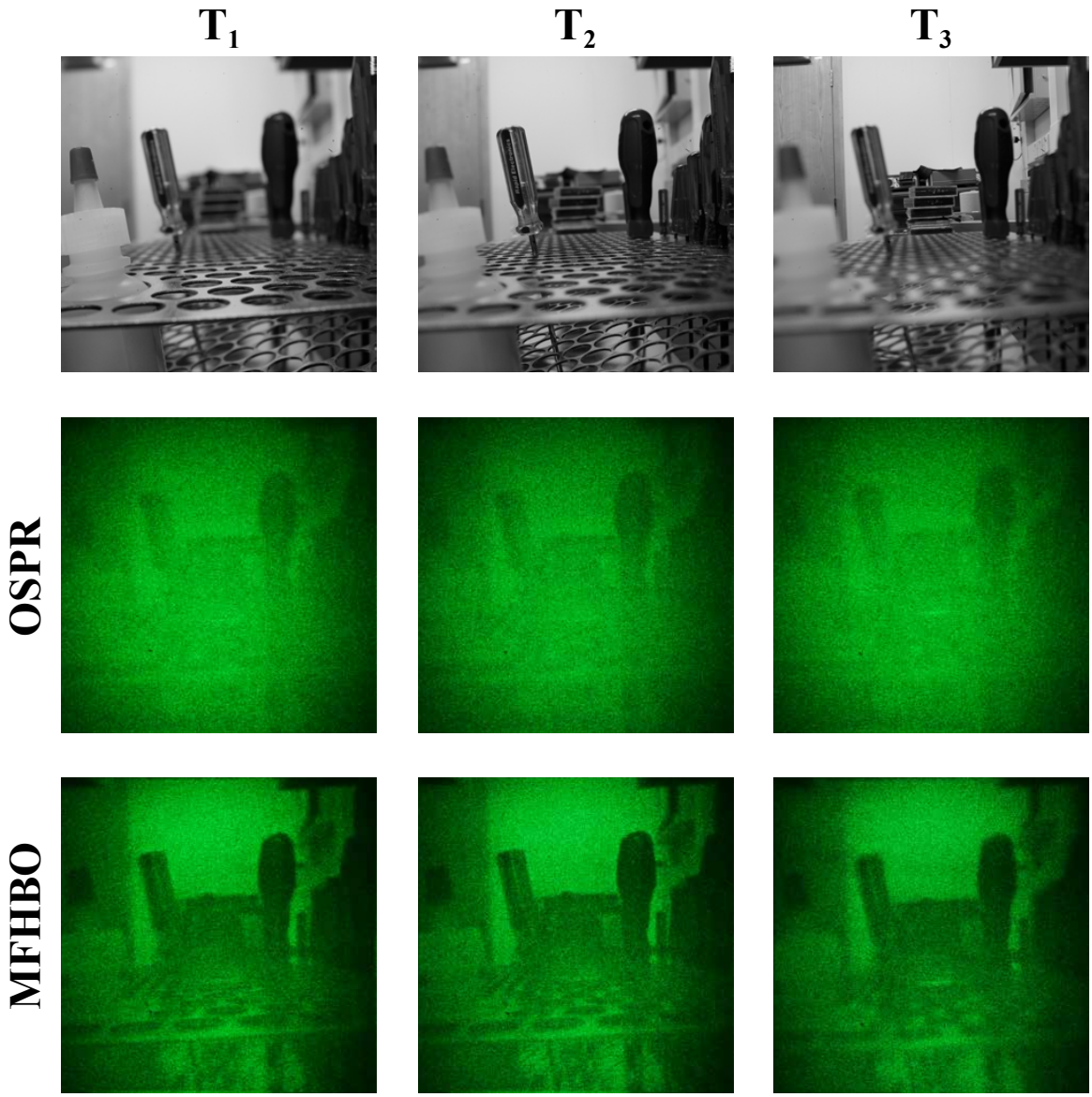


Fig. 5.8 Real-life captured image as target field and their reconstruction results

Lastly, a set of real-life scene was captured in the lab using near, middle and far focus, as shown in $\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3$ in Fig. 5.8 respectively. The z values were set to $1.1m, 1.2m, 1.3m$ for hologram generation, and the reconstruction results of the existing OSPR and the proposed MFHBO methods are compared in Fig. 5.8. The proposed MFHBO method is shown to have achieved better reconstruction quality than the existing OSPR method.

Method	Metric	Slice 1	Slice 2	Slice 3	Average
OSPR	NMSE($\times 10^{-6}$)	3.70	3.69	3.47	3.62
	SSIM	0.37	0.28	0.34	0.33
MFHO	NMSE($\times 10^{-6}$)	3.20	2.78	3.06	3.01
	SSIM	0.42	0.32	0.32	0.35

Table 5.4 Quantitative analysis of the optical results in Fig. 5.8

A quantitative analysis was conducted again, with NMSE and SSIM values measured and listed in Table 5.4. The proposed MFHBO method has shown a 17% reduction in NMSE and a 7% improvement in SSIM on average than the existing OSPR method, proving the effectiveness of the proposed method.

5.4 Summary

This chapter proposed the MFHBO method to generate multi-frame binary-phase holograms to be displayed on a high refresh rate binary-phase SLM. The proposed MFHBO method was shown to achieve much better reconstruction quality and higher contrast than the existing multi-frame binary-phase hologram generation methods OSPR [46] and AD-OSPR [64] on the holographic projector with a binary-phase SLM, for all the single-slice far-field targets and the multi-slice near-field targets tested. Although the proposed MFHBO method is slower than the existing OSPR and AD-OSPR methods, its much better reconstruction quality makes it suitable for pre-computed high-quality hologram applications. Its strong advantage for high contrast target such as the 'holography' ambigram in Fig. 5.5, with much suppressed noise in the background, makes it well-suited for photo-lithography applications. The proposed method can also be adapted for multi-level SLM's by simply removing the quantisation step in Fig. 5.1. This could be the case for applications such as photo-lithography, where the time response of the system is much longer than it is for human vision, and the high refresh rates of the SLM are not necessary.

Chapter 6

Information Capacity of Phase-Only Computer-Generated Holograms

Note: The work in this chapter has been published in Ref. [8]

Despite many years of development in computer-generated holography, perfect phase-only holograms for most target images are still yet possible to compute. All computational phase retrieval algorithms end up with some error between the target image and the reconstruction of the computer-generated hologram (CGH), except for specific targets. This chapter focuses on the fundamental limits of phase-only CGH quantised to limited bit-depth levels, from the information theory point of view, revealing the information capacity of CGH and its effect on reconstruction quality, with an attempt to quantify how hard a target image is for phase-only CGH computation.

6.1 Introduction

As introduced in the previous chapters, Holography is a technology that can fully reconstruct the wavefront of 3D objects. CGH is a technique for converting a 3D object scene into a 2D complex-valued hologram [83], without the need for the 3D object to physically exist. However, despite many advancements in liquid crystals and micro mirrors technologies, complex-valued SLM's are still not available yet, and the currently available SLM's can

only achieve either amplitude or phase modulation, among which the phase modulation is usually preferred in holographic projections for its lower zero order and higher energy efficiency due to fewer blockage of light. There are many algorithms available to compute good quality phase-only holograms as shown in the previous chapters; however, none of them can guarantee to compute a perfect phase hologram for a 3D scene or even a 2D image, where they always end up with some error between the reconstruction of the hologram and the target scene, especially when the phase holograms are quantised to be able to display on SLM with limited bit depth. An intuition therefore arose that the bit depth of the phase hologram is limiting the reconstruction quality, and that the target scene's entropy seems to denote how difficult it is for phase hologram generation. It is obvious that the entropy of the target can certainly never exceed the bit depth limit of its corresponded perfect hologram, otherwise a lossless compressor breaking the Shannon's information theory [84] would be invented; however, such statement is not quite useful in practice as perfect holograms are generally impossible to find.

The literature review had found no related work on the information entropy of computer-generated holograms, with the closest match being the research on hologram compression using optical method to achieve lossy compression [85], which cannot be integrated in CGH processes. Therefore, this chapter aims to investigate how much information content can a bit-depth-limited phase hologram contain, taking from an information theory point of view. Previously, work had been done to investigate the effect of Bit-depth in Stochastic Gradient Descent (SGD) performance for phase-only computer-generated holography displays [3]. This chapter extends on previous research onto the Gerchberg-Saxton (GS) [30] algorithms for hologram generation, and investigates the correlation between the quality of the reconstructed image from the hologram and the information entropy of the target image, with an aim to reduce the hologram's entropy during the CGH process, for smaller size holograms.

6.2 Methods

6.2.1 Quantised CGH Algorithm

To compute phase-only holograms, the Gerchberg-Saxton (GS) [30] algorithm, which is the classic and robust phase retrieval algorithm introduced in Section 2.3.4, is selected. Taking the Fraunhofer propagation in Eq. (2.31) as an example, where the reconstructed field is simply the Fourier Transform (FFT when discretised) of the hologram field, the operation flowchart of GS algorithm is illustrated in Fig. 6.1. GS algorithm functions that it iteratively

determines the phase profile ($\angle A$) of the hologram (A) required to reconstruct a target image (T); it loops between the hologram (A) and the diffracted plane (E), and applying constraints to each plane accordingly during each iteration [30], with the total number of iterations denoted by N.

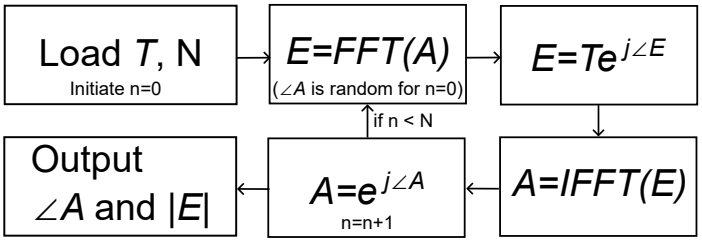
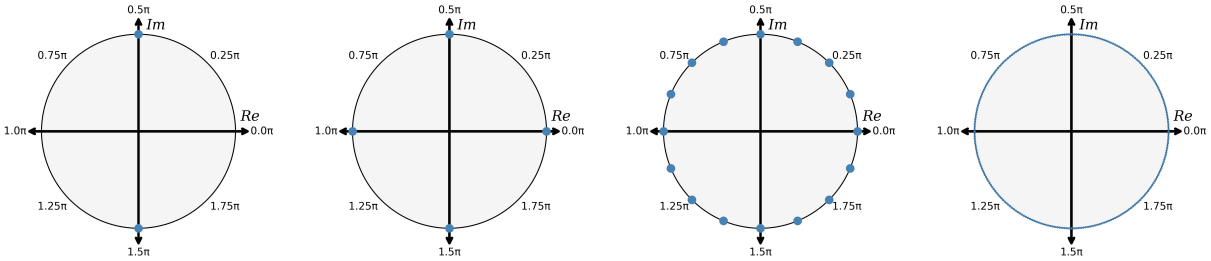


Fig. 6.1 Gerchberg-Saxton (GS) algorithm flowchart

Then, as illustrated in Fig. 6.2, a quantisation function (Q) can be defined by finding the closest point from one of the 2^d quantisation levels, given the phase ($\angle A$) and the quantisation bit depth d as input.



(a) bit depth = 1 (b) bit depth = 2 (c) bit depth = 4 (d) bit depth = 8

Fig. 6.2 Quantisation of phase holograms

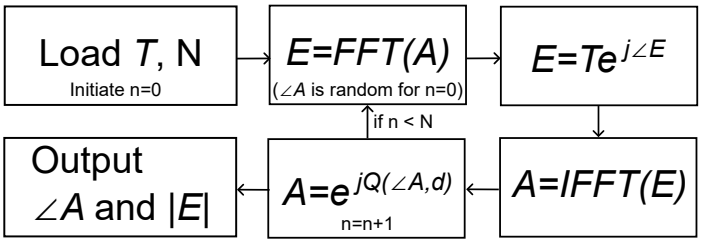


Fig. 6.3 Quantised Gerchberg-Saxton (GS) algorithm flowchart

To compute phase holograms quantised to certain bit depth (d), the GS algorithm is modified to include an additional quantisation operation (Q) when applying the 'phase-only' constraint

as shown in Fig. 6.3. Such method is better than applying the quantisation at the end of the loop, as it includes the quantisation constraint throughout the iterations, instead of introducing significant quantisation noise in the end.

6.2.2 Measurement of Information

Shannon entropy

To quantify the information content, the classical one-dimensional (1D) Shannon entropy [84] with equation shown in Eq. (6.1) is selected.

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x) \tag{6.1}$$

Although the Shannon entropy was designed for 1D data, it can also be implemented to two-dimensional (2D) data by ignoring the 2D spatial correlations and summing $p(x)\log_2 p(x)$ over the histogram of the 2D data. As only discrete data can have a meaningful Shannon entropy, the entropy can only be calculated for quantised holograms and target images, which are usually quantised to less than 8 bit depth.

Delentropy

To account for 2D spatial correlation, the delentropy [86] is also used. Delentropy is an extension of the 1D Shannon entropy that it first computes the gradient (del) vector field image, whose entropy is then named as the delentropy, so that the spatial image structure and pixel co-occurrence can be captured [86].

(a) Sample image (b) x-derivative (c) y-derivative

Fig. 6.4 Del operation on a sample image

As an example, the sample image in Fig. 6.4 (a) is the file of filename '0500.png' under the 'DIV2K_train_HR' folder sourced from the DIV2K dataset [87]. The sample image is calculated to have a Shannon entropy of 7.502 bits/pixel.

$$H_{PGS}(f_x, f_y) \leq \frac{H(f_x) + H(f_y)}{2} \tag{6.2}$$

By taking the x-derivative (f_x) and y-derivative (f_y) as shown in Fig. 6.4 (b) and (c), and using the Papoulis Generalized Sampling (PGS) [88] theory, the delentropy is calculated using Eq. (6.2)[86] to be 5.867 bits/pixel.

6.3 Results

6.3.1 Targets at far field (Fraunhofer region)

In this subsection, the target images are set at far field, so the Fraunhofer diffraction formula in Eq. (2.31) is used.

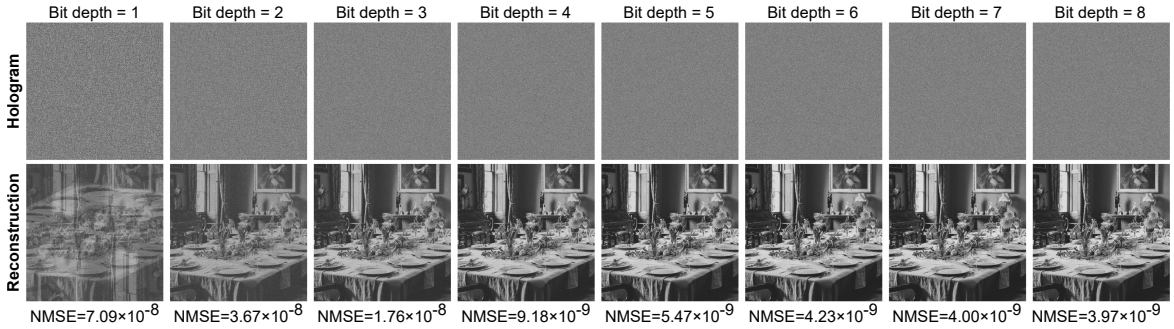


Fig. 6.5 Holograms generated at bit depths level from 1 to 8 and their according reconstructions at far field

An example run of the quantised GS algorithm (in Fig. 6.3) for the sample image in Fig. 6.4 (a) is shown in Fig. 6.5, which demonstrates qualitatively how the reconstruction quality improves with the increase in the bit depth of the hologram, and also quantitatively as the NMSE between the reconstruction and target image has shown a decreasing trend as the bit depth of hologram increases. The rotational symmetry in the reconstruction of the hologram with bit depth 1 is explained by the conjugate properties of Fourier Transforms in Section 2.2.3.

Then the same quantised GD algorithm is run on the 800 images in the 'DIV2K_train_HR' folder in the DIV2K dataset [87], for hologram bit depth set to integers ranging from 1 to 8, with the total number of iterations (N) set to 100.

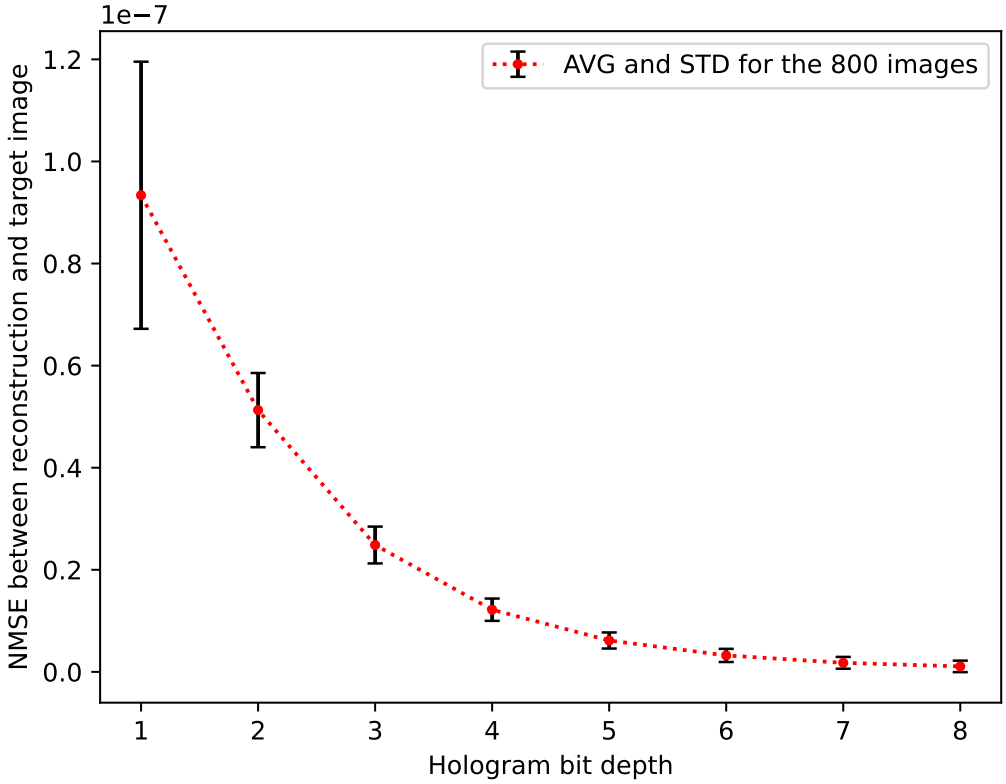


Fig. 6.6 The average and standard deviation of the far-field reconstruction errors among the 800 target images plotted against the hologram bit depth

For the 800 target images, the average (AVG) and standard deviation (STD) of the NMSE values between the far-field reconstructions of the holograms and their corresponding target images are plotted against the hologram bit depth in Fig. 6.6 (the raw data is accessible from the published research dataset [89]). It can be concluded that, the NMSE between the resulting reconstructions and their according target images decreases as the hologram bit depth increases, inferring that holograms with higher bit depth carries more sufficient information in order to better reconstruct the target images.

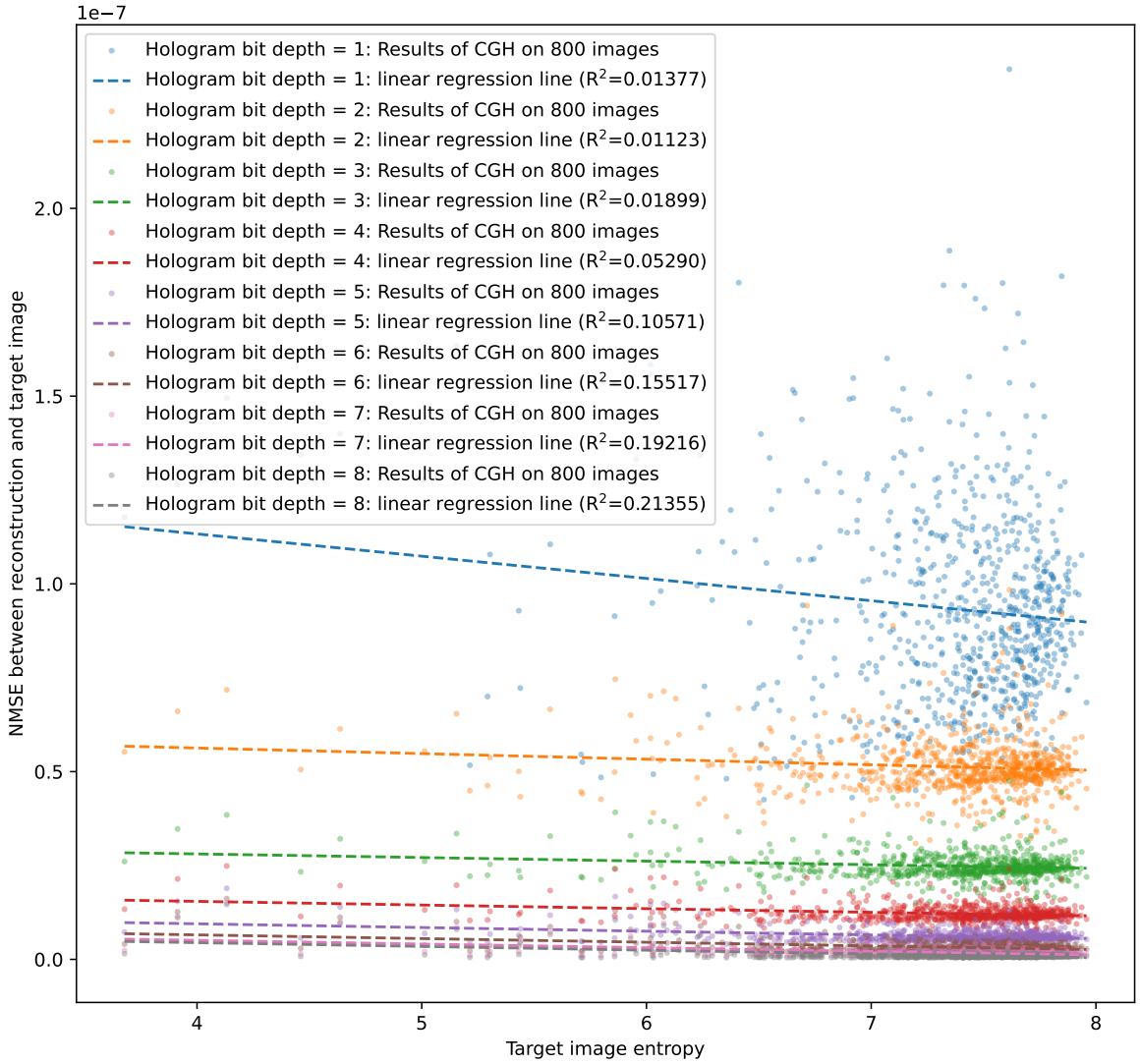


Fig. 6.7 Scatter plot of the far-field reconstruction errors v.s. entropy of target image among the 800 target images, with different colours indicating hologram bit depth constraints

Then the entropy of each of the 800 target images are calculated using Eq. (6.1), and a scatter plot of the NMSE between the far-field reconstruction and target image against the target image entropy is plotted for all 800 target images run with each of the 8 hologram bit depth levels, as shown in Fig. 6.7. To avoid the effect of the different initial random phases on the final result, 5 different randomly generated initial phases are used for each run; however, the result [89] has shown negligible difference between the runs using different random initial phase holograms. Unfortunately, as the linear regressions between NMSE and entropies of target image have coefficients of determination (R^2) much less than 0.5, no correlation has

been found between the NMSE and the target image entropy. Therefore, the Shannon entropy cannot be used to quantify the difficulty of CGH for a given target image.

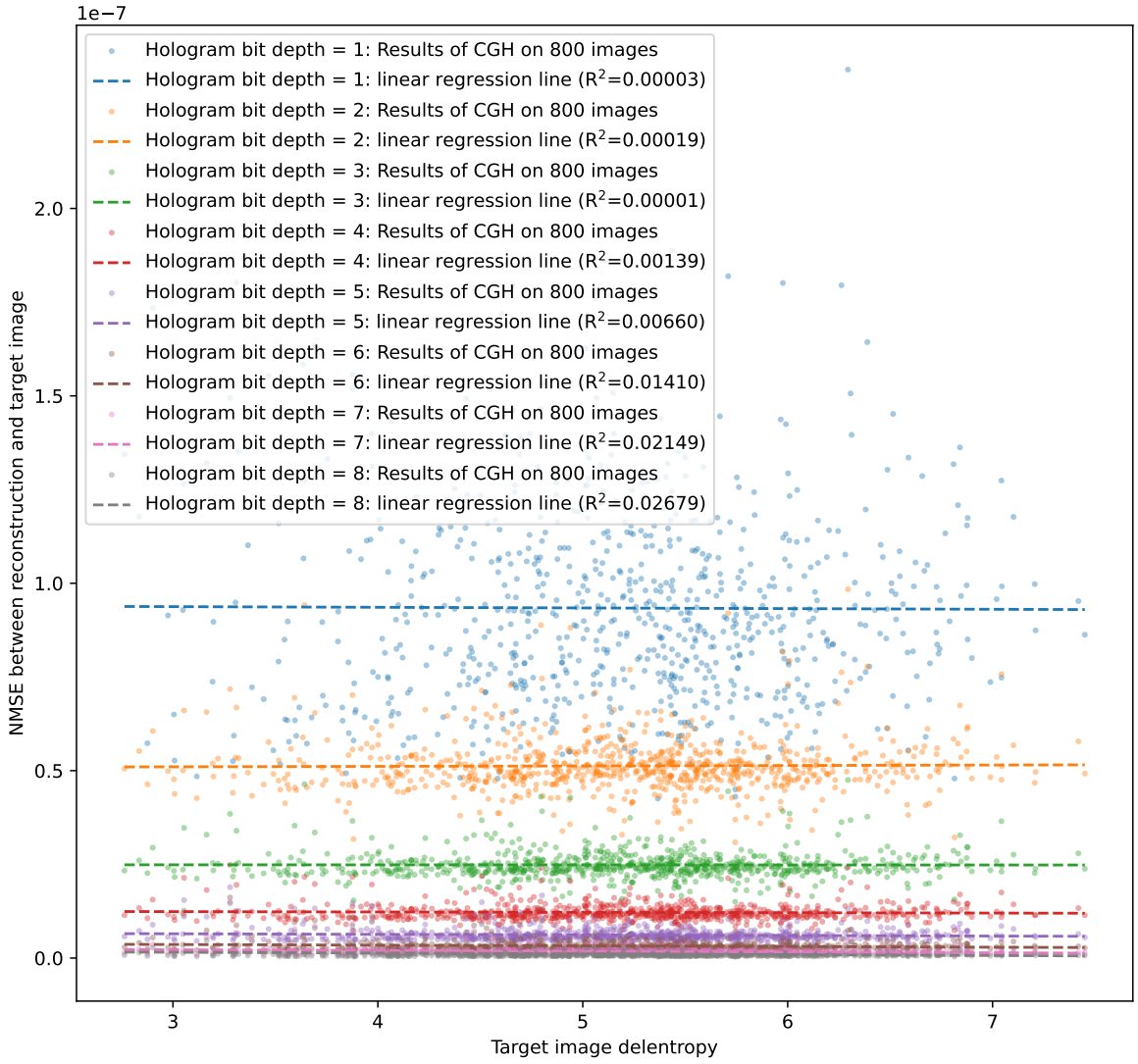


Fig. 6.8 Scatter plot of the far-field reconstruction errors v.s. delentropy of target image among the 800 target images, with different colours indicating hologram bit depth constraints

The delentropy of each of the 800 target images are then calculated using the method in Section 6.2.2, and a scatter plot of the NMSE between the far-field reconstruction and target image against the target image delentropy is plotted in Fig. 6.8 for the 800 target images. And again, as all the R^2 are much less than 0.5, no correlation is shown between the NMSE and the target image delentropy, inferring that the 2D delentropy also fails to predict the error of CGH for a given target image.

6.3.2 Targets at near field (Fresnel region)

The target images are now set to near field, where the Fresnel diffraction formula in Eq. (2.29) applies; therefore the FFT and IFFT stages in Fig. 6.3 are modified to include the phase term in Eq. (2.29), and for experimental purpose, the distance (z) is set at $10cm$, the hologram's pixel pitch (sampling resolution of x and y) has a size of $13.62\mu m$ and the incident light's wavelength (λ) is $532nm$.

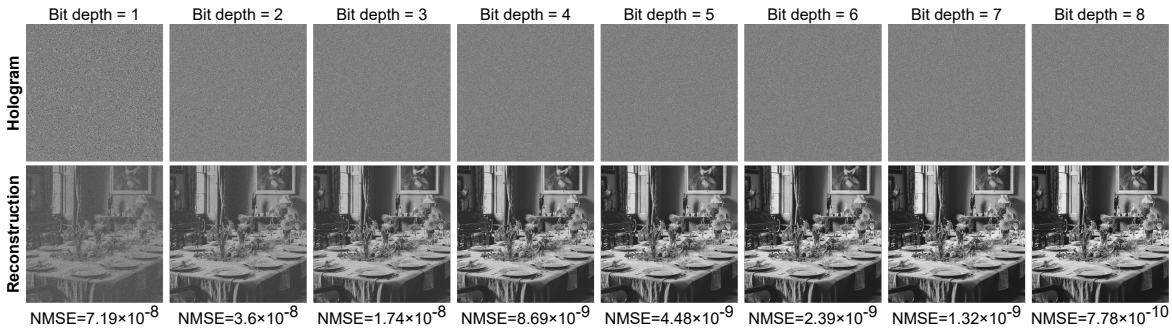


Fig. 6.9 Holograms generated at bit depths level from 1 to 8 and their according reconstructions at near field

Fig. 6.9 shows both qualitatively and quantitatively how the reconstruction quality improves (i.e. NMSE decreases) with the increase in the bit depth of the hologram. Such trend is the same for target images placed at near field as those placed at far field in Fig. 6.5. The rotational symmetry is gone for the binary phase hologram (bit depth = 1) due to the extra phase term in the Fresnel diffraction formula making the product of binary phase hologram and the phase term to be complex-valued whose complex conjugate does not equal to itself; however, the complex conjugate wouldn't disappear, but it will appear at a different distance to where the target image is set at, leading to extra defocused noise onto the reconstruction plane. Nevertheless, the trend of NMSE shows that holograms with higher bit depth produces better quality in the reconstruction plane.

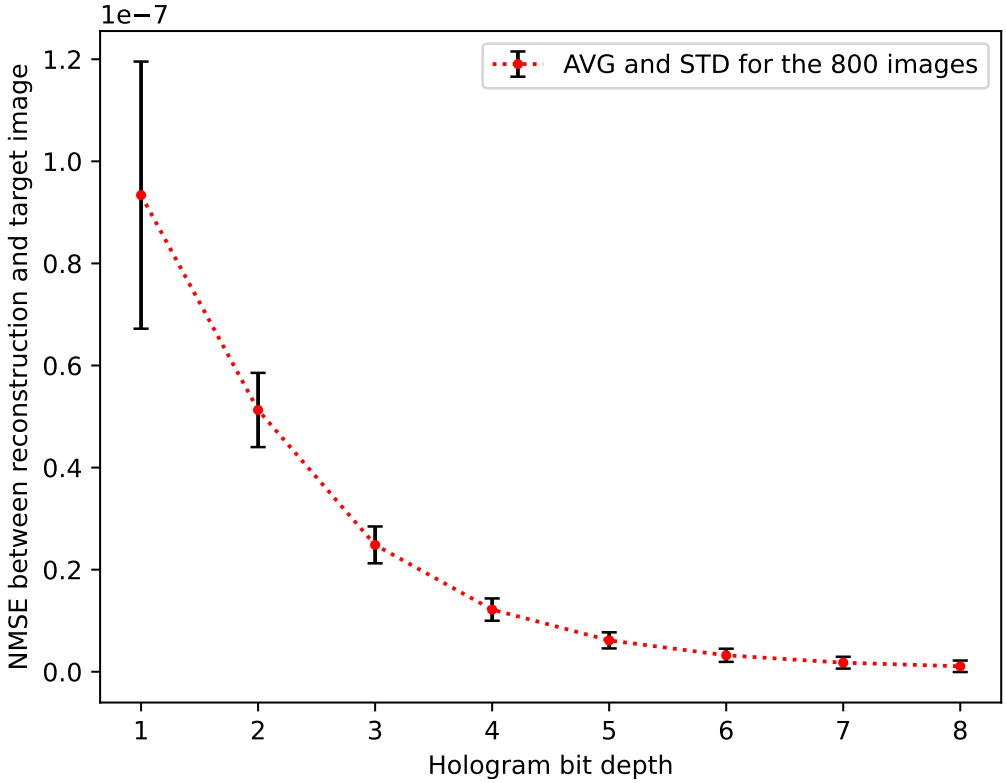


Fig. 6.10 The average and standard deviation of the near-field reconstruction errors among the 800 target images plotted against the hologram bit depth

Fig. 6.10 plots the average (AVG) and standard deviation (STD) of the NMSE values between the near-field reconstructions of the holograms and their corresponding target images against the hologram bit depth (the raw data is accessible from the published research dataset [89]). In Fig. 6.10, the same trend as the one for far field in Fig. 6.6 can be observed, where NMSE decreases as hologram bit depth increases, for every single target image. Therefore, the previous conclusion of higher hologram bit depth being able to produce better reconstruction quality is still valid.

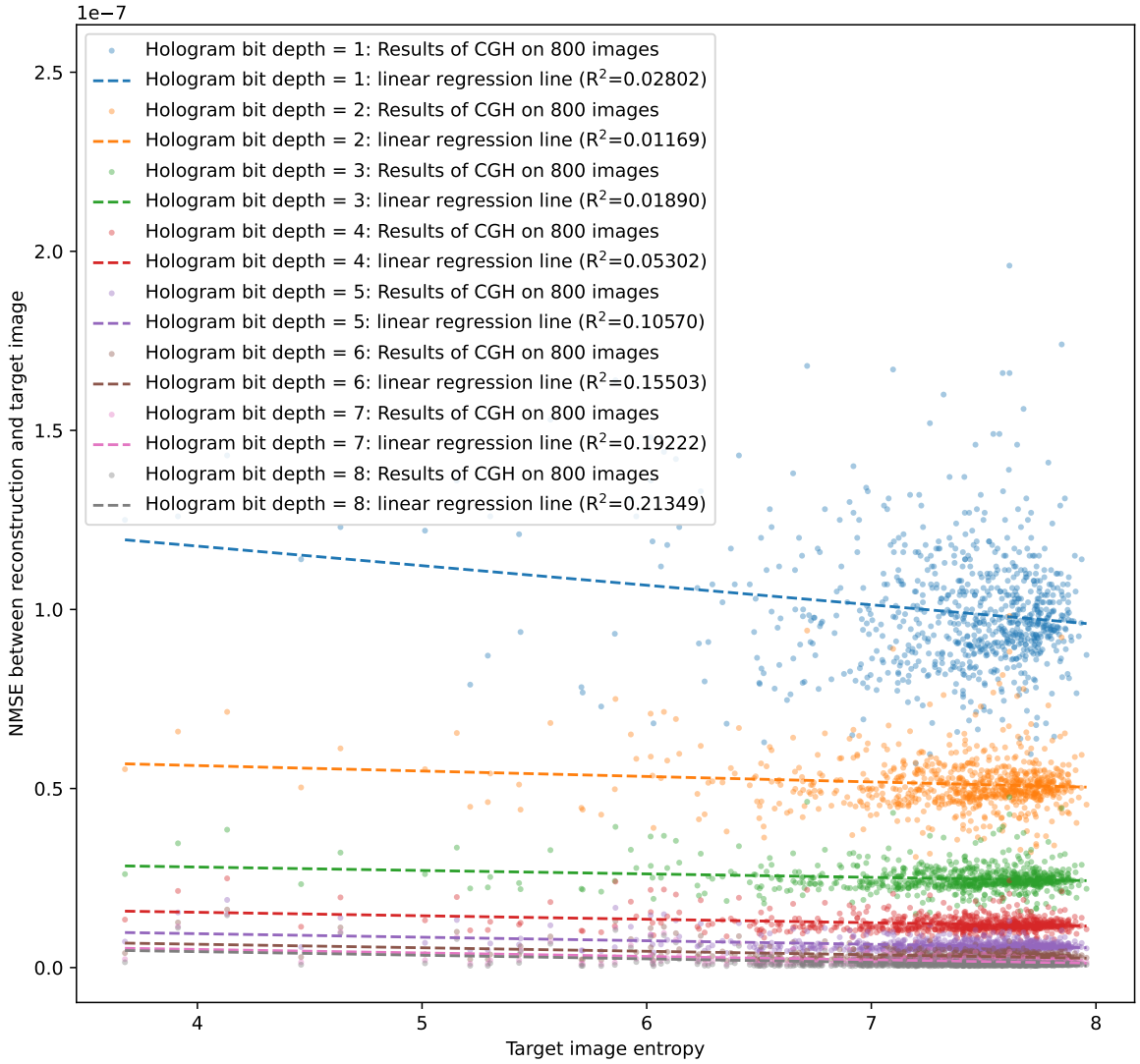


Fig. 6.11 Scatter plot of the near-field reconstruction errors v.s. entropy of target image among the 800 target images, with different colours indicating hologram bit depth constraints

A scatter plot of the NMSE between the near-field reconstruction and target image against the target image entropy is plotted for all 800 target images run with each of the 8 hologram bit depth levels as shown in Fig. 6.11, with the hologram bit depth distinguished by different colours. Unfortunately, as the linear regressions between NMSE and entropies of target image have coefficients of determination (R^2) much less than 0.5, no correlation has been found between the NMSE and the target image entropy. Therefore, the Shannon entropy cannot be used to quantify the difficulty of CGH for a given target image.

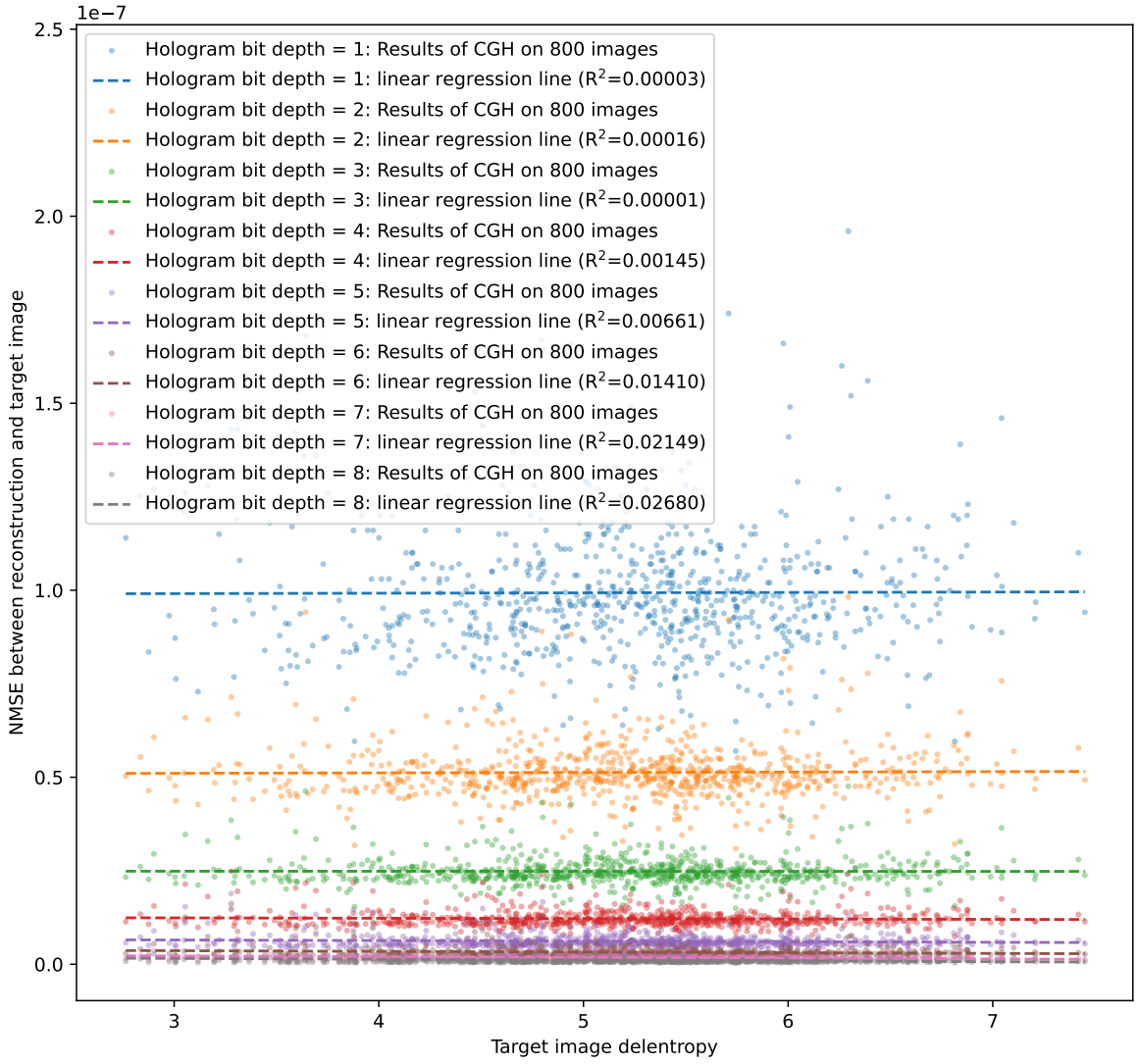


Fig. 6.12 Scatter plot of the near-field reconstruction errors v.s. delentropy of target image among the 800 target images, with different colours indicating hologram bit depth constraints

The scatter plot ofthe NMSE between the near-field reconstruction and target image against the target image delentropy in Fig. 6.12 again shows no correlation between the NMSE and the delentropy of target image. Such 'no correlation' result is the same as the results for far field investigated in Section 6.3.1, confirming that neither entropy nor delentropy is suitable for quantifying how difficult a target image is for CGH.

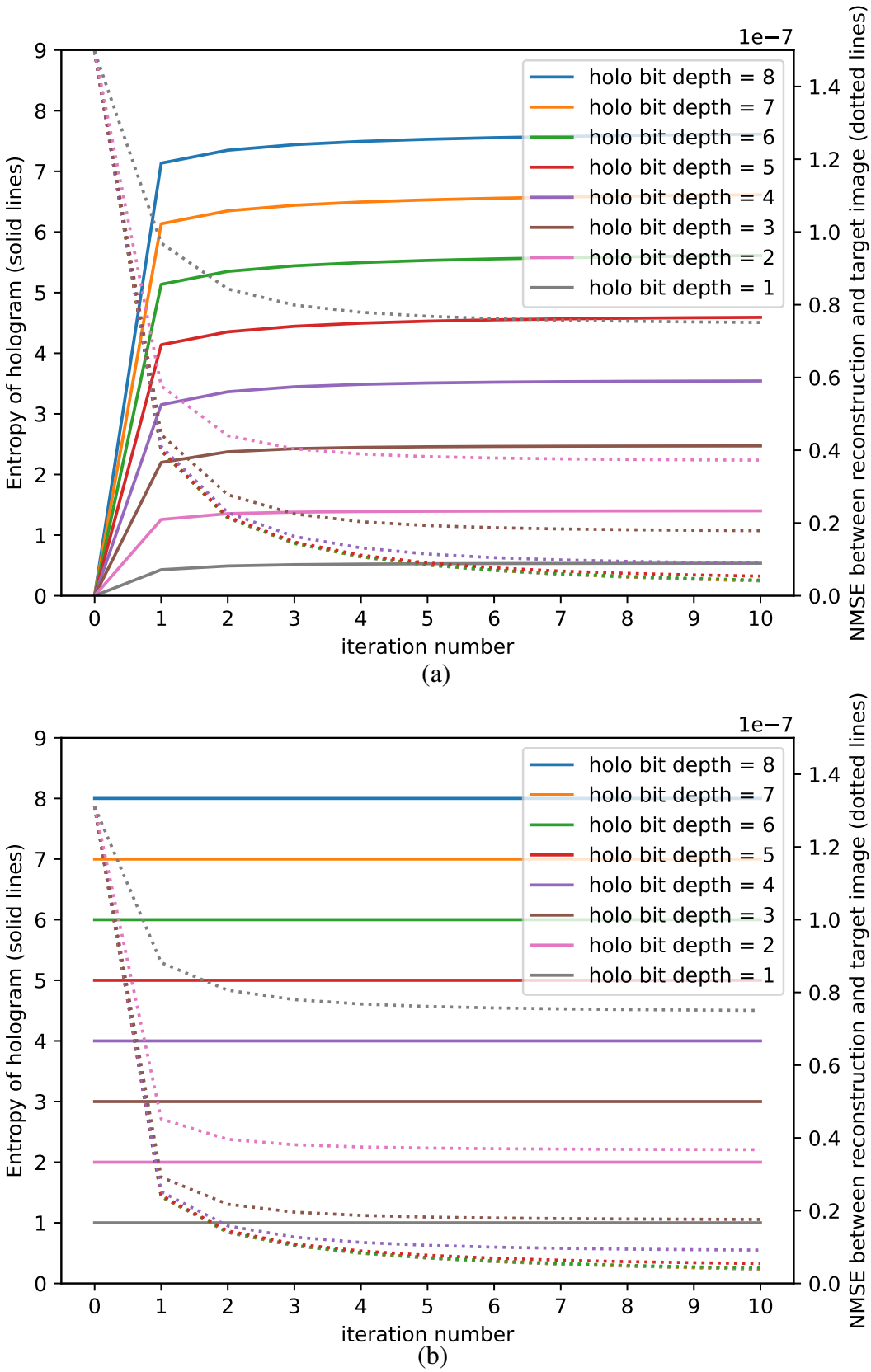


Fig. 6.13 Hologram entropy (solid lines) and NMSE (dotted lines) plotted against the iteration number in a GS run with the initial hologram phase ($\angle A$) being (a) zeros (b) random

The research moves on to investigate the entropy of holograms. In an example run for a quantised hologram generation in the near field, both the hologram entropy and the NMSE between reconstruction and target image are recorded and plotted in Fig. 6.13 (a) and (b) for the first 20 iterations. As randomly generated holograms will by definition create holograms with high entropy, additional experiment has been carried out with initial phase set to zeros (i.e. $\angle A$ is set to zeros at $n = 0$ in Fig. 6.3). Therefore, two diagrams are plotted in Fig. 6.13, with Fig. 6.13 (a) having initial phase of zeros and Fig. 6.13 (b) having random initial phase. Both Fig. 6.13 (a) and (b) have the horizontal axis to be the iteration number n, and the vertical axis in the left is the entropy of hologram (corresponding to solid lines in the plot), while the vertical axis in the right is the NMSE (corresponding to dotted lines in the plot). Colour coding is used to distinguish between the 8 different runs where hologram bit depth is set from 1 to 8.

The solid lines in Fig. 6.13 (a) show that the entropy of hologram keeps increasing towards a value lower than the bit depth, with their corresponding NMSE between reconstruction and target image (dotted lines) decreasing. Such trend can be explained qualitatively that, as the iteration goes on, the hologram is attempting to contain more information to sustain a better reconstruction, while the entropy cannot exceed or even reach the bit depth level. On the other hand, the random initial phase plotted in Fig. 6.13 (b) has a constant entropy approximately equal to the bit depth, which infers that the iterations are improving the reconstruction quality without reducing the information entropy of the hologram. In both cases, the entropy of the hologram does not decrease at any iteration, and the final NMSE does not have significant reduction when the hologram's bit depth exceeds 5.

The entropy of final hologram in Fig. 6.13 (a) is lower than that in Fig. 6.13 (b), although the final NMSE are similar. If the computer-generated holograms were to undergo a lossless compression, the hologram in Fig. 6.13 (a) would be better compressed than the hologram in Fig. 6.13 (b) as the entropy denotes the compression limit, assuming that the holograms are treated as 1D arrays when compressing. Therefore if hologram compression is of a concern, then starting with a low entropy initial hologram in the CGH process is recommended. In case if the reconstruction quality is not as high of a priority than making holograms to occupy less storage space, it is also recommended to use quantised GS algorithm with 5 bit depth instead of 6-7 bit depth as the final reconstruction quality will not degrade significantly.

6.4 Summary

By carrying out the quantised GS algorithm on 800 sample target images placed at both far field (Fraunhofer diffraction) and near field (Fresnel diffraction), and computing the entropy and delentropy of the target images and holograms, this chapter reached the conclusion that, holograms with higher bit depth can sustain more information therefore producing better quality reconstructions. However, the quality of the reconstruction is not correlated to either the entropy or the delentropy of the target image, so neither entropy nor delentropy can quantify how difficult an image is for phase-only hologram generation. Additionally, the entropy of the hologram generated using quantised GS algorithm is not only bounded by the hologram bit depth, but also affected by the entropy of the initial phase. For applications where holograms file size is a high priority, it is advised to start with a low entropy initial phase (e.g. all zeros) rather than a random initial phase and it is recommended to reduce the hologram bit depth limit, for lower entropy hologram generation.

Chapter 7

Conclusion and Outlook

The research presented in this thesis has focused on advancing the field of computer-generated holography (CGH) through the development and optimisation of phase-only holograms. By exploring and implementing various algorithms, we have identified the strengths and weaknesses of each approach in terms of computational efficiency and reconstruction quality.

Chapter 3 proposed the Digital Pre-Distorted One-Step Phase Retrieval (DPD-OSPR) algorithm, described the experimental setup, explained the method for determining the DPD curve, and then applied the DPD curve to improve holographic projection quality. The results demonstrated that DPD can significantly enhance the quality of the reconstructed images. On the grey-scale ramp target, the DPD-OSPR method reduced the NMSE by a stunning 95.45%. Then the DPD-OSPR was applied on two sample images, it was observed that more details were shown in the replay field, and the NMSE's of the two example images were reduced by 19.86% and 15.64% respectively. As the DPD is a one-to-one mapping, the extra computation required is negligible. The effectiveness of the proposed DPD-OSPR method to improve reconstruction quality on the existing OSPR algorithm while still keeping its ability for real-time holography was demonstrated.

Chapter 4 focused on the optimisation of phase-only holograms, and proved the effectiveness of using the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) algorithm for CGH. Then the novel Target Image Phase Optimisation (TIPO) technique was proposed, which optimises the phase of the target image instead of the phase of the hologram. The L-BFGS algorithm was shown to offer efficient convergence, improving the quality of the generated holograms. Then the two existing 3D CGH optimisation methods, the sum-of-

hologram (SoH) and sum-of-loss(SoL) techniques were investigated. The novel method using L-BFGS optimiser with Sequential Slicing (SS) technique was proposed to generate phase-only hologram for multi-depth target, which is faster than SoL and of better quality than SoH. The L-BFGS with SS technique has demonstrated a good suppression on the quality imbalance across the multi-depth slices, benefiting from the nature of L-BFGS being a second order optimiser, which implicitly records the historical gradients by other slices for the determination of the descent direction. For both GD and L-BFGS optimisation algorithms, the SS technique runs faster and produces better reconstruction quality than the simple SoH technique, and it is much quicker than the SoL technique especially when the number of slices get large. The proposed SS method also proved effective for complicated 3D targets and demonstrated great ability of time-limited applications.

Chapter 5 proposed the Multi-Frame Holograms Batched Optimisation (MFHBO) algorithm to generate multi-frame binary-phase holograms to be displayed on the high-refresh-rate binary-phase SLM in the lab. By comparing simulation and optical experiment results, the proposed MFHBO method was shown to have superior performance in reducing noise and improving reconstruction quality for multi-frame holograms than the existing multi-frame binary-phase holograms generation methods OSPR and AD-OSPR on the holographic projector with binary-phase SLM, for all the single-slice far-field targets and the multi-slice near-field targets tested. Although the proposed MFHBO method is slower than the existing OSPR and AD-OSPR methods, its much better reconstruction quality makes it especially suitable for pre-computed high-quality hologram applications. Its strong advantage for high contrast target also makes it well-suited for photo-lithography applications. The proposed method can also be adapted for multi-level SLM's in the future once high-refresh-rate high-resolution SLM's are available.

Finally, Chapter 6 explored the information capacity of phase-only CGH. This chapter examined quantisation effects on hologram bit depth and their impact on reconstruction quality, and reached the conclusion that, holograms with higher bit depth can sustain more information therefore producing better quality reconstructions. However, the quality of the reconstruction is not correlated to either the entropy or the delentropy of the target image, so neither entropy nor delentropy can quantify how difficult an image is for phase-only hologram generation. Additionally, the entropy of the hologram generated using quantised GS algorithm is not only bounded by the hologram bit depth, but also affected by the entropy of the initial phase. For applications where holograms file size is a high priority, it is advised to start with a low entropy initial phase rather than a random initial phase and it is

recommended to reduce the hologram bit depth limit. In future work, a suitable metric will be the goal to quantify how difficult any image is for phase hologram generation.

Overall, this thesis has contributed a series of novel phase retrieval techniques, with improvements achieved in the computational speed and reconstruction quality of CGH. While current methods provide a foundation for practical implementations, there still remains substantial room for improvement, particularly in achieving real-time, high-quality holography. Future research should continue to build on these findings, exploring novel algorithms and advancing technologies to realise the full potential of CGH.

References

[1] Jana Skirnewskaja, Yunuen Montelongo, Jinze Sha, and Timothy D. Wilkinson. Holographic lidar projections with brightness control. In *Imaging and Applied Optics Congress 2022 (3D, AOA, COSI, ISA, pcAOP)*, page 3F2A.6. Optica Publishing Group, 2022.

[2] Jinze Sha, Andrew Kadis, Fan Yang, and Timothy D. Wilkinson. Limited-memory bfgs optimisation of phase-only computer-generated hologram for fraunhofer diffraction. In *Digital Holography and 3-D Imaging 2022*, page W3A.3. Optica Publishing Group, 2022.

[3] Andrew Kadis, Benjamin Wetherfield, Jinze Sha, Fan Yang, Youchao Wang, and Timothy D. Wilkinson. Effect of bit-depth in stochastic gradient descent performance for phase-only computer-generated holography displays. *London Imaging Meeting*, 3:36–40, 7 2022.

[4] Jinze Sha, Andrew Kadis, Fan Yang, Youchao Wang, and Timothy D. Wilkinson. Multi-depth phase-only hologram optimization using the l-bfgs algorithm with sequential slicing. *J. Opt. Soc. Am. A*, 40(4):B25–B32, Apr 2023.

[5] Jinze Sha, Adam Goldney, Andrew Kadis, Jana Skirnewskaja, and Timothy D. Wilkinson. Digital pre-distorted one-step phase retrieval algorithm for real-time hologram generation for holographic displays. *Journal of Imaging Science and Technology*, 67(3):030405–1–030405–1, 2023.

[6] Jana Skirnewskaja, Yunuen Montelongo, Jinze Sha, Phil Wilkes, and Timothy D. Wilkinson. Accelerated augmented reality holographic 4k video projections based on lidar point clouds for automotive head-up displays. *Advanced Optical Materials*, 12(12):2301772, 2024.

[7] Roubing Meng, Jinze Sha, Zhongling Huang, and Timothy D. Wilkinson. Extending FOV of holographic display with alternating lasers. In Peter Schelkens and Tomasz Kozacki, editors, *Optics, Photonics, and Digital Technologies for Imaging Applications VIII*, volume 12998, page 129981J. International Society for Optics and Photonics, SPIE, 2024.

[8] Jinze Sha, Andrew Kadis, Benjamin Wetherfield, Roubing Meng, Zhongling Huang, Dilawer Singh, Antoni Wojcik, and Timothy D. Wilkinson. Information capacity of phase-only computer-generated holograms for holographic displays. In Peter Schelkens and Tomasz Kozacki, editors, *Optics, Photonics, and Digital Technologies for Imaging*

Applications VIII, volume 12998, page 129980J. International Society for Optics and Photonics, SPIE, 2024.

[9] Jinze Sha, Antoni Wojcik, Benjamin Wetherfield, Jianghan Yu, and Timothy D. Wilkinson. Multi frame holograms batched optimization for binary phase spatial light modulators. *Scientific Reports*, 14(1):19380, Aug 2024.

[10] Ivan Y. Lo. A photo of the holographic portrait of dennis gabor, 2018.

[11] Johannes Kalliauer. An illustration of the 'double-slit experiment' in physics, 2017.

[12] Jeff Hecht. *Solid-State and Fiber Lasers*, chapter 8, pages 223–263. John Wiley & Sons, Ltd, 2008.

[13] Arne Nordmann. Wave diffraction in the manner of huygens and fresnel, 2007.

[14] Timothy D. Wilkinson. Lecture notes of 4b11 photonics systems course, 2019. University of Cambridge.

[15] T D Wilkinson, W A Crossland, S T Warr, T C B Yu, A B Davey, and R J Mears. New applications for ferroelectric liquid crystals. *Liquid Crystals Today*, 4(3):1–6, 1994.

[16] J. Freeman. Visor projected helmet mounted display for fast jet aviators using a fourier video projector, 2009. PhD thesis, Department of Engineering, University of Cambridge, United Kingdom.

[17] A. J. Cable. Real-time high-quality two and three-dimensional holographic video projection using the one-step phase retrieval (ospr) approach, 2006. PhD thesis, Department of Engineering, University of Cambridge, United Kingdom.

[18] Allan Weber. Sipi image database - misc, 2022. [retrieved 17 Nov 2022].

[19] Xuetun Zhao. Suzhou center mall, 2017. Suzhou, Jiangsu, China.

[20] John P. McIntire, Paul R. Havig, and Eric E. Geiselman. Stereoscopic 3d displays and human performance: A comprehensive review. *Displays*, 35(1):18–26, 2014.

[21] Simon J. Watt, Kurt Akeley, Marc O. Ernst, and Martin S. Banks. Focus cues affect perceived depth. *Journal of Vision*, 5(10):7–7, 12 2005.

[22] Barbara Klinger. Three-dimensional cinema: The new normal. *Convergence*, 19(4):423–431, 2013.

[23] Catherine Allen and Verity McIntosh. Safeguarding the metaverse. *The Institution of Engineering and Technology*, 2022.

[24] Michael E. McCauley and Thomas J. Sharkey. Cybersickness: Perception of self-motion in virtual environments. *Presence: Teleoperators and Virtual Environments*, 1(3):311–318, 08 1992.

[25] Hyun Taek Kim Eunhee Chang and Byounghyun Yoo. Virtual reality sickness: A review of causes and measurements. *International Journal of Human-Computer Interaction*, 36(17):1658–1682, 2020.

[26] D. Gabor. A new microscopic principle. *Nature*, 161:777–778, 1948.

[27] Eugene Hecht. *Optics*. Pearson Education Limited, 5 edition, 2017.

[28] Michael A. Seldowitz, Jan P. Allebach, and Donald W. Sweeney. Synthesis of digital holograms by direct binary search. *Applied Optics*, 26, 1987.

[29] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220, 1983.

[30] R W Gerchberg. A practical algorithm for the determination of phase from image and diffraction plane pictures. *Optik*, 35:237–246, 1972.

[31] Jingzhao Zhang, Nicolas Pégard, Jingshan Zhong, Hillel Adesnik, and Laura Waller. 3d computer-generated holography by non-convex optimization. *Optica*, 4:1306, 10 2017.

[32] Shujian Liu and Yasuhiro Takaki. Optimization of phase-only computer-generated holograms based on the gradient descent method. *Applied Sciences (Switzerland)*, 10, 2020.

[33] Chun Chen, Byounghyo Lee, Nan-Nan Li, Minseok Chae, Di Wang, Qiong-Hua Wang, and Byoungho Lee. Multi-depth hologram generation using stochastic gradient descent algorithm with complex loss function. *Optics Express*, 29:15089, 5 2021.

[34] Suyeon Choi, Jonghyun Kim, Yifan Peng, and Gordon Wetzstein. Optimizing image quality for holographic near-eye displays with michelson holography. *Optica*, 8:143, 2 2021.

[35] Isaac Newton. *Opticks*. Dover Press, 1704.

[36] Christiaan Huygens. *Traite de la lumiere. Où sont expliquées les causes de ce qui luy arrive dans la reflexion, & dans la refraction. Et particulierment dans l'etrange refraction du cristal d'Islande, par C.H.D.Z. Avec un Discours de la cause de la pesanteur*. chez Pierre Vander Aa marchand libraire, 1690.

[37] Thomas Young. Ii. the bakerian lecture. on the theory of light and colours. *Philosophical Transactions of the Royal Society of London*, 92:12–48, 1802.

[38] Augustin Jean Fresnel. *Memoir on the Diffraction of Light*. 1826.

[39] John Daintith. *A Dictionary of Physics*. Oxford University Press, 2009.

[40] Timothy D. Wilkinson. *Electrical Data Book*. Cambridge University Engineering Department, 2017.

[41] James Clerk Maxwell. Viii. a dynamical theory of the electromagnetic field. *Philosophical Transactions of the Royal Society of London*, 155:459–512, 1865.

[42] A. Einstein. On a heuristic point of view about the creation and conversion of light. *Annalen der Physik*, 322(6):132–148, 1905.

[43] Gould R. Gordon. The laser, light amplification by stimulated emission of radiation. page 128. Ann Arbor, 6 1959.

[44] Edwin Cartlidge. Theodore maiman 1927–2007. *Physics World*, 20, 2007.

[45] John B Develis, Merrimack College, North Andover, and George O Reynolds. Three dimensional hologram reconstruction and image speckle, 1966.

[46] A. J. Cable, E. Buckley, P. Mash, N. A. Lawrence, T. D. Wilkinson, and W. A. Crossland. 53.1: Real-time binary hologram generation for high-quality video projection applications. *SID Symposium Digest of Technical Papers*, 35:1431, 2004.

[47] Tim Stangner, Hanqing Zhang, Tobias Dahlberg, Krister Wiklund, and Magnus Andersson. Step-by-step guide to reduce spatial coherence of laser light using a rotating ground glass diffuser. *Applied Optics*, 56:5427, 7 2017.

[48] Linxiao Deng, Tianhao Dong, Yuwei Fang, Yuhua Yang, Chun Gu, Hai Ming, and Lixin Xu. Speckle reduction in laser projection based on a rotating ball lens. *Optics and Laser Technology*, 135, 3 2021.

[49] Philip J W Hands, Calum M Brown, Daisy K E Dickinson, Stephen M Morris, and Jia-De Lin. Liquid-crystal lasers: Recent advances and future opportunities, 2022.

[50] Joseph W. Goodman. *Introduction to Fourier Optics, Fourth Edition*. W. H. Freeman, 2017.

[51] M. Schadt and W. Helfrich. Voltage-dependent optical activity of a twisted nematic liquid crystal. *Applied Physics Letters*, 18, 1971.

[52] Dennis R. Pape and Larry J. Hornbeck. Characteristics of the deformable mirror device for optical information processing. *Optical Engineering*, 22, 1983.

[53] Kristina M. Johnson, Douglas J. McKnight, and Ian Underwood. Smart spatial light modulators using liquid crystals on silicon. *IEEE Journal of Quantum Electronics*, 29, 1993.

[54] Yongmin Lee, James Gourlay, William J. Hossack, Ian Underwood, and Anthony J. Walton. Multi-phase modulation for nematic liquid crystal on silicon backplane spatial light modulators using pulse-width modulation driving scheme. *Optics Communications*, 236, 2004.

[55] S. E. Broomfield, M. A.A. Neil, E. G.S. Paige, and G. G. Yang. Programmable binary phase-only optical device based on ferroelectric liquid crystal slm. *Electronics Letters*, 28, 1992.

[56] Zhengzhong Huang and Liangcai Cao. Quantitative phase imaging based on holography: trends and new perspectives. *Light: Science & Applications*, 13(1):145, Jun 2024.

[57] Claude Sammut and Geoffrey I. Webb, editors. *Mean Squared Error*, pages 653–653. Springer US, Boston, MA, 2010.

[58] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[59] Han Jin Yang, Jeong Sik Cho, and Yong Hyub Won. Reduction of reconstruction errors in kinoform cghs by modified simulated annealing algorithm. *Journal of the Optical Society of Korea*, 13, 2009.

[60] Guo zhen Yang, Bi zhen Dong, Ben yuan Gu, Jie yao Zhuang, and Okan K. Ersoy. Gerchberg–saxton and yang–gu algorithms for phase retrieval in a nonunitary transform system: a comparison. *Appl. Opt.*, 33(2):209–218, Jan 1994.

[61] Haichao Wang, Weirui Yue, Qiang Song, Jingdan Liu, and Guohai Situ. A hybrid gerchberg–saxton-like algorithm for doe and cgh calculation. *Optics and Lasers in Engineering*, 89:109–115, 2017. 3DIM-DS 2015: Optical Image Processing in the context of 3D Imaging, Metrology, and Data Security.

[62] Pengcheng Zhou, Yan Li, Shuxin Liu, and Yikai Su. Dynamic compensatory gerchberg-saxton algorithm for multiple-plane reconstruction in holographic displays. *Optics Express*, 27:8958, 3 2019.

[63] Edward Buckley. 70.2: Invited paper: Holographic laser projection technology. *SID Symposium Digest of Technical Papers*, 39(1):1074–1079, 2008.

[64] Andrzej Kaczorowski, George S. D. Gordon, and Timothy D. Wilkinson. Adaptive, spatially-varying aberration correction for real-time holographic projectors. *Opt. Express*, 24(14):15742–15756, Jul 2016.

[65] Xu Guan, Su Jian, Pan Hongda, Zhang Zhiguo, and Gong Haibin. An image enhancement method based on gamma correction. In *2009 Second International Symposium on Computational Intelligence and Design*, volume 1, pages 60–63, 2009.

[66] Sung Jin Kang and Sung Il Chien. Apl-adaptive inverse gamma correction for improving gray-level linearity of pdp-tv. *Molecular Crystals and Liquid Crystals*, 499(1):185/[507]–192/[514], 2009.

[67] Po-Ming Lee and Hung-Yi Chen. Adjustable gamma correction circuit for tft lcd. In *2005 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 780–783 Vol. 1, 2005.

[68] C. Alejandro Párraga, Jordi Roca-Vila, Dimosthenis Karatzas, and Sophie M. Wuerger. Limitations of visual gamma corrections in lcd displays. *Displays*, 35:227–239, 2014.

[69] Charles Poynton. *Digital Video and HD: Algorithms and Interfaces*. Morgan Kaufmann, 2 edition, 2012.

[70] The MathWorks Inc. Matlab version: 9.13.0 (r2022b), 2022.

[71] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 2006.

[72] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

[73] Tijmen Tieleman, Geoffrey Hinton, et al. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4:26–31, 2012.

[74] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. 2015.

[75] Dong C. Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical Programming*, 45, 1989.

[76] G. Cybenko, D.P. O'Leary, and J. Rissanen. *The Mathematics of Information Coding, Extraction and Distribution*. The IMA Volumes in Mathematics and its Applications. Springer New York, 1998.

[77] S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79 – 86, 1951.

[78] Xunying Liu, Shansong Liu, Jinze Sha, Jianwei Yu, Zhiyuan Xu, Xie Chen, and Helen Meng. Limited-memory bfgs optimization of recurrent neural network language models for speech recognition. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2018-April:6114–6118, 9 2018.

[79] Jun Amako, Hirotsuna Miura, and Tomio Sonehara. Speckle-noise reduction on kinoform reconstruction using a phase-only spatial light modulator. *Appl. Opt.*, 34(17):3165–3171, Jun 1995.

[80] Nicolas Bacaër. *Verhulst and the logistic equation (1838)*, pages 35–39. Springer London, London, 2011.

[81] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. volume 32, 2019.

[82] Jinze Sha, Antoni Wojcik, Benjamin Wetherfield, Jianghan Yu, and Timothy D. Wilkinson. Research data supporting "multi-frame holograms batched optimization". Apollo - University of Cambridge Repository, 2024.

[83] P. W. M. Tsang, T.-C. Poon, and Y. M. Wu. Review of fast methods for point-based computer-generated holography. *Photon. Res.*, 6(9):837–846, Sep 2018.

[84] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27, 1948.

[85] Joel S Kollin. Design and information considerations for holographic television, 1988.

[86] Kieran G. Larkin. Reflections on shannon information: In search of a natural information-entropy for images, 2016.

[87] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. July 2017.

[88] A. Papoulis. Generalized sampling expansion. *IEEE Transactions on Circuits and Systems*, 24(11):652–654, 1977.

[89] Jinze Sha, Andrew Kadis, Benjamin Wetherfield, Roubing Meng, Zhongling Huang, Dilawer Singh, Antoni Wojcik, and Timothy D. Wilkinson. Research data supporting 'information capacity of phase-only computer-generated holograms for holographic displays', 2024.