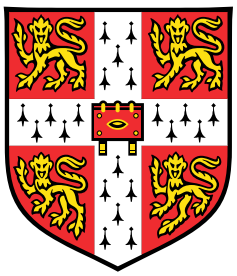
Adaptive Aberration Correction for Holographic Projectors



Andrzej Kaczorowski

Department of Engineering
University of Cambridge

This dissertation is submitted for the degree of
Doctor of Philosophy

Jesus College

February 2017

I dedicate this thesis to my parents, Danuta and Roman for their love and support,
and to my teachers and supervisors[1] for the best gift I received from them - an education.

"A PhD is never finished, only abandoned"

- author unknown[2]

[1] In chronologic order: Mrs Apolonia Hejzner, Ms Malgorzata Szopa, Mrs Irena Glowacka, Dr Maciej Wisniewski, Dr Alessandro De Vita, Dr Cyril Renaud, Prof Timothy Wilkinson, Dr Phillip Hands

[2] overheard from Dr George Gordon

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.

Andrzej Kaczorowski
February 2017

Acknowledgements

I would like to thank the Centre for Doctoral Training in Integrated Photonic and Electronic Systems for the financial and not-so-financial support.

My dear supervisor, Professor Timothy Wilkinson deserves a huge amount of credit. He was always there for me when I desperately needed help, while allowing me to explore my own wild ideas and tolerating my terrible working hours.

A special thanks goes to Dr George Gordon, who familiarised me with nuances of the academic world, such as publishing articles and applying for grant money. Without him, our two papers would probably never have been published. George taught me that no question is too stupid, and I was asking him a lot.

I would like to thank Dr Phillip Hands for our fruitful collaboration and his mentoring.

I owe a thank you to various CMMPE people for small, medium and large favours: 3D printed gears by Alex, MatLAB OSPR code by SJ, introduction to 3D holo by Yunuen, assembled computer by Rachel, and I'm sure there are people I've forgotten...

I certainly would not be where I am now without Penteract28 and all of its members. Special thanks goes to Dr Darran Milne for great supervision, countless discussions about quantum gravity, quantum computing, guitars, and singing lessons, and for proof-reading this work. Every single P28 member deserves special credit: Tom for standing up for me and giving me a cup of coffee in the morning, Konstantin for being a lovely Eastern-European and oversaying the word "hash table", Marcin for a chilled-out attitude that made me feel at home, Roman for the programming in-jokes, and Eduardo for dreaming big. Thanks to all the project students, who taught me as much as I taught them: Ziheng, Vamsee, Shengjun, Yuan. Last year was an amazingly productive and stimulating period, and that is just the start...

I would like to thank all my friends for doing what friends should do: providing enough entertainment and dealing with my grumpiness when things go wrong. Listing all the names of important people, without particular order: Bartek and Manjula Jedrzejewski, Tomek Mentzen, Stachu Czerniawski, Amrita Desai, Greg Racz, Kuba and Suki Sikorowski, Karolina Ciecholewska, Daniel Zurawski, Helen Fox, Tomasz Cebo, Antoni and Ania Wrobel, Kasia

Buzanska, Michal Wlodarski, Gosia Gancarz, Peter Nixon, Natalia Hartung, and Aleksandra Pe.

I would like to thank Mari Monti Muccioli for being the best girlfriend-cum-PA trainee I could ever dream of.

Finally, great thanks should go to my two Princess Leias: Anna Fyda (aka the mysterious lady with the rose) and Aleksandra Pe (aka the busy businesswoman).

Abstract

This work builds up on the greatest minds of Cambridge Holography: Adrian Cable, Edward Buckley, Jonathan Freeman, and Christoph Bay. The methods designed here were initially targetted at Light Blue Optics (LBO), a technology start-up sun out of the engineering department.

Cable and Buckley, the founders of LBO, developed a One-Step Phase Retrieval (OSPR) algorithm which was the first to provide high-quality real-time hologram generation using general-purose hardware while Freeman studied aberration correction on a Ball-lens projector. The method developed by him, termed Pixel-To-Wrapped Phase Summation, was very robust in eliminating the aberrations, but extremely slow in operation. Even rewritten using highly-parallel GPU programming, it needed 4 minutes to compute a single frame. In the world of real-time holographic projection, it was still 3 orders of magnitude too slow.

Addressing this issue, a novel variant of OSPR suited for spatially-varying aberration is presented. The algorithm is based on the continuity of the aberration-correcting phase mask and combines the approaches of Cable, Buckley and Freeman to provide real-time hologram generation. It can be tuned to any accuracy, by dividing the replay field into a set of masks, each one correcting a particular region of the image. In the proof-of-principle ball lens projector, 6 regions proved sufficient to correct a replay field 90x45 degrees. The algorithm incorporates various corrections, including aberration, distortion, and pixel shape envelope. An efficient implementation using high-performance computing on the GPU achieved a real-time hologram generation reaching up to 12 frames per second on a mid-range GPU while incororating all of the corrections.

The next topic studied throughout this thesis is an adaptive optical correction. In previous work, Freeman used ZEMAX ray-tracing software to model the projector's aberrations and interferrometric measurements to correct for the non-flatness of the Spatial Light Modulator. This approach worked reasonably well as long as the projector was well calibrated. Any errors in the assembly process, such as slight misplacement of the lens, caused the variation in the aberration parameters leading to an imperfect correction. Within the framework of Freeman's approach, these errors cannot be easily accounted for, unless they're known

precisely. The approach taken in this thesis is somehow a reverse one: given the projector as a black box, its imperfections are characterized based only on the projected image.

This work attempts to construct a set of methods, forming an automated testbed for holographic projectors. Each model, after exiting the production line is placed on such testbed, where all of its imperfections are characterized. After such callibration, the projector is assigned a set of correction parameters, specific to that particular model. Once callibrated, each projector is able to display highest-quality image throughout its life-span.

Another topic studied is maskless holographic photolitography. The work is a result of a collaboration with Dr Phillip Hands from the University of Edinburgh and LumeJET photo printing company. A number of demonstrator projectors is constructed with intention to develop a cost-effective system for maskless holographic lithography. The projector is later characterized using the developed testbed. Using the supersampled version of Adaptive OSPR with Liu-Taghizadeh optimization, the diffraction limit has been surpassed 2.75 times in each dimension allowing to drastically increase the patterning area. Due to the time constraint, the final goal of patterning features at a sub-micron resolution was not achieved. However, the presented material concludes that the goal is easily within reach, whenever more work is done. This combines approaches of Cable, Buckley, Freeman and Bay in order to achieve both: a wide field-of-view and high pixel-count replay field using inexpensive, off-the-shelf components.

This thesis is finished with a description of work on 3D holographic real-time projection done with a start-up company, Penteract28. It is shown that the 2D hologram in the presence of spatially-varying aberrations is mathematically equivalent to a 3D hologram. Therefore, by applying the same algorithm developed for the purpose of 2D projection, a significant speedup of 3D hologram generation can be achieved.

Table of contents

List of figures

List of tables

List of Acronyms

- AO - Adaptive Optics
- BS - Beamsplitter
- CCD - Charge-Coupled Device
- CUDA - Compute Unified Device Architecture
- DMD - Digital Micromirror Device
- dSLR - digital Single-Lens Reflex Camera
- ff - Fit Function
- FT - Fourier Transform
- GA - Genetic Algorithm
- GP-GPU - General-Purpose Graphics Processing Unit Programming
- HD - Heuristic Descent (A novel correction algorithm)
- HGA - Hybrid Genetic Algorithm (A novel correction algorithm)
- IFTA - Iterative Fourier Transform Algorithms
- LC - Liquid Crystal
- OSPR - One-Step Phase Retrieval Algorithm
- PBS - Polarizing Beamsplitter
- PC-OSPR - Piecewise-Corrected One-Step Phase Retrieval Algorithm (A novel hologram generation algorithm)
- PSF - Point Spread Function

- PWPS - Pixel-to-Wrapped Phase Summation Algorithm
- RPF - Replay Field
- SLM - Spatial Light Modulator

Chapter 1

Introduction

1.1 Holography: Inception and early developments

The concept of holography was first proposed by Dennis Gabor, an Engineer at the British Thomson-Houston Research Laboratories while he was attempting to solve imaging problems in electron microscopy. Gabor saw a great potential that this novel technology offered. Unfortunately, lenses used in the experiment were preventing the desired resolution, because of a severe spherical aberration [1, 2]. However, Gabor reasoned that if it were possible to record the whole image of a sample and translate the same wave field into the optical regime, the aberrations could be easily eliminated by the use of high-quality optical lenses. He described his work in a 1948 Nature paper "A new microscopic principle" [3] together with an experimental verification [3, 4]. That discovery earned Gabor a Nobel Prize in Physics in 1971 [1].

While creating Holography, Gabor combined two great ideas. The first was an interference of an object beam with a coherent background, which was previously described by Fritz Zernike in his invention of a Phase Contrast Microscopy [5]. The second was a two-step process, where the intensity of the light field is first recorded on the photographic film and then re-illuminated with another wavelength of light. This approach was taken from L. W. Bragg who in 1939 conducted research on the X-ray microscope. Very few people know that the same idea was coined nearly 20 years earlier by an extraordinary Polish physicist, Mieczysław Wolfke in a publication "About the possibility of imaging molecular lattices" [6]. Wolfke, however, did not present any experimental verification. The publication caught little interest and was eventually forgotten by the scientific community [7–10].

Initially, Holography attracted a lot of attention, but this slowly faded off, as the problems with the method became evident, namely the presence of a conjugate image, and the absence of sufficiently highly coherent optical sources. The issues were resolved years later after

the construction of the laser in 1960. Subsequent breakthroughs followed quickly, namely the invention of white light holography by Yuri Denisyuk [11], the construction of Off-axis recording geometry by Emmett Leith and Juris Upatnieks [12] in 1962 and the construction of so-called Rainbow hologram by Benton [13], which made viewing in incoherent illumination possible. Holography quickly spread outside physics laboratories and became used in security materials [14] as well as art [15].

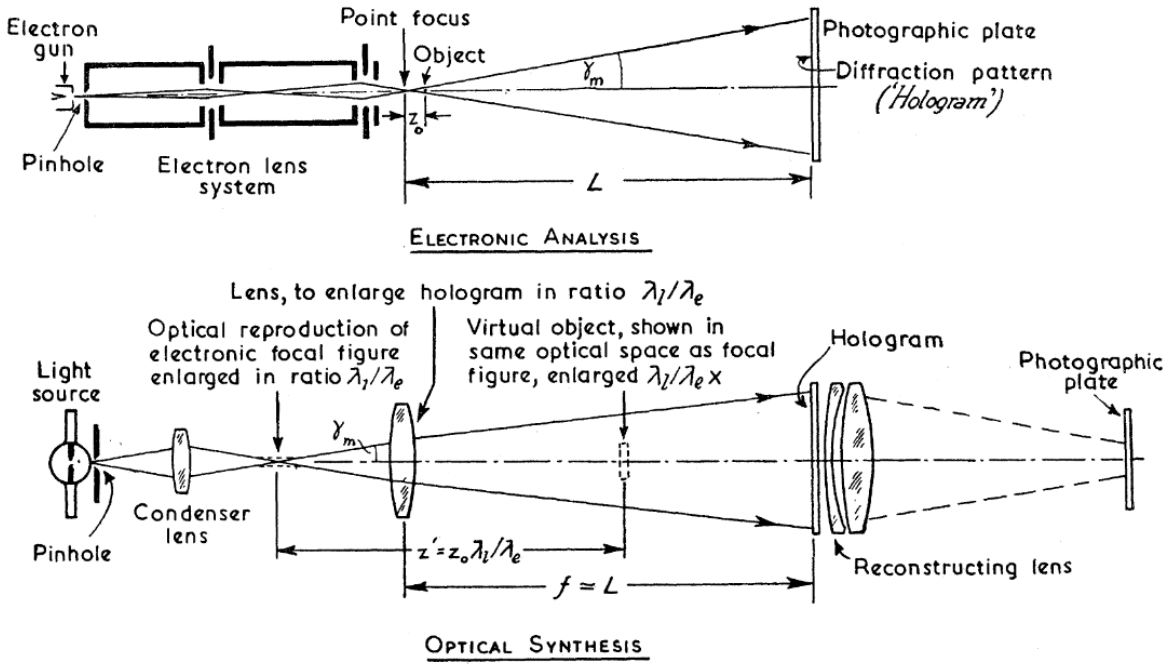
1.2 Holographic Process

The Holographic process is split into two stages: recording and reconstruction (seen in Fig. 1.1a). During the recording phase, a complex object beam gets recorded on the photographic plate by the means of interference with a reference beam. The phase variations of the wavefront, to which the traditional detectors are insensitive, are translated into amplitude variations.

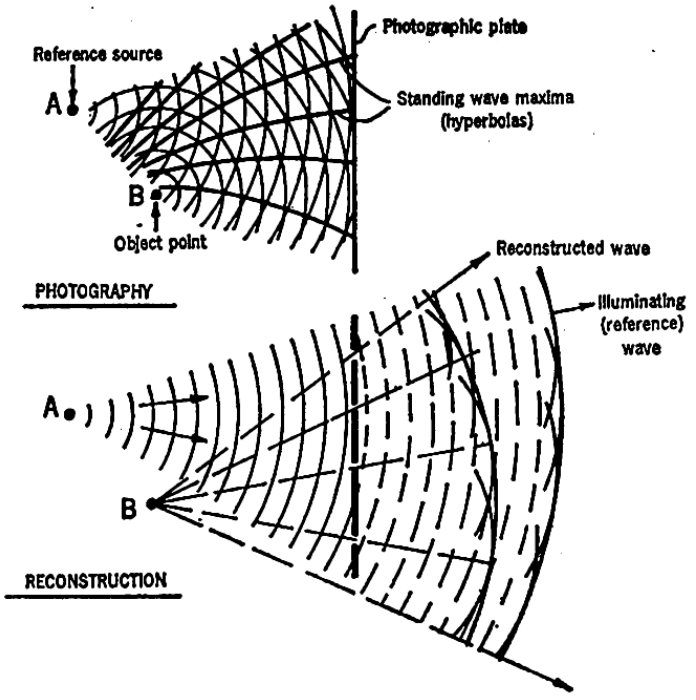
Such an interferrogram is then illuminated with a beam identical to the reference beam. By the virtue of diffraction and interference, one of the wavefronts created is the exact copy of the recording wavefront (as seen in Fig. 1.1b). An extensive mathematical analysis of the process can be found in multiple sources [2, 16–20].

1.3 Digital and Computer-Generated Holography

The holography that Gabor envisioned, was purely an analogue process, using photographic film for both, recording and reconstruction. However, with vast technologic advancements of the recent decade, any of these steps can be digitized [21]. Digital cameras replace photographic films in the recording process. Hologram can then be either reconstructed digitally, or displayed on a Spatial Light Modulator [17]. In particular, holograms of fictitious objects can be synthesized with the aid of a computer [22, 10, 23–25]. The term digital holography is then a very broad definition of employing the principle of light diffraction and interference to acquire, store and/or display data [22]. The first record of a Computer-generated hologram was made by Lohmann and Paris in 1967 [25]. The hologram calculated by them was printed, demagnified optically and recorded on photographic film. Once such a negative is illuminated by a laser, the resultant holographically-projected image appears in the far field pattern. The hologram and the respective replay field can be seen in Fig. 1.2.



(a) Original Setup proposed by Dennis Gabor [4]



(b) Pictorial representation of holographic reconstruction [1]

Fig. 1.1 Original holographic setup and reconstruction

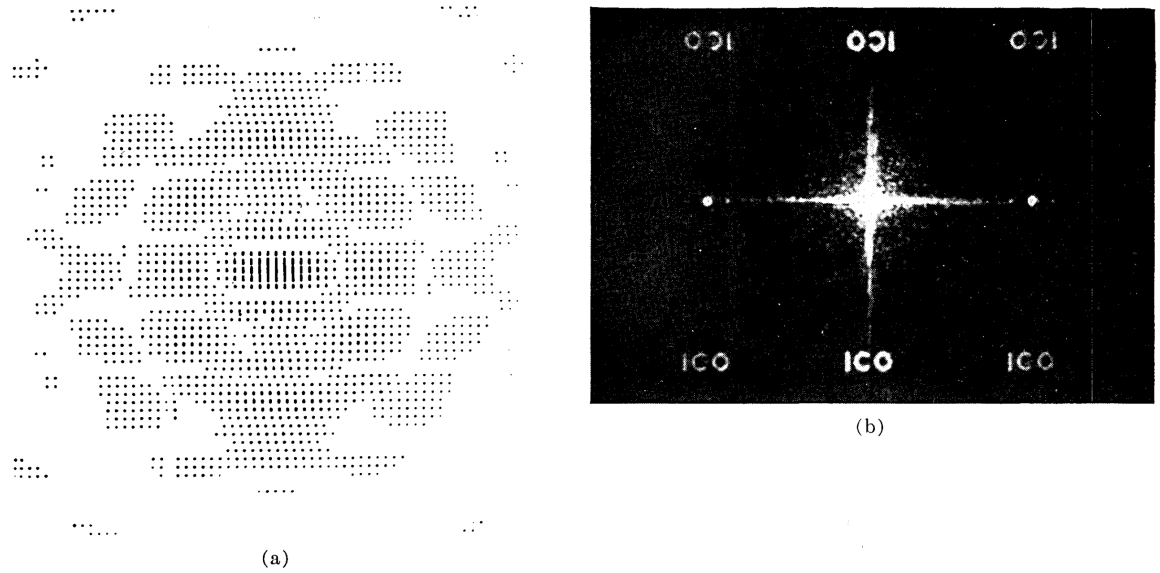


Fig. 1.2 The first hologram synthesized by a computer [25]: (a) synthesized hologram, (b) respective replay field

The next important advancement was the development of the Kinoform, a phase-only hologram. It represents the idea that rather by performing amplitude modulation on the hologram plane, one can perform a phase modulation. Instead of explicitly calculating the interference of two light waves, one can calculate the complex diffraction pattern exactly [26].

Therefore, in this thesis, what we refer to as "hologram" is no longer bound to the physical process of interference with a reference beam. Rather, it is a complex electric field at a particular plane, that, after diffraction leads to the formation of a virtual object behind, or in front of that plane.

1.4 Shortcomings of modern digital holography

It was postulated that holographic projection, because of its unique ability to display real 3-dimensional structures will become the main, "wonder" display technology [27, 28]. Yet, despite multiple attempts of researchers around the world, it has yet to see widespread commercial success. The two fundamental limitations holding the holographic technology back are heavy computational loads while generating holograms and insufficient display hardware [28].

1.4.1 Computational complexity of hologram generation algorithms

Simulating the diffraction of a light wave is a computationally-heavy process. Even approximate solutions require huge computational resources [29]. Several ways to work around this limitation have been reported in literature. A very popular is the Look-up table approach, where certain intermediate heavy computations are pre-computed and stored in memory [30]. This approach trades the computational load for memory usage. A yet another approach [31, 32] relies on approximating the exact formula, based on the viewing conditions. When the range of distances is restricted, the diffraction equations simplify to the Fresnel transform, which can be implemented using number of Fourier Transform operations.

A number of research groups around the world have been working towards both: the algorithmic speed-ups and the optimization speed-ups. Attention should be paid to the work carried out at the Massachusetts Institute of Technology. Back to 1990, they have constructed a custom-made supercomputer termed a Connection Machine CM-2, which for the first time provided real-time hologram generation [33].

With vast advancements on the front of personal computers, the barrier of real-time hologram computation on general-purpose hardware has been surpassed in recent years. Although multiple researchers claimed real-time hologram generation before, the reconstruction quality has been severely degraded by noise, and hence, impractical for high-quality display purposes. The work of Cable and Buckley and the inception of One-Step Phase Retrieval algorithm resolved this problem and led to the construction of a portable holographic projector, Light Touch [34].

1.4.2 Display hardware

The second limitation is the display hardware, namely pixel size, pixel count, and its relationship to the diffraction angle. The situation is much more dramatic in the case of 3D directly-viewed holography than 2D Fourier projecttion.

As shown by Montelongo [35], there exists an inverse relationship between the pixel size and the diffraction angle. Although, one would think it is beneficial to reduce a size of a single pixel, this procedure implies the reduction of the overall size of the display. For instance, assuming the UHD resolution ($4K \times 2Kpx$), a display with pixel size $1.5um$, providing an impressive diffraction angle of 10 degrees would be just $6mm \times 3mm$ in size. On the other hand, if the size is increased to $8cm \times 4cm$ at the same resolution, the diffraction angle is reduced to 0.8 degrees. In order to achieve a display of both the desired size and diffraction angle, the number of pixels need to be increased significantly. To continue with an example of a display $8cm \times 4cm$ in size and a diffraction angle of 10 degrees, that display

would require $50K \times 25K$ pixels (1.25 Gpixels in total). For a number of reasons, such displays are not available on the market. Few workarounds have been proposed and some will be discussed in the subsequent paragraphs.

In the case of 2D Fourier projection, the requirement is far less strict. Since only the far-field pattern, projected onto a screen is observed, the hologram can be demagnified using a telescope in order to increase the diffraction angle. The resolution of the SLM used only defines the number of addressable pixels in the replay field. As proven by Light Blue Optics [36, 34], even an SLMs with the resolution of 1280×1024 can provide a good image reproduction.

A number of research groups constructed workarounds to display high pixel count holograms using low pixel count modulators. Two particular approaches are most popular: tiling systems and scanning systems.

As far as 1989, researchers from the Spatial Imaging Group at the Massachusetts Institute of Technology constructed the MARK-I display based on the one-dimensional acousto-optic modulator [33]. That 1D fringe is then scanned over to form a 2D hologram using an assembly of a polygonal mirror and a galvanometric scanner.

On the other hand, researchers from QinetiQ constructed an active-tiling system, where a single electrically-addressed SLM is replicated onto a bigger optically-addressed SLM [27], therefore drastically increasing the pixel count.

Nonetheless, none of the holographic technologies currently available on the market, including the two mentioned, are easy to mass-manufacture and scale-up [37].

1.5 Thesis Motivation

This thesis attempts to address several problems listed above. First and foremost, it aims to reduce the cost of display hardware by employing cheaper optical components and optimizing the manufacturing costs. Cheap optics as well as imperfect factory assembly inevitably reduces the quality of the image by introducing distortions and aberrations into the image. In this work, we present a variety of methods to characterize and correct these errors. The correction is performed based only on the optical output of the projector. Once calibrated, each device receives a correction information, which is specific to the particular model and allows it to display high-quality image throughout its entire life span.

The methods are tailored for 2D Fourier projectors, but with little more work, can easily be ported into the 3D domain (as outlined in Chapter 7).

The second problem handled is the computational complexity of current 2D hologram generation algorithms. A method previously designed by Freeman was very robust in elimi-

nating aberrations, but slow in operation. By deriving an approximate solution of Freeman's algorithm and combining it with OSPR approach, the algorithm has been speeded up by more than 3 orders of magnitude. Rewriting it using general-purpose graphics processing unit (GP-GPU) computing provided the further speed-up, hence bringing the hologram generation times from several hours to hundreds of milliseconds on a general-purpose GPU device. This provided real-time hologram generation with 2D aberration correction (Chapter 4) and 3D real-time hologram generation (Chapter 7).

All of these methods are tested on a demonstrator wide-angle holographic projector, which has been intentionally misaligned to mimic a realistic factory assembly. The same set of methods and algorithms is then applied to a custom-built projector, designed for maskless photolitographic printing. That shows that very precise image display can be achieved using an imperfect system employing cheap off-the-shelf components.

Solutions to other problems affecting display holography, either postulated or found in literature are listed in Chapter 7.

1.6 Project overview and Thesis organization

Chapter 2 describes in detail the mathematical background, algorithms and methods, serving as a basis for the rest of the thesis.

Chapters 3-5 attempt to establish a set of algorithms for an automated testbed for holographic projectors. Chapter 3 develops an adaptive-optical feedback loop mechanism, designed to characterize aberrations of the holographic projectors. In Chapter 4, common projector imperfections are presented, namely, a spatially-varying aberrations, non-flatness of the SLM, distortions and image intensity attenuation, coming from the pixel shape. A novel algorithm termed Piecewise-Corrected One-Step Phase Retrieval algorithm is developed, incorporating the correction of all these errors. Chapter 5 concludes the previous two by summarizing all of the characterization methods described in Chapter 4 and employs a feedback loop developed in Chapter 3. This concludes the automated testbed, which is used to correct all of the errors, based only on the projector's output.

Chapter 6 explores the topic of maskless holographic lithography. Several demonstrator projectors are constructed with an attempt to establish a cost-effective holographic lithography system. The demonstrators are then characterized using the described testbed. A number of novel techniques are used and expanded on, namely image tiling and breaking of the diffraction limit. Because of a rather short duration of a project, a number of improvements are suggested. Chapter 7 is dedicated to current and future work. It summarises all the

previously discussed work and describes a set of follow-up projects that are currently being researched further.

1.7 Novelty of work

The first important innovative topic discussed throughout this thesis is the automatic correction of common errors in holographic projectors. A set of correction methods for the vast majority of errors found in holographic projectors is designed. Best attempts have been made towards a full automation of the process. This Automated testbed for Holographic Projectors seems to be the most advanced and comprehensive among all the methods found in literature.

The second milestone was the construction of the algorithm, incorporating the correction of all the previously mentioned errors. The algorithm, termed Piecewise-Corrected One-Step Phase Retrieval Algorithm (PC-OSPR) offered a speedup of 3 orders of magnitude, bringing the hologram generation times from the order of days to the order of minutes (using MatLAB implementation). The algorithm was then implemented on a GPU using Highly-Parallel General Purpose GPU computing in nVidia Compute Unified Device Architecture (CUDA), achieving real-time generation at up to 12 fps on a mid-range graphics card. The novel elements here are the construction of the algorithm itself and its optimized implementation.

The third large topic discussed throughout this thesis is the holographic projection for maskless lithography applications. Although multiple authors in the past discussed surpassing the diffraction limit in imaging, the topic of surpassing it in projection was still largely unexplored. Using a variant of Gerchberg-Saxton algorithm with substantial supersampling of the replay field allowed to explicitly break the diffraction limit of the system by a factor of 2.75. Novel contributions here are again, the construction of the algorithm and its efficient implementation.

1.8 Publications

- F. Yang, **A. Kaczorowski**, T. D. Wilkinson (2014), *Fast precalculated triangular mesh algorithm for 3D binary computer-generated holograms*, Applied Optics, vol. 53.
- **A. Kaczorowski**, G. S. D. Gordon, A. Palani, S. Czerniawski, T. D. Wilkinson (2015), *Optimization-based adaptive optical correction for holographic projectors*, Journal of Display Technology, vol. 11.
- F. Yang, **A. Kaczorowski**, T. D. Wilkinson (2016), *Enhancing the quality of reconstructed 3D objects by using point clusters*, Applied Optics, vol. 54.

- **A. Kaczorowski**, G. S. D. Gordon, T. D. Wilkinson (2016), *Adaptive spatially-varying aberration correction for real-time holographic projectors*, Optics Express, vol. 24.

Chapter 2

Mathematical preliminaries

The purpose of this chapter is to introduce the reader to the mathematical notation used in this thesis as well as to the background concepts and algorithms being expanded on. We will begin with a brief discussion of diffraction and different ways of approximating it, then proceed to the description of state-of-the art algorithms and techniques used in digital holography. Finally, we will introduce the topic of aberration correction, Zernike Polynomials, and present algorithms to correct holographically the aberrations.

2.1 Diffraction

While dealing with coherent light, the process of light interference and diffraction is encountered. The extensive analysis of the process and the derivation of the formulae can be found in [16, 19, 20].

We will be focusing on a scalar diffraction theory, where the quantity of interest is the scalar electric field $E(x,y,z)$. Assuming that the field is defined on a particular x-y plane and, in the absence of any sources, we intend to find an accurate description of the same field after travelling a distance z. Let us define the hologram to be a complex scalar electric field, defined on the x-y plane at a distance $z = 0$:

$$H(x,y) \equiv E(x,y,0)$$

while the same electric field, after travelling a distance z is called by the convention, the replay field (RPF) of the hologram:

$$\psi(u,v,z) \equiv E(u,v,z)$$

One of the solutions binding these two quantities, called the Rayleigh-Sommerfeld diffraction formula can be represented as:

$$\psi(u,v,z) = \frac{1}{i\lambda} \iint H(x,y) \frac{e^{ikr_{01}}}{r_{01}} cos\theta \, dx \, dy$$

where λ is the wavelength of light, $k = \frac{2\pi}{\lambda}$ is the wavenumber, r_{01} is the distance between the points (u,v,z) and $(x,y,0)$ and $cos\theta$ is the obliquity factor - an angle between the point of observation and the wave-vector $\vec{k}$. This result can be intuitively understood when pictured a number of point-source emitters located at discrete locations:

- The factor $e^{ikr_{01}}$ reflects the oscillation of the optical phase with distance.
- $\frac{1}{r_{01}}$ is the attenuation of wave's amplitude as it progresses. Energy conservation implies that the product of light intensity with the surface area of the wavefront needs to be conserved. The surface area of the sphere is $S = 4\pi r^2$, therefore intensity falls as $I \propto \frac{1}{r^2}$ with the distance. And since the intensity is the square of the electric field's amplitude, it follows that $E \propto \frac{1}{r}$.
- Obliquity factor $cos\theta$ can be thought of as the spatial cutoff function. It reflects the idea that the central area right behind the radiating source is affected the most, while the area in the same plane, but further away from the centre is affected less. In most real-life situations when small angles are involved, it can be safely approximated as equal to 1.

A number of researchers use this formula directly to calculate the complex field at the hologram plane. It is the most accurate description of the diffraction process, but such computations are lengthy and therefore, not suited for approaches where speed is the concern.

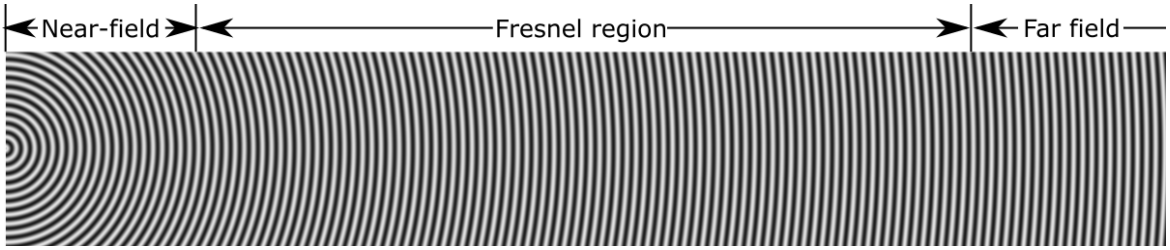


Fig. 2.1 Diffraction regions, depending on the source distance

Depending on the specific conditions, such as the observation distance and angle, it is possible to distinguish 3 regions, as shown in Fig. 2.1:

- **Near-field region**
  When the distance is small enough, the exact formula has to be used.
- **Fresnel region**

If the distance from the source is larger so that the spherical wavefronts can be approximated by parabolas, the Fresnel approximation can be used.

- **Fraunhofer region** - Far field
  If the distance from the source is sufficiently large, the wavefront can be approximated as planar. In this process, all of the depth information that the wave carries is lost and the relationship becomes a simple Fourier Transform.

2.1.1 Fresnel approximation

In order to approximate the exact formula, following assumptions are made:

- Only small angles will be dealt with, so the obliquity factor will be equal to one and $r_{01} \approx z$ in the denominator
- r_{01} in the numerator can be rewritten as $r_{01} = z\sqrt{1+\frac{(x-u)^2}{z^2}+\frac{(y-v)^2}{z^2}}$ and binomially expanded employing the Taylor identity:

$$\sqrt{1+\alpha} = 1+\frac{1}{2}\alpha-\frac{1}{8}\alpha^2+\frac{1}{16}\alpha^3+\dots$$

- Higher-order terms from the expansion can be discarded, corresponding to approximating spherical wavefronts to parabolic ones, giving:

$$\begin{aligned} r_{01} &\approx z+\frac{(x-u)^2}{2z}+\frac{(y-v)^2}{2z} \\ &= z-\frac{xu+yv}{z}+\frac{(x^2+y^2)+(u^2+v^2)}{2z} \end{aligned} \tag{2.1}$$

After some straight-forward rearrangement, one arrives at the Fresnel Diffraction formula expressed using a Fourier Transform operation.

$$\psi(u,v,z) = \frac{e^{ikz}}{i\lambda z}e^{i\frac{\pi}{\lambda z}(u^2+v^2)}\mathscr{F}\left\{H(x,y)e^{i\frac{\pi}{\lambda z}(x^2+y^2)}\right\} \tag{2.2}$$

2.1.2 Fraunhofer approximation - far field region

Assuming the distance z is greater than a certain limit, one arrives at the Faunhofer approximation [16]:

$$\psi(u,v,z) = \frac{\exp\left[ik\left(z+\frac{u^2+v^2}{2z}\right)\right]}{i\lambda z}\mathscr{F}\left\{H(x,y)\right\}$$

This tells us that the image observed infinitely far from the viewer is essentially a Fourier Transform of the input complex hologram. The distance sufficient to observe a far field

pattern can be calculated using the formula [16]:

$$z \gg \frac{\pi(x^2+y^2)_{max}}{\lambda}$$

This formula is troublesome, as it usually gives unreasonably large distances and contains a not very well-defined "≫" relationship. A more reasonable limit is given by the "Antenna designer's formula" [16]:

$$z > \frac{2D^2}{\lambda}$$

where D is the maximum linear dimension of the aperture (hologram).

2.2 Spatial Light Modulation

In computer-generated holography, spatial light modulators (SLMs) are used to display a hologram. Such devices modulate certain properties of the light field, such as amplitude, phase, or some combination of both in a discrete fashion. There exist a number of device architectures, falling in two broad categories: optically addressed and electrically addressed [16]. As the name implies, optically addressed devices modulate the properties of the wavefront depending on the intensity of light falling on the device in the previous pass. The "write" light, as it is called, can have any state of polarization and coherency, while the "read" light, i. e. the light that is being spatially modulated has to be a polarized and coherent [16].

The devices used in this thesis fall within the other category of electrically-addressed SLMs. Among these, there are further subtypes. The most commonly used in holography are acousto-optic modulators, digital micromirror devices (DMDs) and liquid crystal (LC) devices. Acousto-optic modulators are inherently one-dimensional, while DMDs and LCs are two-dimensional.

2.2.1 Liquid Crystal Spatial Light Modulators (LC SLMs)

The devices used in this thesis are liquid crystal devices. The operation of LC SLM is presented here briefly with the emphasis that a curious reader may consult the references. Liquid crystal molecules are rod-like in shape. Whenever a light field encounters such a molecule, it experiences a refractive index which depends on the rotation of the molecule. Whenever an external voltage is applied, molecules position themselves accordingly. The pixelated structure allows to spatially change the orientation of molecules within a single pixel, and therefore, introduce a variable retardance to the light field. This process is pictorially represented in the Fig. 2.2.

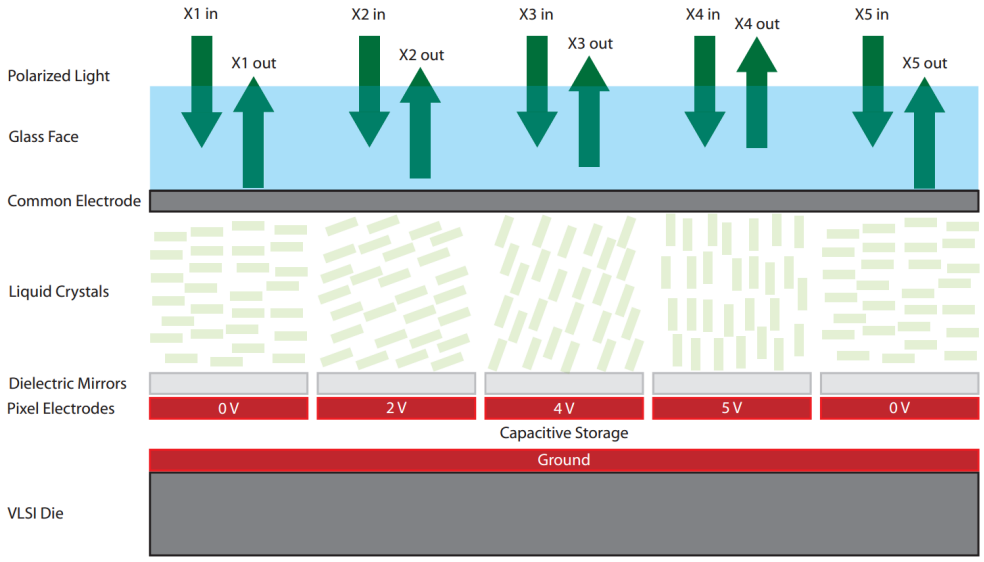


Fig. 2.2 Modulation of the wavefront by the Liquid Crystal device [38]

The modulation of various Liquid Crystal devices can be seen in Fig. 2.3. We will focus on two particular architectures: a continuous phase Nematic Liquid Crystal SLM and a binary phase Ferroelectric Liquid Crystal SLM (also called Smectic C* devices in Fig. 2.3). The nematic devices allow continuous phase modulation, which, in reality means a large, but finite number of quantized states, for instance 256 levels for an 8-bit device. These devices are reasonably slow with response times on the order of milliseconds. Ferroelectric LCs allow only for two possible phase states, but the modulation is relatively fast (tens to hundreds of microseconds).

2.3 Hologram Quantization

A calculated hologram is a complex-valued function. None of the existing SLM architectures are able to display such a vast range of complex values. Therefore, to display such a hologram on an SLM, every hologram pixel needs to be mapped into one of the states that can be represented by the particular SLM architecture. This procedure is called quantization. It degrades the displayed image and leads to quantization noise [40].

One approach to this problem is the following: having the phase freedom in the image plane, coming from the fact that for display purposes, the phase can be chosen arbitrarily, one can try to iteratively change the image phase so that the output hologram better approximates the real SLM modulation. Another approach involves time-multiplexing holographically-projected images in order to temporally average the noise. The two approaches: Iterative

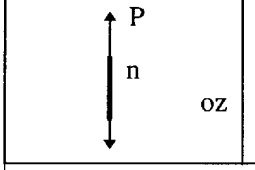
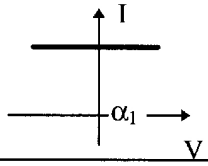
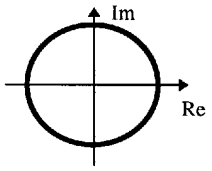
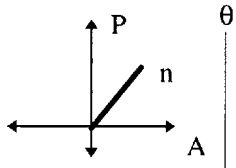
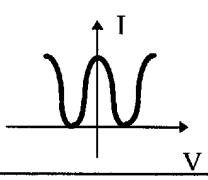
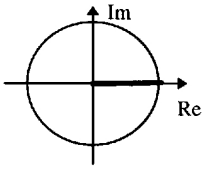
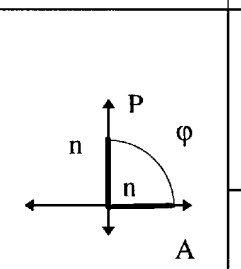
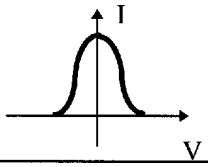
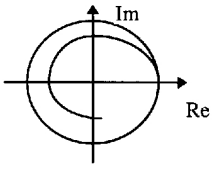
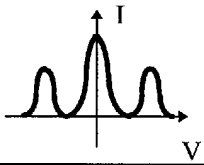
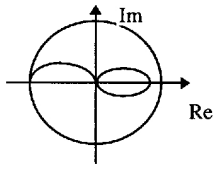
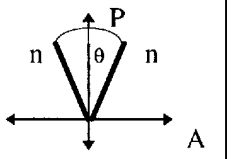
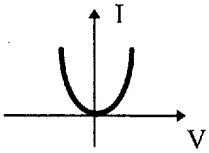
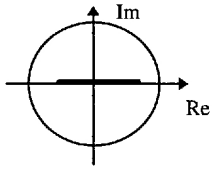
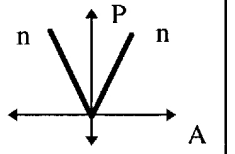
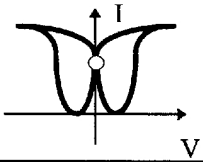
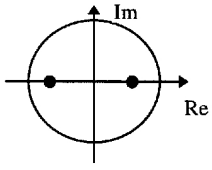
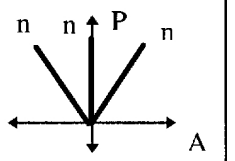
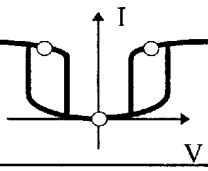
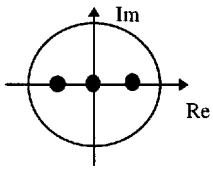
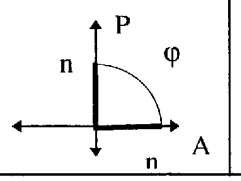
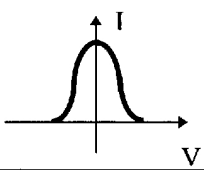
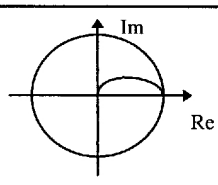
Liquid crystal	Optical configuration	Modulation range	Complex amplitude modulation	Response times
Nematic				τ_{rise} = 10ms (function of d and V) τ_{decay} = 10-100 ms
				
Twisted Nematic (transmission mode)				τ_{rise} = 20ms (function of d and V) τ_{decay} = 20 -50 ms
Twisted Nematic (reflexion mode)				
Liquid crystal	**Optical Configuration**	**Modulation range**	**Complex amplitude modulation**	**Response times**
Smectic A (half-wave plate)				1-100 µs
Smectic C* (half-wave plate)				10-100 µs
Antiferroelectric (half-wave plate)				τ AFLC-FLC : ≈ 10-100µs τ FLC- AFLC : ≈1 ms
Twisted SmC* transmission				τ_{rise} = 10-100 µs (fuction of d and V) τ_{decay} < ms

Fig. 2.3 Modulation schemes of different Liquid Crystal SLMs [39]

Fourier Transform Algorithms and One-Step Phase Retrieval Algorithm are presented subsequently.

2.4 Iterative Fourier Transform Algorithms

Algorithm 1: An example of an Iterative Fourier Transform Algorithm

H : Hologram to be generated
I : Input target image
N : Number of iterations

1 Add uniformly distributed random phase $\vartheta_{rnd}(u,v)$ to the image:

$$\mathsf{T}(u,v) = \sqrt{I(u,v)}\, e^{i2\pi\, \vartheta_{rnd}(u,v)}$$

for $\mathtt{q} \leftarrow 1$ **to** N **do**

2 Perform an inverse Fourier Transform:

$$H(x,y) = \mathscr{F}^{-1}\left\{T(u,v)\right\}$$

3 Quantize the hologram to one of the complex SLM states:

$$H_{quant}(x,y) = \texttt{QuantizeHologram(}H(x,\ y)\texttt{)}$$

4 Perform a forward Fourier Transform:

$$T_{rec}(u,v) = \mathscr{F}\left\{H_{quant}(x,y)\right\}$$

5 Retrieve the phase from the reconstructed image:

$$\vartheta_{rec}(u,v) = \angle T_{rec}(u,v)$$

6 Combine a target image with the retrieved phase:

$$T(u,v) = \sqrt{I(u,v)}\, e^{i2\pi\, \vartheta_{rec}(u,v)}$$

7 **end**

Imaging devices always measure only the amplitude of light and not its phase. Inferring the phase of the wave from only the amplitude information became a well-known problem [41]. Having two intensity measurements of the same light field at different planes, it is possible to infer the phase at both of these planes. This was the main idea of Iterative Fourier Transform Algorithms. The first algorithm of this type was described by Gerchberg and

Saxton in 1972 [42]. Holography researchers soon realized that it can be generalized for any set of constraints and hence, used to construct holograms of an object [43].

The problem in its simplest form is defined as follows: both the image and hologram planes are constrained. Constraint in the image plane is that the square amplitude of the field has to be equal to the target image. While the hologram is constrained by the particular SLM modulation and a fixed illumination profile. Having freedom to choose the phase of the target image, one would like to find such pair: hologram-image that meets the Fourier Transform relationship as close as possible. One implementation of this algorithm is shown in Algorithm 1.

2.5 One-Step Phase Retrieval Algorithm

Cable and Buckley studied the properties of noise in the holographic replay field [40, 44]. Using psychometric tests, they constructed an improved metric for perceived image quality. They concluded that a human visual perception is 25 times more sensitive to the noise variance rather than the noise mean [40]. Therefore, from the point of view of the perceived image quality, it is a lot more effective to display a series of noisy holograms. The independent noise fields will average out due to the central limit theorem, leading to perceived noise reduction. That was a basis for the One-Step Phase Retrieval (OSPR) algorithm [29, 40, 44, 34, 36]. Adaptive OSPR improves the image further by compensating for the error from the previous frames [40]. The flow of this algorithm is presented in Algorithm 2.

Adaptive OSPR reduces the noise as $\frac{1}{N^2}$ as opposed to plain OSPR, which reduces it as $\frac{1}{N}$, where N is the number of time-sequential frames. The speedup gained by using the algorithm offered real-time hologram generation using off-the shelf general-purpose computing resources available at the time.

2.6 Super-resolution algorithms

Cable describing the OSPR algorithm, discussed "the super-resolution algorithm" as one of methods of further improving the quality of the replay field and the pixel count [40]. We will look closely at this particular algorithm and study its operation. It is well-known that increasing the number of sampling points in the replay field increases its resolution. However, there is only a limited number of pixels that the SLM can modulate. Using a variation of the IFTA connected with the increase of the Fourier Transform resolution, breaking of the diffraction limit in the holographic projector can be achieved. While this result seems counter intuitive, it is clear from physical considerations that this effect is possible. Our argument

Algorithm 2: Adaptive One-Step Phase Retrieval Algorithm (AdOSPR)

$I(u,v)$	**:**	Input target image
N	**:**	Number of OSPR frames to generate
$H_{j,quant}(x,y)$	**:**	Output j-th binary hologram
F	**:**	Total visual field

1 Assign $F \leftarrow 0$

2 Calculate replay field scaling factor:

$$S_I = \sum_{u,v} I(u,v)$$

3 Calculate the target field for the first iteration:

$$T(u,v) = \sqrt{I(u,v)}$$

for $j \leftarrow 1$ **to** N **do**

4 Add random phase $\vartheta_{j,rnd}(u,v)$ to the target:

$$T(u,v) = T(u,v)\, e^{i2\pi\, \vartheta_{j,rnd}(u,v)}$$

5 Perform an inverse Fourier Transform:

$$H_j(x,y) = \mathscr{F}^{-1}\left\{T(u,v)\right\}$$

6 Quantize the hologram to one of the complex SLM states:

$$H_{j,quant}(x,y) = \texttt{QuantizeHologram}(H_j(x,y))$$

7 Reconstruct an image by performing a forward Fourier Transform:

$$T_{rec}(u,v) = \mathscr{F}\left\{H_{j,quant}(x,y)\right\}$$

8 Calculate a total reconstructed field:

$$F(u,v) = F(u,v) + \left|T_{rec}(u,v)\right|^2$$

9 Calculate an adaptive compensation for the next frame:

$$T(u,v) = \begin{cases} \sqrt{(j+1)\, I(u,v) - \frac{F(u,v)\, S_I}{\sum F(u,v)}} & \text{if} \quad \frac{F(u,v)\, S_I}{\sum F(u,v)} < (j+1)\, I(u,v) \\ 0 & \text{otherwise} \end{cases}$$

10 **end**

proceeds as follows. The light field is always a continuous function and it is safe to assume that this function is going to be constrained in as many places as there are pixels in the hologram. Performing a Fourier Transform at a native resolution (resolution of the SLM) is the most straight-forward sampling operation. However, when an FT in a higher resolution is performed, the mentioned sampling is much denser. One can imagine the situation, where a number of very dense sampling points is chosen in one region (called from now on the signal region), while the rest of the function is unconstrained and free to fluctuate (called the noise region). This procedure can technically achieve much higher resolution in the signal region than the native FT, whilst not breaking fundamental information theory constraint.

The second argument, presented by Cable is based on inter-pixel interference. Usually, it is an unwanted effect resulting from two point-spread functions overlapping each other. When two PSFs overlap coherently, we begin to see the interference effects. Whereas in incoherent optical systems, this overlap cannot in any way be controlled, in Holography, the pixel's phase as well as amplitude can be adjusted. When a large number of such PSFs are put in a very close proximity with both: amplitude and phase tuned appropriately, it is possible to display structures, which are much smaller than the size of the PSF itself, and hence, break the diffraction limit of the system.

To incorporate super-resolution into the Gerchberg-Saxton type algorithm, we have to revise the constraints that it imposes on the replay field and the hologram:

- The SLM can only represent $M \times N$ pixels out of a large hologram. Pixels that belong to the SLM will have non-zero amplitude equal to the SLM's illumination profile and an appropriately quantized phase state. All the pixels "outside" of the SLM will have the amplitude forced to 0.
- The image intensity will be constrained in a given signal region. Outside of this region, the replay field is unconstrained and allowed to fluctuate.

Using these revised constraints, it is possible to artificially increase the resolution inside a given signal window, at a cost of having no control over the rest of the replay field. This allows to display structures smaller than the diffraction limit of an optical system. This property of the algorithm will be utilized and expanded on in Chapter 6.

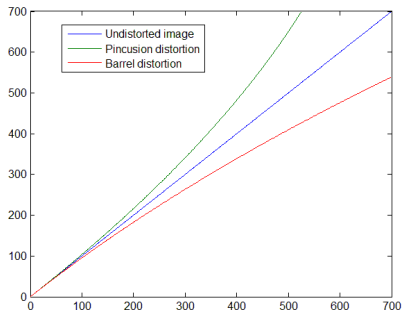
2.7 Aberration correction

In a perfect optical system, the image of an object is its identical reproduction. In reality, this behaviour is prevented by distortions and aberrations.

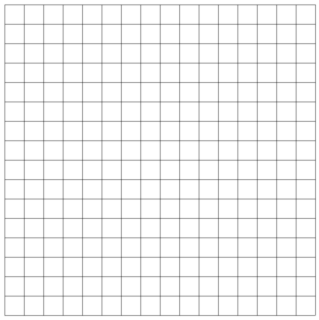
Distortion is the geometric deformation of the image [45–47]. It is a form of optical aberration, but it is often omitted, as it only affects the shape of the image, and not its

sharpness [48]. Strictly speaking, distortion has a linear hologram coordinate dependence (similar to Zernike 1 and 2), but it also has a strong spatial dependence [45, 46].

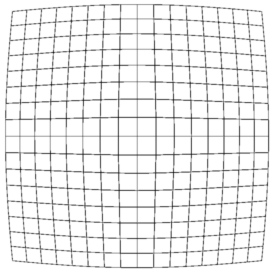
The majority of the optical systems are rotationally symmetric, so is the projector used in this work. For this type of distortion, the dependence of the real (distorted) radius versus paraxial radius is sufficient to characterize and correct it.



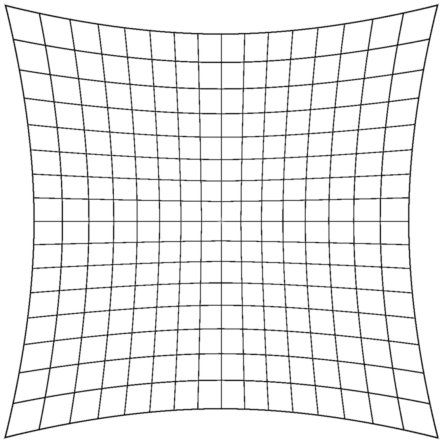
(a) Distortion curve



(b) Grid



(c) Barrel distortion



(d) Pincushion distortion

Fig. 2.4 Distortion

This type of distortion depends only on the distance from the centre of the replay field and not on the angle. The two most common types of distortion are pincushion distortion and barrel distortion. In this research, it is assumed that there is no distortion present in the centre of the field (the curve is tangential to $y = x$ as $x \to 0$). Therefore, by looking at the shape of the curve in Fig. 2.4a , one can intuitively understand the type of distortion dealt with. When the graph curves downwards, the pixels are dragged towards the centre of the

field. This is called barrel distortion (Fig. 2.4c). For the upward curvature, the pixels are dragged outwards and this distortion is called pincushion (Fig. 2.4d).

Aberrations, on the other hand are disturbances in the phase of the wave and are caused by the optical design or imperfections of the optical elements . Aberrations can be imagined as the blur factor introduced to the image. There are two main treatments of aberrations, developed by Seidel, and later by Zernike. For the purpose of this thesis, Zernike polynomials are used to characterize and eliminate the aberrations.

2.7.1 Zernike polynomials

Zernike described an infinite set of 2-dimensional functions used to characterize the aberrations [49]. Zernike polynomials are defined on the unit circle and are represented by their radial and angular parts. They form a complete, orthonormal set, and hence, any function defined on the unit circle can be approximated up to arbitrary precision given enough terms in the expansion. The exact form of Zernike polynomials is [49, 50]:

$$Z_n^m(\rho,\theta) = N_n^m\, V_n^m(\rho)\, G^m(\theta)$$

where n is the order of a polynomial, m is the azimuthal frequency, N_n^m is the normalization constant, $V_n^m(\rho)$ is the radial polynomial, and $G^m(\theta)$ is the angular polynomial, such that:

$$V_n^m(\rho) = \sum_{s=0}^{\frac{n-m}{2}} \frac{(-1)^s\,(n-s)!}{s!\,\left(\frac{n+m}{2}-s\right)!\,\left(\frac{n-m}{2}-s\right)!}\rho^{n-2s}$$

$$G^m(\theta) = \begin{cases} cos(m\theta) & \text{if } m \geq 0 \\ cos(m\theta) & \text{if } m < 0 \end{cases}$$

$$N_n^m = \sqrt{\frac{2(n+1)}{1+\delta_{m0}}}$$

where δ_{m0} is the Kronecker delta symbol.

Single numbering schemes

An expansion using two coefficients m and n is not very convenient to use. Therefore, multiple attempts were made to construct a single-numbering scheme by Noll [51], Wyant and Creath [45], and American National Standards Institute [52, 50, 53]. They differ by the ordering of the terms.

Table 2.1 Zernike polynomials in University of Arizona single-numbering scheme

Index #	n	m	Polynomial		Aberration	
0	0	0	1		Piston	
1	1	1	ρ	$\cos(\theta)$	Tip	
2	1	-1	ρ	$\sin(\theta)$	Tilt	
3	1	0	$2\rho^2-1$		Focus	
4	2	2	ρ^2	$\cos(2\theta)$	Astigmatism	at 0°
5	2	-2	ρ^2	$\sin(2\theta)$		at 90°
6	3	1	$\rho(3\rho^2-2)$	$\cos(\theta)$	Coma	at 0°
7	3	-1	$\rho(3\rho^2-2)$	$\sin(\theta)$		at 45°
8	2	0	$6\rho^4-6\rho^2+1$		Spherical	
9	3	3	ρ^3	$\cos(3\theta)$	Trefoil	at 0°
10	3	-3	ρ^3	$\sin(3\theta)$		at 45°
11	3	2	$\rho^2(4\rho^2-3)$	$\cos(2\theta)$	Secondary Astigmatism	at 0°
12	3	-2	$\rho^2(4\rho^2-3)$	$\sin(2\theta)$		at 90°
13	3	1	$\rho(10\rho^4-12\rho^2+3)$	$\cos(\theta)$	Secondary Coma	at 0°
14	3	-1	$\rho(10\rho^4-12\rho^2+3)$	$\sin(\theta)$		at 45°
15	3	0	$20\rho^6-30\rho^4+12\rho^2-1$		Secondary Spherical	

The scheme followed in this thesis is the one described by Wyant [45] and used in ZEMAX as Zernike Fringe Phase [54]. To avoid any confusion, the explicit form of 15 polynomials together with their corresponding indices is shown in Table 2.1

Once the single numbering is established, an arbitrary wavefront $\varphi(x,y)$ can be approximated as a summation of Zernike polynomial contributions:

$$\varphi(x,y) \approx \sum_{q=0}^{N} a_q Z_q(x,y) \tag{2.3}$$

where a_i is the i-th coefficient in the expansion and Z_i is the i-th Zernike polynomial (according to the numbering scheme chosen). By varying the total number of terms used (N), one can approximate the function $\varphi(x,y)$ up to arbitrary precision.

2.7.2 Holographic aberration correction

In the most general form, the aberrations of the optical system can depend on both the image coordinates as well as the hologram (aperture) coordinates [55, 46]. The convention is to call (x,y) the hologram (aperture) coordinates and (u,v) the image (field) coordinates.

The simplest to correct is the field-independent aberration. This type depends only on the hologram coordinates, and not on the image coordinates. On the other hand, spatially varying aberration (also called field dependent aberration) depend not only on the aperture coordinates, but also differ spatially over the image plane.

2.7.3 Field-independent aberration correction

Cable argues that the aberrated replay field of the hologram can be represented as [40]:

$$\psi(u,v) = \mathscr{F}\left\{B(x,y)\,H(x,y)\,e^{2\pi i\,\varphi(x,y)}\right\} \tag{2.4}$$

where $H(x,y)$ is the hologram, $B(x,y)$ is the SLM illumination profile (usually a Gaussian beam), and $\varphi(x,y)$ is the phase profile (aberration). It is easy enough to see that if an identical phase mask with an opposite sign is applied to the hologram before the quantization step, the aberrations present can be precisely eliminated:

$$\begin{aligned}\psi(u,v) &= \mathscr{F}\left\{B(x,y)\,\left[H_{uncorr}(x,y)\,e^{-2\pi i\,\varphi(x,y)}\right]\,e^{2\pi i\,\varphi(x,y)}\right\}\\ &= \mathscr{F}\left\{B(x,y)\,H_{uncorr}(x,y)\right\}\end{aligned} \tag{2.5}$$

2.7.4 Spatially-varying (field-dependent) aberrations

For the projector discussed in this work, such treatment is insufficient, because these aberrations are severe enough to have a strong spatial variation. To illustrate this phenomenon, let us consider the previous equation with a generic phase mask dependent on image as well as hologram coordinates $\varphi(x,y,u,v)$ and let us assume that the correction is made at a position (u_0,v_0) in the replay field:

$$\begin{aligned}\psi(u,v) &= \mathscr{F}\left\{B(x,y)\,\left[H_{uncorr}(x,y)\,e^{-i2\pi\,\varphi(x,y,u_0,v_0)}\right]\,e^{i2\pi\,\varphi(x,y,u,v)}\right\}\\ &= \mathscr{F}\left\{B(x,y)\,H_{uncorr}(x,y)\,e^{i2\pi\left(\varphi(x,y,u,v)-\varphi(x,y,u_0,v_0)\right)}\right\}\end{aligned} \tag{2.6}$$

It can be seen that the aberrations will be precisely eliminated only in the point (u_0,v_0), but as the spatial coordinates deviate from it, the given correction will, in general, be insufficient. Freeman [56, 57] dealt with this problem by replacing a Fourier Transform operation with a more basic summation of contributions coming from all the pixels. Once single pixel contributions are separated, the spatially-varying dependence can be superimposed on the top of it.

2.7.5 Pixel-to-Wrapped Phase Summation (PWPS) Algorithm

The way the hologram is usually generated is by applying a Fourier transform operation. Because a Fourier transform processes all of the pixels at once, it is difficult to impose a spatially-varying correction on it. Jonathan Freeman handled this problem, by decomposing a Fourier transform into a set of more basic operations, acting on single pixels [56, 57].

To illustrate the development of this algorithm, one can consider a single point in the replay field. The hologram needed to display such structure is a grating, or thinking more generally, a continuous phase surface. The orientation of such surface defines the position of the pixel. Having a pixel at a position (u,v) and the replay field of size (u_{max}, v_{max}), the continuous phase surface corresponding to a point with a phase $\vartheta(u,v)$ will be:

$$\Phi_{uv}(x,y) = \left(\frac{u}{u_{max}} x + \frac{v}{v_{max}} x + \vartheta(u,v) \right) \tag{2.7}$$

The hologram, corresponding to this single pixel would then be represented as:

$$H_{uv}(x,y) = \sqrt{A(u,v)}\, e^{i2\pi\Phi_{uv}(x,y)}$$

$A(u,v)$ being an amplitude of that pixel.

Since an image can be thought of as a summation of single pixels, one is allowed to write:

$$I = \sum A(u,v)$$

Since FT is additive, it is then straight-forward to realize that the hologram of an entire image will then become a weighted sum of these pixel contributions:

$$\begin{aligned} H(x,y) &= \sum H_{uv}(x,y) \\ &= \sum \sqrt{A(u,v)}\, e^{i2\pi\Phi(x,y)} \end{aligned} \tag{2.8}$$

Up till this point, a hologram generated does not include aberration correction, and therefore, the above formula 2.8 is equivalent to a Discrete Fourier Transform. The approach to generating hologram is, however, slightly different, as it separates contributions, coming from single pixels. In order to add a spatially-varying aberration correction, one needs to modify the pixel phase accordingly:

$$\begin{aligned} H(x,y) &= \sum H_{uv}(x,y)\, e^{-i2\pi\varphi(x,y,u,v)} \\ &= \sum \sqrt{A(u,v)}\, e^{i2\pi\left(\Phi(x,y)-\varphi(x,y,u,v)\right)} \end{aligned} \tag{2.9}$$

Storing all the possible values of $\varphi(x,y,u,v)$ would be highly inefficient, therefore Freeman used the Zernike expansion in a following manner:

$$\varphi(x,y,u,v) \approx \sum_{q=0}^{N} a_q(u,v) \times Z_q(x,y)$$

Now, the expansion coefficients $a_i(u,v)$ are 2-dimensional arrays that indicate how much of each Zernike aberration exists at a particular (u,v) location. These arrays were termed Zernike maps.

Using this method, Freeman was able to fully correct the aberrations of a wide-angle holographic projector.

Distortion correction within PWPS

Distortion is a geometric deformation of the image. For the purpose of this work, only radial distortion is considered (as most of the real optical systems are rotationally-symmetric). PWPS allows its straight-forward correction in two equivalent ways. The calculation of the continuous phase surface [Eq. 2.7] requires (u,v) pixel positions and a correcting phase mask $\varphi(x,y,u,v)$. The distortion correction can either be achieved by directly modifying the values of (u,v) to counteract distortion or in the form of adding the first and second Zernike polynomials into the wavefront. Given the dependence of real (distorted) radius vs. paraxial radius $r'(r)$ The first method can be rewritten as:

$$r = \sqrt{u^2+v^2}$$
$$u_{dist} = r'(r)\,\frac{u}{r}$$
$$v_{dist} = r'(r)\,\frac{v}{r}$$

The output (u_{dist}, v_{dist}) can then be used instead of (u,v) in the continuous phase calculation. It should be emphasized here that this procedure is, in practice, a non-uniform sampling operation. The image pixels are not any more placed on a regular grid, but instead, the grid is skewed in the way inversely proportional to the distortion of the lens.

2.8 Conclusions

The algorithms presented in this chapter serve as a starting point for this thesis.

It can be noted that the holographic image projections, as opposed to the refractive projection systems, allow to use a variety of techniques to improve the quality and the resolution of the image, namely the aberration correction, diffraction image breaking. A realistic reproduction of 3D images is also possible. All of these techniques will be expanded on and combined in following chapters.

Chapter 3

Adaptive-optical feedback loop mechanisms

Considering the further development of holographic technologies, a way to make holographic projectors appealing for general public is to further the miniaturization, while reducing costs and preserving the exceptional image quality. Obviously, there is always a trade-off, high-quality components improve the image, but at the same time increase the cost [40]. Lenses with high numerical aperture tend to be large and bulky.

Optical systems are usually fragile, where precision is crucial to the performance. Even a slight misplacement of a single component can result in the degradation of the optical quality [58, 59]. A precise, automated assembly process preserves the supreme image quality, but also increases the overall cost per device.

The discussion on aberration correction capabilities of holographically projected images proves that holography is fundamentally different from incoherent imaging systems. Due to a precise control over the wavefront's amplitude and phase, one can use mechanisms to directly remove the errors introduced by the imperfect optics as well as an imperfect assembly process. This powerful property will be explored in greater details in this and following chapters.

To prove the principle of aberration correction, Freeman constructed a wide-angle holographic projector [56, 57]. The system contains one particularly inexpensive sapphire ball lens. It is a ball of glass, 6mm in diameter and, due to its shape, it introduces a substantial amount of distortion and aberration into the projected image. Freeman performed aberration correction of this projector, by ray-tracing simulations using ZEMAX package.

There are certain errors which the ray-tracing simulations cannot account for, namely the non-flatness of the spatial light modulator used and manufacturing errors, such as a

misplacement of the optical component. The first problem can be handled by interferrometric measurements. This approach is again troublesome, as it requires a separate experiment.

In this work, a different approach is proposed. Instead of optimizing the optical design and perfecting the assembly process, the errors introduced at each of these stages can be characterized and corrected following the assembly process. The benefit of this approach is simple. The algorithm, relying only on the output of the projector, is agnostic to the source of the errors. Using aberrations as an example, in general, they are being influenced by the SLM used, imperfect optics as well as imprecise alignment. The algorithm, which looks only at the projected image, is not concerned about the source of each particular wavefront error, but characterizes and corrects their total contribution.

This chapter forms a coherent whole with the two following chapters. Here, the general idea of an adaptive-optical feedback loop mechanism is introduced and the implementation of such a system is demonstrated. In the next chapter, correction of various errors of holographic projectors are discussed (distortion, aberration and intensity error) and a novel algorithm, incorporating all of these corrections is developed. The following chapter concludes the previous two by demonstrating the characterization and correction of all the aforementioned errors and showing, how such correction information can be obtained.

3.1 Introduction to Adaptive Optics

Historically, the field of adaptive optics has been developed to a great extent within the area of astronomy [60]. It is not the intention to cover the topic of astronomical adaptive optics, therefore, the introduction will be brief. However, the interested reader can consult the references [60, 61, 55, 62, 63].

The most commonly addressed issue is the problem of atmospheric turbulence. A typical adaptive optical setup can be seen in Fig. 3.1. The telescope, located on the ground acquires an image of a celestial object. Atmospheric turbulence introduces an error to the wavefront, hence degrading the image quality.The adaptive-optical system can account for this error, by directly correcting the wavefront.

The wave is first split into two parts, one of which gets recorded on the camera and the other is directed towards a wavefront sensor, where the wavefront is characterized. The feedback algorithm then outputs a wavefront correction to the modulating device. Once the aberrations are eliminated, a sharper image of an object is recorded on the camera.

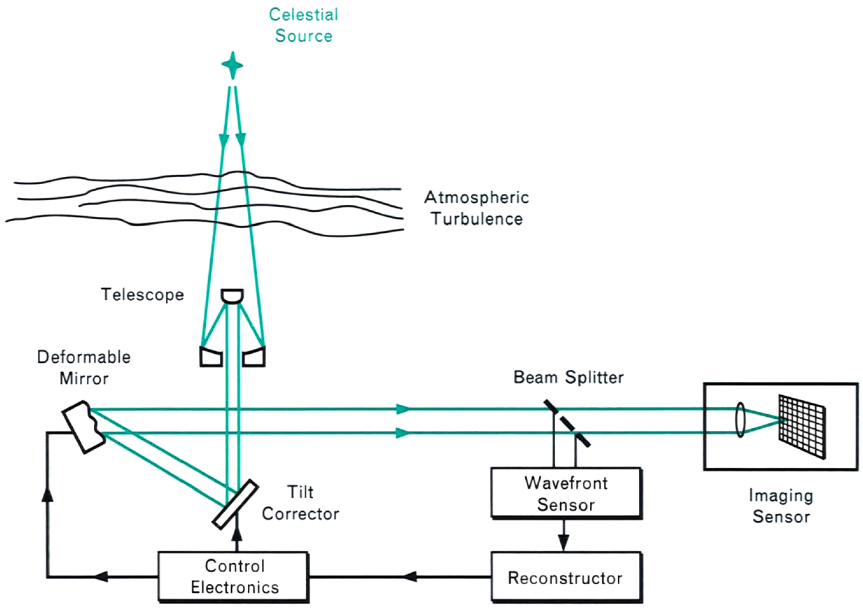


Fig. 3.1 Typical adaptive-optical setup implemented in astronomy [61]

3.1.1 Wavefront Sensors

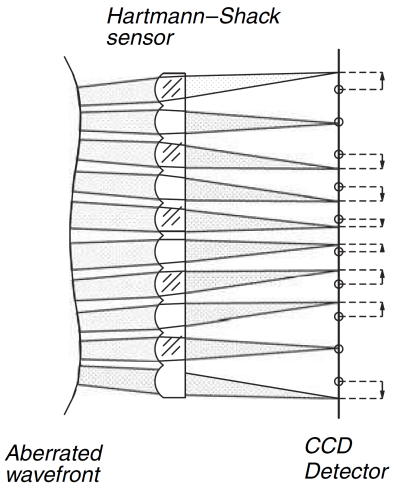
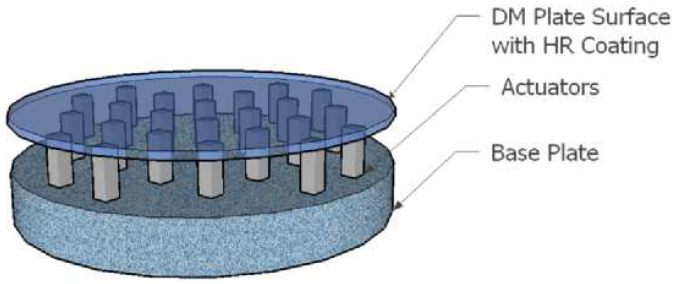


Fig. 3.2 A Shack-Hartman wavefront sensor [55]

The most popular wavefront sensor is the Shack-Hartman sensor. A schematic view can be seen in Fig. 3.2. It works by splitting a wavefront into a set of segments. After each segment, a microlens is inserted, which focuses the particular part of a wavefront onto a spot. The deviation of this spot's position from the centre then defines the slope of the wavefront at a particular segment.

3.1.2 Wavefront Modulators

Deformable Mirror



(a) An example of a deformable mirror architecture [62]

(b) A plate deformable mirror [62]

Fig. 3.3 Deformable mirror

In adaptive optics for astronomy, the most popular type of wavefront modulator is the deformable mirror (DM). It shapes a wave in a continuous fashion using a number of actuators [Fig. 3.3a], which can be adjusted to arbitrary heights. The flexible mirror [Fig. 3.3b], behind the actuators changes its curvature depending on the voltages applied to the electrodes. When a wave reflects off that non-flat surface, its phase changes accordingly. Since optical aberrations are phase disturbances, a change in phase is sufficient to eliminate them. The simulation of the DM operation can be seen in Fig. 3.4

Liquid Crystal and Digital Micromirror Device Spatial Light Modulators

Liquid Crystal (LC) and Digital Micromirror Devices (DMD) Spatial Light Modulators (SLMs) have a similar architecture. They are divided into a set of discrete square pixels. In the case of a LC SLM, the modulation occurs due to the birefringence property of a liquid crystal. In the case of a DMD SLM, parts of the wave get deflected by a mirror,

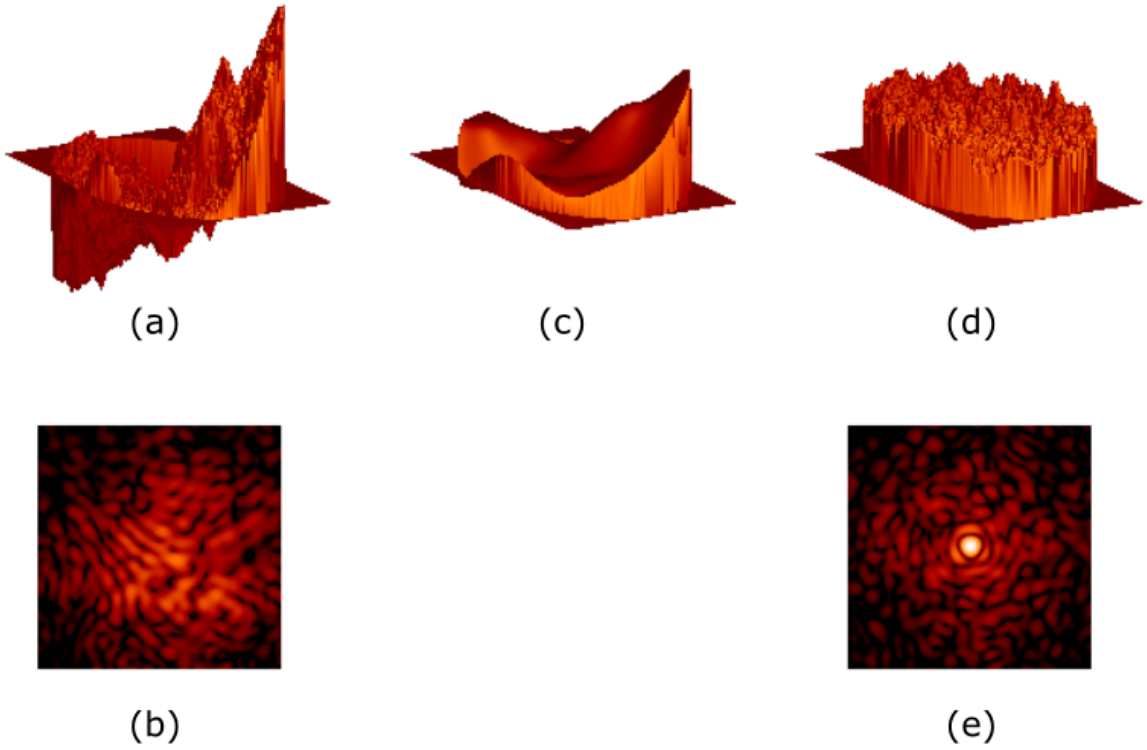


Fig. 3.4 Simulation of a feedback loop operation [63]:
(a) the wavefront, affected by turbulence, (b) a point-spread function corresponding to such aberrated wavefront, (c) shape of the deformable mirror defined by the feedback algorithm, (d) a corrected wavefront, (e) the PSF after the correction

hence corresponding to a binary amplitude modulation. Both of these devices have been successfully used to correct aberrations in holography.

3.2 Difference between Astronomy AO and Holography AO

The typical AO system used in astronomy, differs significantly from the mechanism developed for the purpose of this thesis. Below, we list the differences between these two system and elaborate how they influence the design.

The aberrations corrected are constant in time, but can be spatially-varying

Atmospheric turbulence changes dynamically, and hence requires constant correction. The aberrations in holographic projection depend only on the optical components of the projector, and hence are time-independent, since the layout of a projector does not change. It means

that, once corrected, the same correction parameters can be applied to any image generated by a given projector thereafter.

Aberrations come from imperfect optics

The majority of the aberrations in a holographic projector come from the imperfections in the optics used (the SLM and the lenses). It would be increasingly difficult to insert a wavefront-sensor into an existing architecture of such a projector. Therefore, the simplest way of characterizing the aberrations is by blind, sensorless correction based on the far-field image.

SLMs are used for image formation as well as aberration correction

SLMs in holography are used not only to correct aberrations, but most importantly, to display a hologram. Aberration correction is therefore linked to image display in a fundamental way. This property is an advantage, rather than a disadvantage, because the methods developed in this work can straight-forwardly be applied to any holographic projector, provided that it employs the electrically-addressed SLM (either LC or DMD) as a display medium.

For all of the reasons discussed above, holographic adaptive optical correction will use a spatial light modulator for image display and aberration correction as well as the far-field camera in a sensorless AO optimization [64]. The details of such system, which have been implemented are given below.

3.3 Brief Overview of The Feedback Loop mechanism

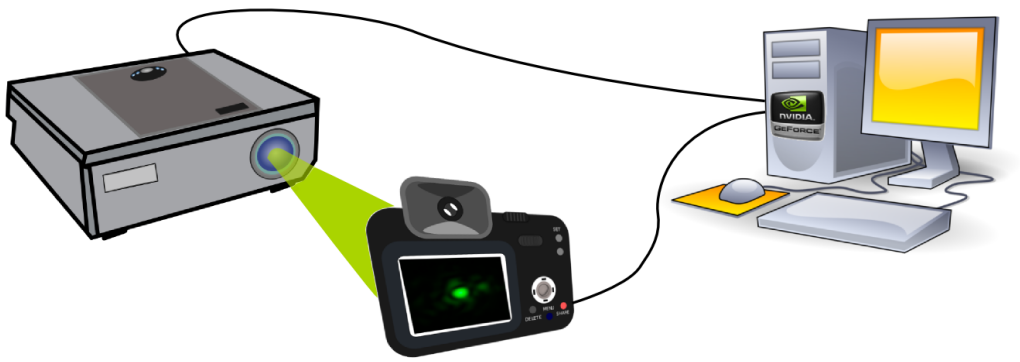


Fig. 3.5 A pictorial explanation of the adaptive-optical feedback loop implemented. A projector displays an image of a single spot directly onto a camera's sensor, which is then fed back to the computer.

An adaptive-optical feedback loop system can be simplistically represented as in Fig. 3.5. A projector being characterized displays an image of a single spot in the replay field ,which is then captured by a feedback device, such as a CCD/CMOS sensor. Because of the aberrations included by the projector's optics, instead of a well-defined spot, a blurry undefined shape is observed. That imperfect shape is characterized by the system and has a fitness metric assigned to it. By applying various corrections to the hologram and measuring its fitness, one iteratively arrives at a better estimate of correction. In the following paragraphs, a detailed description of the process and operation of all the elements is given.

3.4 Hologram Generation

The first ingredient in the aberration correction process is the hologram generation. The easiest method to assess the amount of aberration in any given image is by looking at the most basic shape - a single point. Different shapes, such as crosses and squares have also been investigated [65]. For the purpose of visual inspection, these shapes prove better, but it is far more difficult to construct an error-resistant algorithm assessing their properties.

In order to generate a single point hologram, a simplified version of pixel-to-wrapped phase summation method (Eq. 2.9) is employed. Given a set of Zernike coefficients $a_1 \dots a_{15}$, the aberration-correcting phase mask is calculated according to Eq. 2.3:

$$\varphi(x,y) = \sum_{q=0}^{15} a_q Z_q(x,y) \tag{3.1}$$

This mask is then overlapped with the continuous phase surface:

$$H(x,y) = \exp\left[i2\pi \left(\left(\frac{u}{u_{max}} x + \frac{v}{v_{max}} x + \vartheta_{uv} \right) - \varphi(x,y) \right) \right]$$

For the purpose of binary quantization, this treatment can be simplified further by noticing that only the real part of the wave is necessary to calculate a binary hologram:

$$H(x,y) = \begin{cases} 1 & \text{if } \cos\left[2\pi \left(\left(\frac{u}{u_{max}} x + \frac{v}{v_{max}} x + \vartheta_{uv} \right) - \varphi(x,y) \right) \right] > 0 \\ -1 & \text{otherwise} \end{cases}$$

This procedure is repeated multiple times with different ϑ_{uv} for the purpose of noise elimination. In this manner, an aberration-corrected hologram corresponding to correction parameters $(a_1, a_2, a_3, \ldots, a_{15})$ is generated.

3.5 Fitness function

A crucial element in the system is a suitable fitness function. The desirable function will assign small values to points that are reasonably 'good', i.e. leading to a sharp replay field image, and larger values otherwise.

One metric, characterizing the quality of the point is its **physical size**. It can be intuitively understood how it impacts the replay field when imagining a displayed image as a summation of single point contributions. Two neighbouring image pixels, displayed through a real optical system, are convolved with a point-spread function (PSF) [16, 17, 66]. When the physical size of the PSF is large, the neighbouring pixels will overlap, leading to a blurry image. Therefore, the appropriate size of the PSF should be small enough, so that the overlap, and hence the blur is minimized.

However, the sheer size of the PSF can be ambiguous to the computer program. An example of a false positive PSF can be seen in Figure 3.6. Here, the point is heavily aberrated in such a way that most of the light is distributed in the outer region of the image and is mistakenly perceived by the computer algorithm as noise. That is precisely the reason why we need another metric included in the fit function: **peak intensity** of the point.

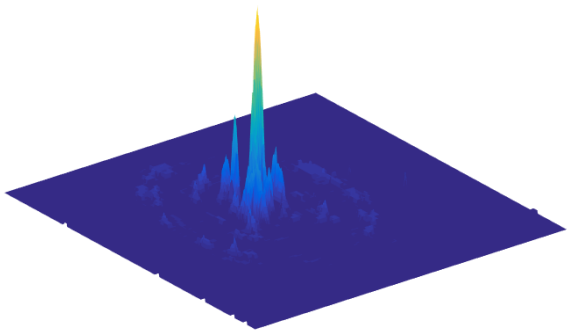


Fig. 3.6 A false positive PSF

These two metrics are interdependent on each other: when the physical size of a point is big, the light is distributed over a larger area and the peak intensity of the point decreases [67].

Some previous approaches use only a peak intensity of the point. However, because of the limited bit-depth of a webcam (8 bits), it would be increasingly difficult to properly differentiate between different points. To increase the sensitivity, a photodiode can be used. However, tip and tilt aberrations can shift the pattern and hence, lead to erroneous results.

Mathematical representation of a Fitness Function

In the following discussion, it is assumed that the image is captured in RGB format. Therefore, what we refer to as the particular channel of the image is in reality the intensity of one colour component. For instance "I_{green}" indicates the intensity of pixels corresponding to green colour.

The fitness function developed in this research can be parametrized in the following manner:

$$FF = w_{blue} \times w_{green} \times \sigma / q$$

where w_{blue} and w_{green} are metrics associated with the peak intensity of the point, σ is the metric associated with its size and q is the normalization factor.

A green laser is used in this research (532nm), because the green channel of the image often contains the most useful information. The peak intensity contribution to the fit function is then defined as:

$$w_{green} = [256 - max(I_{green})]^2$$

where I_{green} is the intensity of the green channel of the image. This function is equal to 1 for the point having highest possible intensity (255 for an 8-bit device) and grows quadratically otherwise. This quadratic growth was introduced to assign a significantly better fitness value to points having much higher peak intensities.

When the intensity falling on the sensor is too high to be fully recorded in the green channel, large values can also be seen in other channels. To assign smaller fit function values to highly saturated images, we introduced an additional factor based on the other colour. Blue has been chosen arbitrarily. A contribution w_{blue} is introduced such that:

$$w_{blue} = \begin{cases} 1 & \text{if } max(I_{blue}) > 150 \\ 100 & \text{if } max(I_{blue}) \leq 150 \end{cases}$$

This way, when overexposure is seen in the blue channel, the fitness function is not altered. Otherwise, it's multiplied by a factor of 100. A value of 100 was again, determined experimentally.

The third component is the spread of the pixels around the centre σ. To decide, what area belongs to the point, a simple thresholding method was implemented. A pixel, having an intensity greater than a certain fraction of the maximum intensity is regarded as being a part of the shape:

$$I_{thres}(x,y) = I_{green}(x,y) > th \times max(I_{green})$$

The threshold value was experimentally determined to be 0.3.

In order to get the size of a point as well as the distribution of pixels around the centre, the centre of the pattern needs to be calculated. A few methods to do so were considered, however, the most robust one appeared to be the summation of pixel intensities along x- and y-direction:

$$SX(x) = \sum_{0<y<y_{max}} I_{green}(x,y)$$
$$SY(y) = \sum_{0<x<x_{max}} I_{green}(x,y)$$

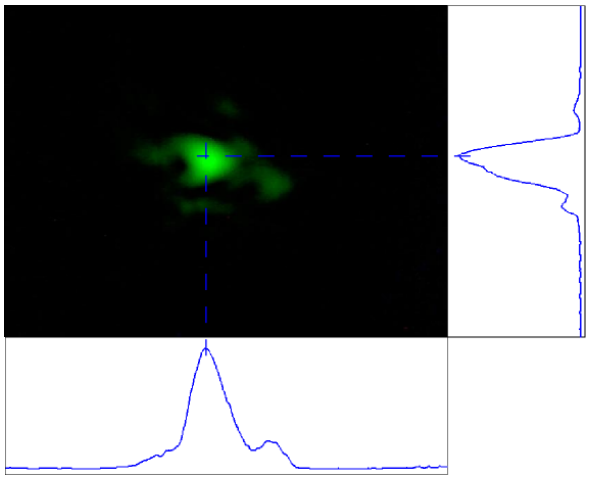


Fig. 3.7 Graphical illustration of finding the approximate centre of the pattern

In order then to find the approximate 'centre' of the pattern, one then needs to find the position of the maximum of the functions $SX(x)$ and $SY(y)$, corresponding to the x- and y-coordinates of the centre points as illustrated in Figure 3.7. This methods proves to work well for points being well-corrected and a bit worse otherwise. However, precision is not really necessary in the case of poorly-corrected points, as they will most likely be discarded in the selection process.

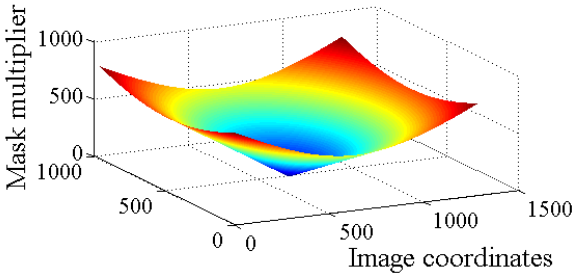


Fig. 3.8 Mask used for finding the spread of points around the centre

Once the centre is established, we can calculate the distribution of points around the centre by multiplying the thresholded pattern with a rotationally symmetric mask:

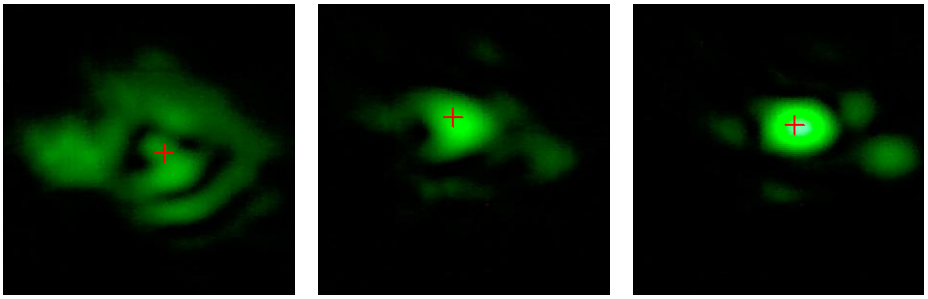
$$mask(x,y) = \sqrt{(x-x_{cent})^2 + (y-y_{cent})^2}$$

And then sum the overall contribution:

$$\sigma = \sum_{x=0}^{x_{max}} \sum_{y=0}^{y_{max}} I_{thres}(x,y)\, mask(x,y)$$

This way, the points closest to the centre will contribute much less to the overall sum than the outliers. And since the mask (seen in Fig. 3.8) is cylindrically symmetric, a round shape of points will be preferred.

To test the fitness function, we can examine three selected patterns seen in Figure 3.9. We can see that indeed we see the correlation between how visually well the point looks and the fitness values. A smeared point of low intensity in Figure 3.9a has a high fit function value of 4.7×10^7, a slightly better point in Fig. 3.9b has a much lower fitness value of 1.5×10^3, and finally, a well-corrected point in Figure 3.9c has a fitness value of 3.2. Also, the central crosses in figures 3.9a, 3.9b, and 3.9c show the position of the centre of the pattern, as assigned by the algorithm.



(a) A bad point $FF = 4.7 \times 10^7$

(b) An average point $FF = 1.5 \times 10^3$

(c) A good point $FF = 3.2$

Fig. 3.9 Patterns used to examine the fit function

Two important points should be made here to understand the concept of the fit function mechanism designed in this research. First, there is a number of different parameters, such as thresholds and multipliers. These parameters have been experimentally fine-tuned until good results were achieved. It is possible that, when more time is spent on optimization, a better fit function can be constructed. However, the mechanism presented here proved satisfactory to achieve correction in each case.

It is also important to realize that the fit function is a relative measure. Its value naturally depends on the experimental conditions, such as: brightness of the laser, distance from the front lens to the camera, and the camera exposure setting. It is however certain that for different conditions, the relations between the fitness of different points are preserved, i.e. a perfectly corrected point will have a smallest fit function value. For this particular reason, the position of the camera should not be moved in between the measurements.

3.6 Correction Algorithms

A hologram is generated using a particular combination of Zernike coefficients $(a_3, a_4, a_5, \ldots, a_{15})$, displayed on an SLM and a picture of the replay field is taken. The picture is then passed to the fit function and the fitness value is assigned. This procedure can be imagined as mapping the 13-dimensional vector of Zernike coefficients to a fitness value: $(a_3, a_4, a_5, \ldots, a_{15}) \mapsto fitness$. The task is now to construct such an algorithm that finds a global minimum of this function with a smallest number of samples. We will here present two families of algorithms used to handle such problems: a steepest-descent algorithm and a genetic algorithm. Both will be discussed and compared. Then a hybrid algorithm combining the two will be presented. The hybrid algorithm is constructed such that convergence time is reduced.

3.6.1 Steepest-Descent algorithm

The method of steepest descent is a well-known technique used in optimization [68]. It travels through the solution space by calculating the gradient of the function and stepping towards the greatest value of the gradient. This method has been applied to some aberration-correction problems [69], however, the fit function described here is discontinuous in places where the quantized intensity of light changes. This limitation was surpassed by constructing another algorithm, that avoids computing the gradient. Instead, it tests each of the 13 Zernike parameters while keeping others constant and then steps towards the greatest change. This algorithm has been termed the heuristic steepest descent (HD) [67]. Assuming the variable $aIn(a_3, \ldots, a_{15})$ is the current position of the algorithm, and $aOut(a_3, \ldots, a_{15})$ are the coefficients of the hologram to test:

Each iteration of this algorithm tests 9 values in each of the 13 directions sampled. Each of these correction parameters has its fit function value calculated. The next position of the algorithm then becomes the one among all the candidates that minimizes the fit function. The procedure is then repeated until the fit function doesn't improve any more.

This procedure was found to work reasonably well, but it tended to get stuck in local minima (a point, which is not a perfect correction, but has the fit function better than all the points around it).

3.6.2 Genetic Algorithm

Genetic algorithms are inspired by the mechanisms of natural selection [70]. They use genetic crossover and fitness selection to find the optimum of a particular problem. A number of researchers have used genetic algorithms to find the optimal aberration-correcting phase

Algorithm 3: Heuristic Steepest Descent Algorithm

aIn : Input Zernike coefficients
aOut : A set of output Zernike Coefficients

```
1  for aIdx ← 3 to 15 do
2      for mult ← −1 to 8 do
3          aOut ← aIn;
4          val ← aIn[aIdx];
5          if val < threshold then
6              val ← threshold;
7          end
8          aOut[aIdx] ← val × mult/7;
9          GenerateHologram(aOut);
10     end
11 end
```

Table 3.1 Number of genes vs. Number of permutations

Number of genes	Number of permutations
2	2
4	6
6	20
8	70
10	252
12	926

mask [71]. We have implemented this method in our system as well, identified a number of shortcomings of the standard approach and made appropriate improvements.

We first implemented a non-deterministic tournament selection [72]. A small number of candidate solutions (5-20) are chosen at random. Among the selected candidates, we probabilistically choose one. The candidate with a smallest value of fit function is the most probable to be chosen and the rest are assigned decreasing probabilities. The selected individual becomes the first parent. The same procedure is repeated to select a second parent.

Having the parents, we need to produce offspring by combining their parameters. The majority of genetic algorithms produces two children out of each crossover. We found that procedure to be hugely ineffective, as a single crossover rarely brought improvement despite very good fit function values from both parents. Following this observation, we constructed an alternative crossover mechanism where we generate all the possible combinations such that half of the children's parameters come from each parent. Having $2n$ parameters and

Table 3.2 Grouping Zernike coefficients depending on the aberration

Group number	Aberration name	Zernike Coefficients
1	Defocus	a_3
2	1st Order Astigmatism	a_4, a_5
3	1st Order Coma	a_6, a_7
4	1st Order Spherical	a_8
5	1st Order Trefoil	a_9, a_{10}
6	2nd Order Astigmatism	a_{11}, a_{12}
7	2nd Order Coma	a_{13}, a_{14}
8	2nd Order Spherical	a_{15}

choosing n parameters to switch, the number of ways it can be done is the number of permutations, calculated in the Table 3.1. As the number of genes increases, the number of children generated using this method grows rapidly. We want to incorporate as many parameters as possible, but at the same time have many crossovers within a single iteration. Without running into very long execution times, 8 parameters is the maximum number that can be incorporated. Nonetheless, if it were possible to shorten the hologram processing time, it would be beneficial to increase the number of genes to 10 or more.

Since the genome contains 13 Zernike coefficients, 8 groups need to be formed from them. We decided to perform the grouping by identifying the aberration that particular coefficients correspond to. The assigned groups can be seen in table 3.2.

The next element is the mutation operation. In our system, this role is divided between the addition of purely random candidates into the solution population and heuristic SD optimization. The random candidates, when selected for the crossover, serve as a source of the new genetic material. The number of such random candidates has to be selected. When this number becomes too big, our population will be random and will not improve every iteration. Too small a number would mean that no genetic material flows into the loop. Initial experiments have shown that around 10% random candidates is the suitable fraction.

3.6.3 Hybrid Algorithm

The genetic algorithm proved effective in terms of finding a global minimum of a function, but was relatively slow to explore the space in the proximity of it. On the other hand, heuristic SD was very quick in finding a nearby local minimum, but once it did, it got stuck there and was unable to escape. To combine the best features of the two algorithms, genetic algorithm is used to explore the solution space and find the correction candidates while heuristic descent

fine-tunes these already existing candidates hoping to find the global minimum. An early version of this algorithm was described in [67]. Continuous improvements were made to the algorithm over a period of 2 years following extensive testing.

3.7 Highly-parallel, error-resistant implementation

Throughout the duration of the project, the feedback loop process was continuously improved. After several iterations of this beta-testing process, the mechanism was made resistant to the most common errors in order to produce high-quality correction even in the most harsh conditions. The technical discussion, outlining how this was achieved can be found in the Appendix A.

3.7.1 Hologram generation kernel

The efficient implementation of the kernel is crucial to a fast operation of the entire mechanism. Hence, it was implemented using highly-parallel GP-GPU programming in nVidia CUDA C.

The kernel supports few modes of hologram generation:

- Generating permutations given two parents
- Producing purely-random candidates
- Performing heuristic descent optimization

3.7.2 Picture acquisition module

The picture acquisition module is called right after the holograms are produced. It then sequentially displays the holograms on a secondary screen (SLM) and captures the replay field image. It supports traditional webcams, as well as a dSLRs in LiveView mode [73]. A screenshot of the module in operation can be seen in Fig. 3.10.

3.7.3 Main feedback loop control script

To ensure the stability of the algorithm as well as easy debugging, the main control script is written in MatLAB and calls the picture acquisition module (implemented in C#) and the hologram generation kernel (implemented in highly-parallel CUDA C).The flow chart of the full algorithm can be seen in Fig. 3.11.

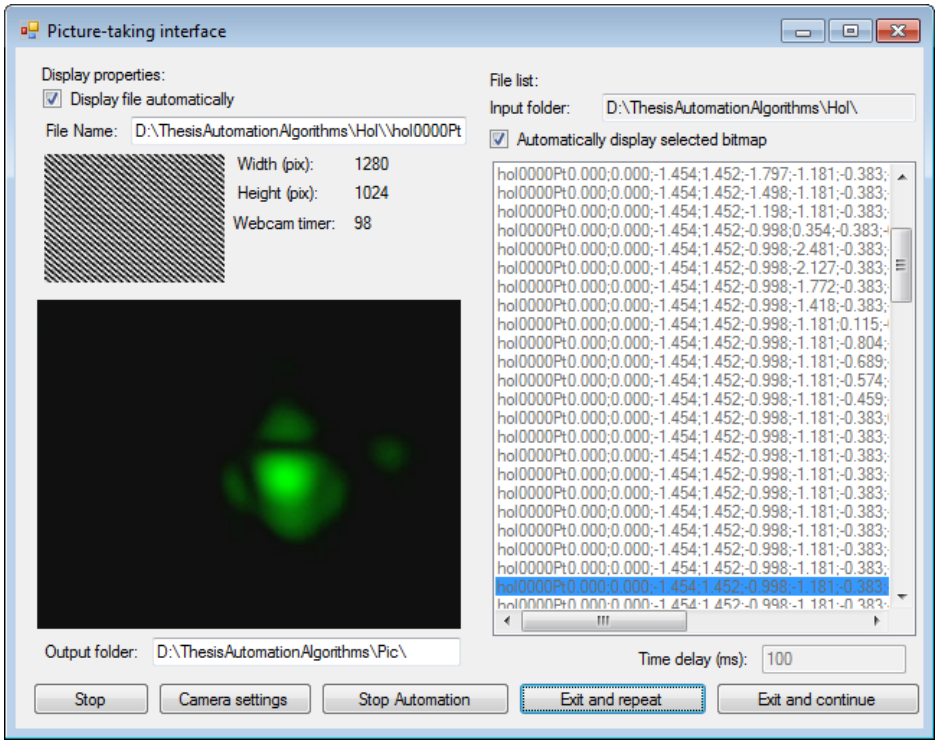


Fig. 3.10 Screenshot of a picture acquisition module in operation

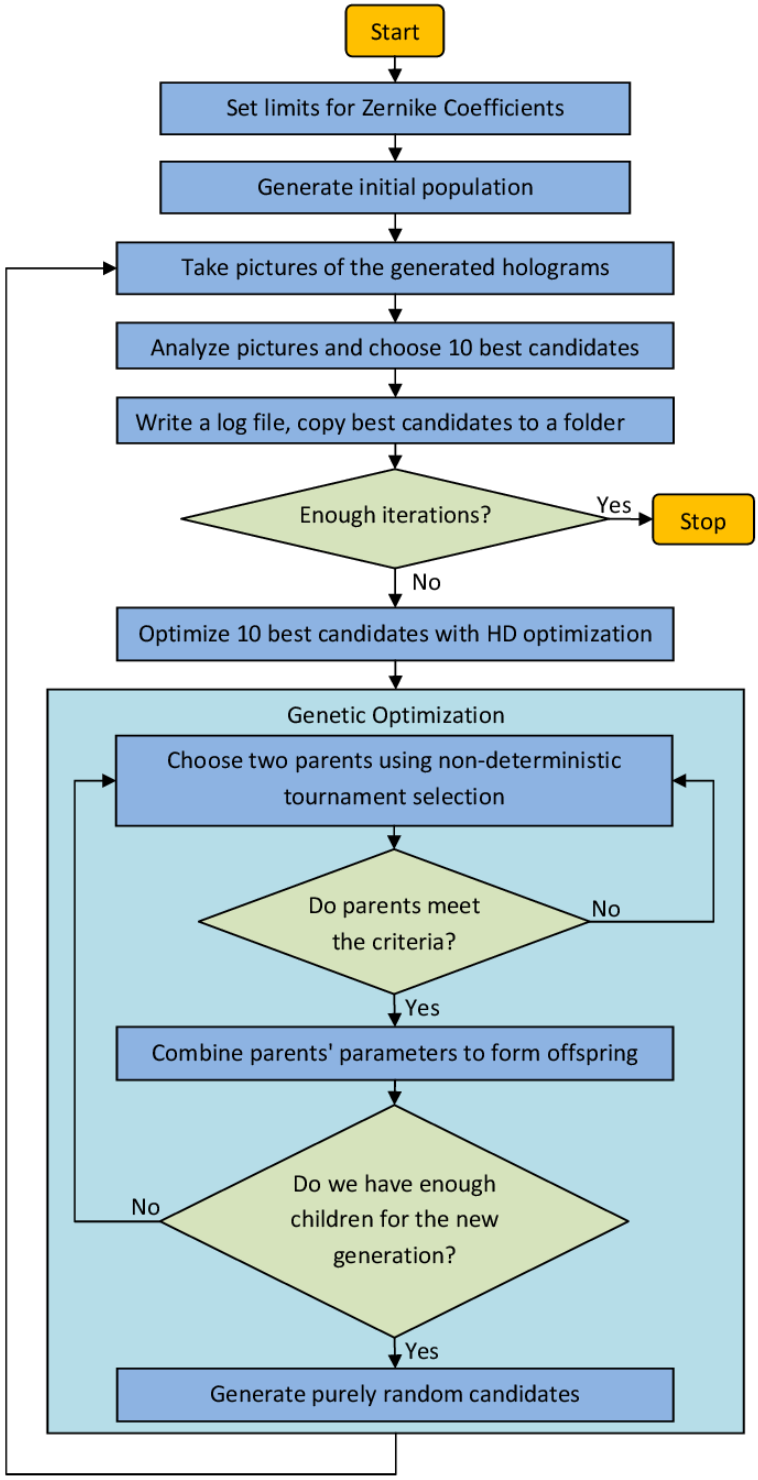
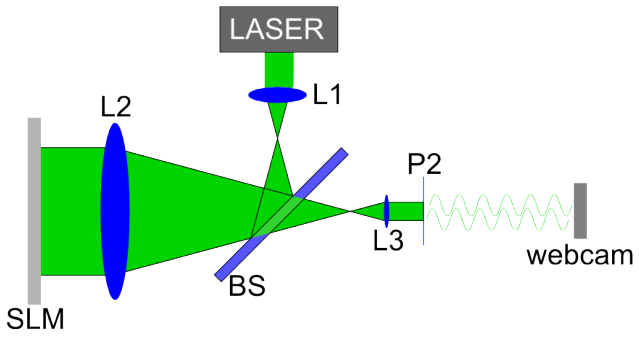
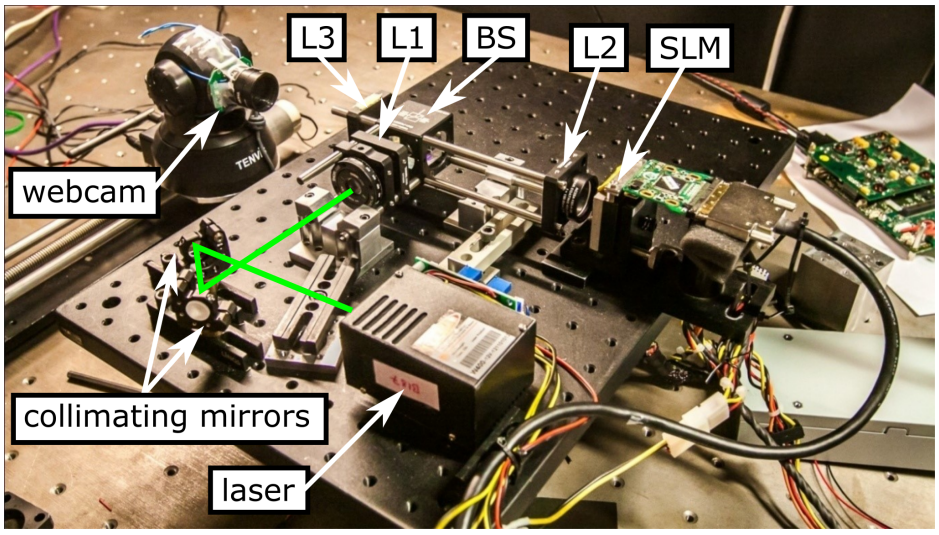


Fig. 3.11 A flowchart of the main feedback loop

3.8 Experimental Setup



(a) The outline of the projector used for experiments



(b) The assembled projector

Fig. 3.12 The projector used in the research

Two projectors are used in this research. One of them was designed and built by Jon Freeman [56], the other is an optically-identical copy, which has been slightly misaligned to mimic a realistic factory assembly. The schematic diagram, outlining the layout of the projector facing a webcam is presented in 3.12a. The laser beam is expanded by the lens L1 ($f_1 = 5mm$) and collimated lens L2 ($f_2 = 150mm$). After the beam is reflected off a beamsplitter (BS) and modulated by the SLM in reflection mode, is then demagnified by the combination of lenses L2 and the front 3mm Sapphire Ball Lens L3 ($f_3 = 3.4mm$). In plane P2 we see a reproduction of a hologram (with introduced phase aberrations), which undergoes diffraction and forms an image on the webcam.

3.9 Results

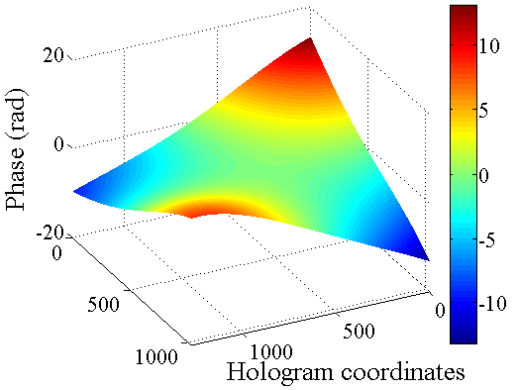
For the purpose of algorithm performance evaluation, all of the tests were performed on the projector's central point. It is postulated that this particular point had a special importance. Because the system is on-axis, its optical performance is very good in the centre of the field and most of the aberrations are introduced by the spatial light modulator. Therefore, correcting that particular point, implicitly means characterizing the non-flatness of the SLM. That property of our feedback loop was first demonstrated in [67]. Next, we present results from the correction of two projectors.

3.9.1 Projector I

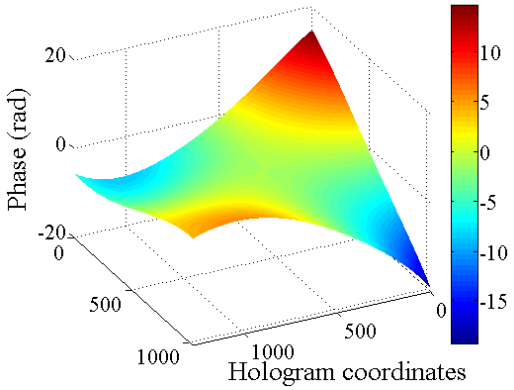
This device was originally designed, built and corrected by Jon Freeman [56, 57]. Freeman calculated the spatial variation of aberrations by simulating the system using the Zemax software. There are, however, a number of errors that the Zemax approach cannot account for, the non-flatness of a spatial light modulator being the best example. Not only does the non-flatness differ from model to model, but also, it can differ depending on the mounting process of the SLM within the projector [56]. Freeman handled this problem by precise interferrometric measurements of the flatness, where one mirror of the Michelson interferrometer was replaced by the SLM under test. This measurement had to be done after the SLM was mounted in order not to perturb the flatness. This approach has a number of disadvantages. It requires setting up a separate experiment, as well as great caution while performing a measurement. Once the measurement is done, the phase needs to be unwrapped and converted to a phase mask which then can be used to correct the non-flatness error. Needless to say, there are many steps in this process. Every single one requires personal attention and each one is subject to additional errors.

Our correction methods have the advantage of being completely independent of the type of the device. The measurements are performed when the projector is already in its assembled state. Once characterized, the correction stays with the projector for its entire lifetime and the quality of the displayed image is improved to that as if high-quality optics was used.

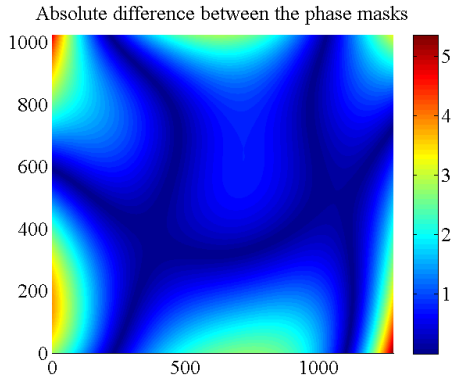
To evaluate the performance of the methods described here, the feedback loop was run on the projector built by Freeman, which had the non-flatness profile characterized using an interferrometric measurement. Below, we compare these two measurements against each other. The feedback Loop system is invariant to tip and tilt aberrations, and the specific position of the front lens affects defocus aberration. Therefore, to qualitatively judge the similarities between the masks, these aberrations were subtracted from both of the masks.



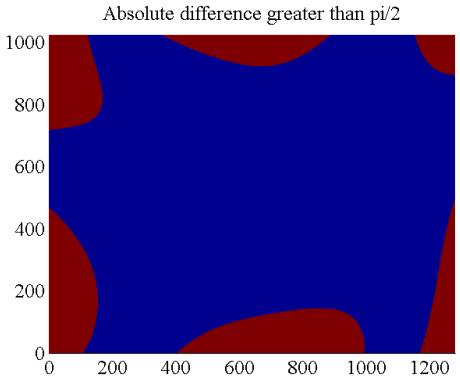
(a)
Mask from interferrometric measurements



(b)
Mask from the feedback loop correction



(c)
Absolute difference between phase masks



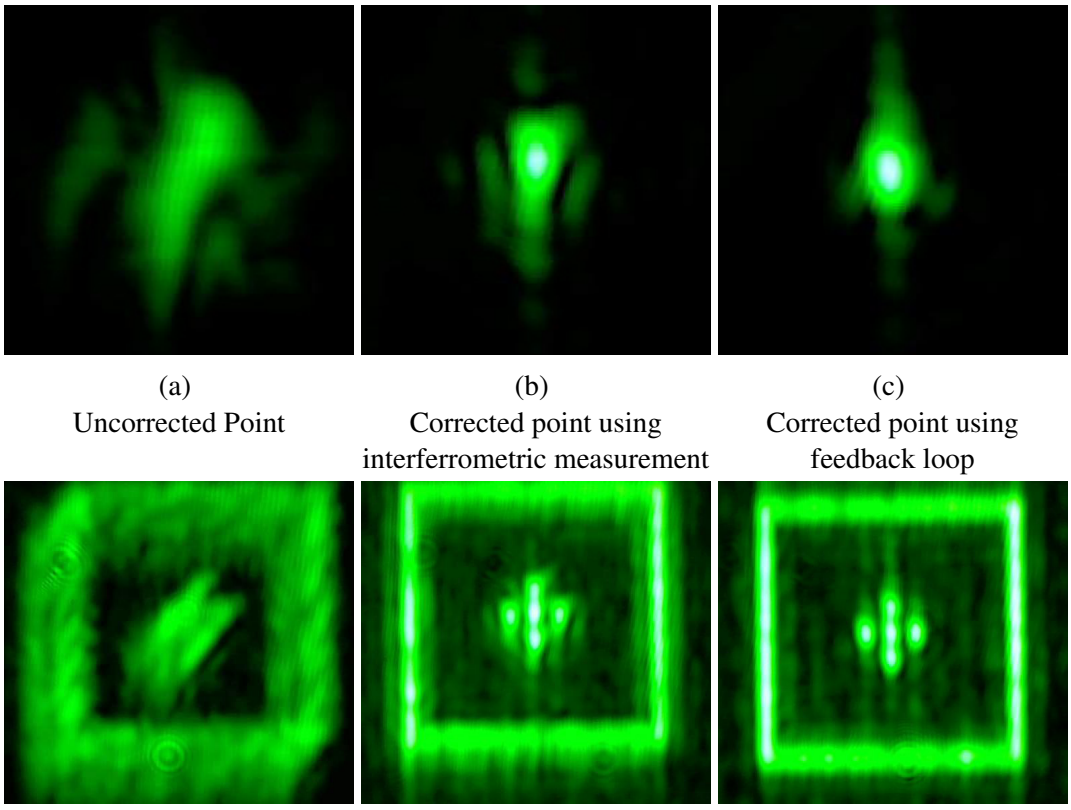
(d)
Absolute difference greater than $\frac{\lambda}{4}$

Fig. 3.13 Phase masks comparison

It can be seen in Figs. 3.13a - 3.13b that the masks are broadly similar in shape. The feedback loop masks shows more curvature, due to a limited number of Zernike polynomials, which in practice work as low-pass filters.

To compare the masks more quantitatively, an absolute difference between them is calculated, as seen in Fig. 3.13c. An interesting observation can be made here - close to the centre of the SLM, the masks are almost identical, but start to differ further from the centre. Going back to the projector's setup, we can understand, why it might be the case. Having Gaussian illumination means that the central points affect the replay field more strongly than the outliers. Therefore, the feedback loop will have a tendency to favour the central points, as they contribute more to the total intensity sum.

Nonetheless, that difference is relatively small. Assuming the Rayleigh quarter-wavelength rule [74], the pixels that differ more than $\frac{\lambda}{4}$ can be seen in Fig. 3.13d. All of them lie on the edges and make up of 18% of total pixels.



(a) Uncorrected Point

(b) Corrected point using interferrometric measurement

(c) Corrected point using feedback loop

(d) Uncorrected Square

(e) Corrected square using interferrometric measurement

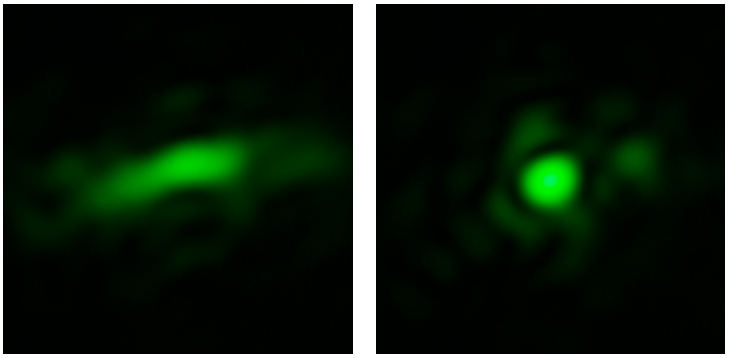
(f) Corrected square using feedback loop

Fig. 3.14 Comparison of Feedback Loop result with Interferrometric measurements

In Figs. 3.14a - 3.14c the comparison of single points can be seen: without correction [Fig. 3.14a] with interferrometric correction [Fig. 3.14b] and with feedback loop correction [Fig. 3.14c]. For the purpose of visual inspection, a test shape was created, it is a square of pixels with 5 single pixels, forming a cross in the middle. That was decided to be the shape tailored to test the quality of the correction, such as the interaction between single pixels from all the directions, vertical and horizontal lines. It can be seen in Fig. 3.14e that the feedback loop results are surprisingly superior to the interferrometric measurement results seen in Fig. 3.14f. The reason for this might be the fact that the interferrometric measurements were acquired a long while ago and in between that time, various external factors might have influenced the slight changes in flatness of the SLM (for instance strains imposed by moving the projector and changing the front lens).

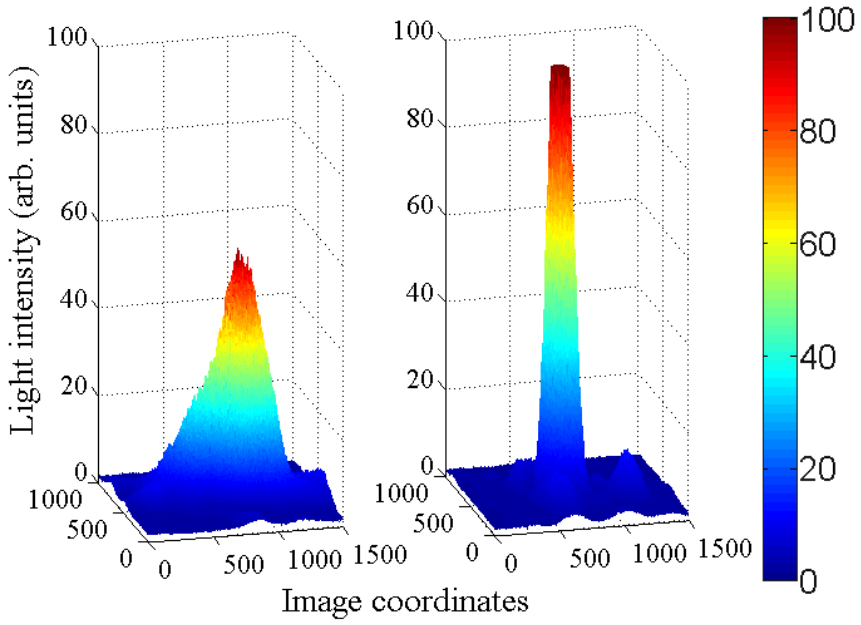
3.9.2 Projector II

The second projector is optically identical to the first one, but has been intentionally miscalibrated to mimic a realistic factory assembly. Correction of this projector can be seen in Fig. 3.15. It can be seen that a corrected point (Fig. 3.15b) has a nice, round shape and the peak intensity nearly twice as high as the uncorrected point (Fig. 3.15c).



(a) Uncorrected spot from Projector II

(b) Corrected spot from Projector II



(c) Surface view of the corrections: Uncorrected (left) and Corrected (right)

Fig. 3.15 Correction for Projector II

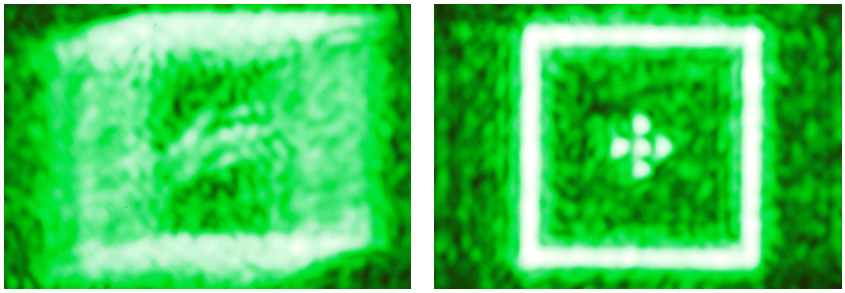
It was previously hypothesized that, because lenses usually perform best in the field centre, the aberrations of the central point should be identical to the non-flatness of the SLM used.

To verify this hypothesis, the following two properties will be tested:

- The correction should be independent of the front lens used (L3)
- The correction should scale with the wavelength. If one wavelength λ_1 was used for correction, and the other wavelength λ_2 was used to display a shape, the optimal phase mask should be $\propto \frac{\lambda_2}{\lambda_1}$ [67]

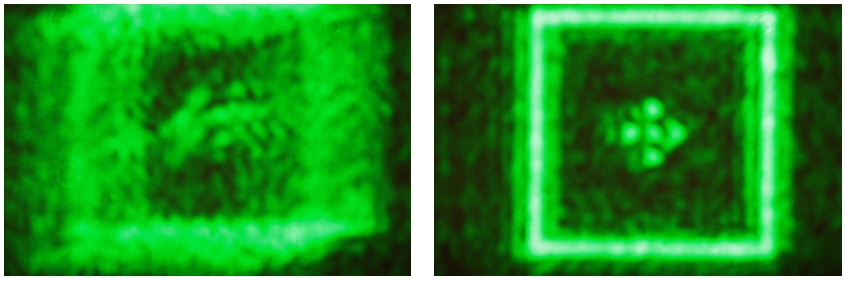
In Figs. 3.16 and 3.17, we are presenting the results of the two tests.

The measurement performed using two different lenses: one high-quality CCD lens and a poor-quality ball lens can be seen in Fig. 3.16. It can be seen that in both cases, a given phase masks corrects the image well, indicating that the majority of the aberrations come from the SLM non-flatness.



(a)
Poor-Quality Ball Lens
Uncorrected square

(b)
Poor-Quality Ball Lens
Corrected square



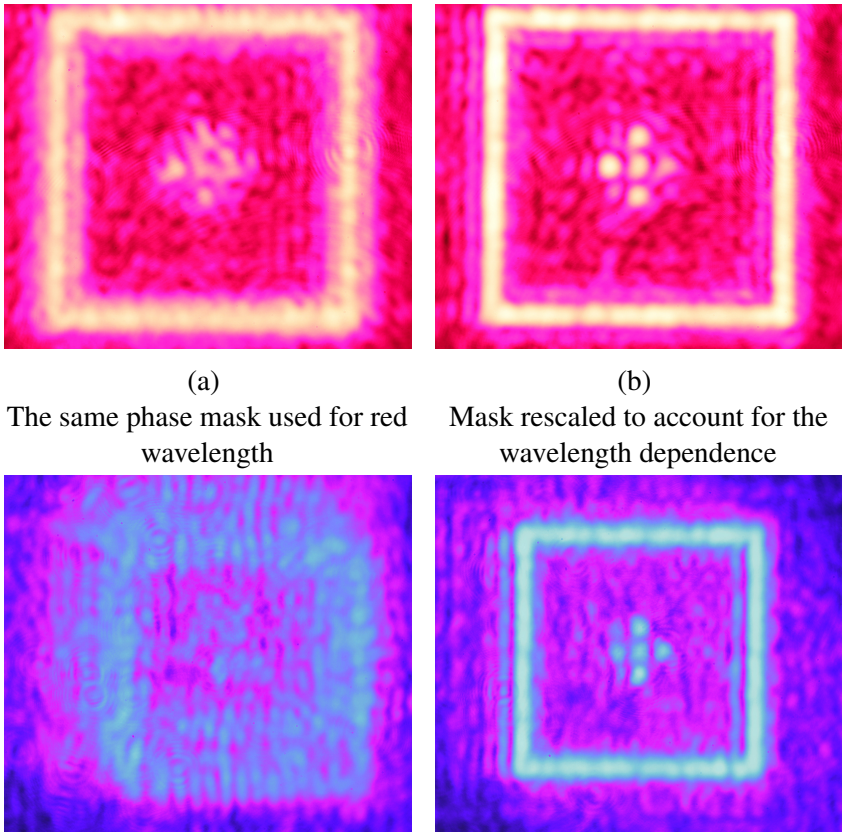
(c)
High-Quality CCD Lens
Uncorrected square

(d)
High-Quality CCD Lens
Corrected square

Fig. 3.16 Testing the flatness of a projector with different front lenses

The results from the second test of our hypothesis can be seen in Fog. 3.17. Here, different wavelengths are used to test to what extent the aberrations come from the non-flatness. In Fig. 3.17a we have used the same mask, as for the green wavelength and in Fig. 3.17b we have rescaled the mask by a factor $\frac{\lambda_1}{\lambda_2}$. It can be seen that the rescaled mask produces a sharp image. In Fig. 3.17c, we have rescaled the mask with an appropriate factor, but the image still appeared out of focus. That error was accounted for by adding a fourth

Zernike polynomial. This behaviour most likely comes from a chromatic aberration in one of the lenses combined with a slight defocus of an input laser beam.



(a) The same phase mask used for red wavelength

(b) Mask rescaled to account for the wavelength dependence

(c) Rescaled mask

(d) Rescaled and re-focused mask

Fig. 3.17 Testing the flatness of a projector with different wavelengths (λ_R = 650nm, λ_G = 532nm, λ_B = 450nm)

Therefore, we can conclude that the aberrations of the central point in the field indeed come from the non-flatness of the SLM.

3.9.3 Correction algorithms comparison

The fact that the presented algorithm did find the correction on two given projectors does not yet prove its utility in all the possible cases. In practice, it is impossible to assess the performance of these algorithms on many devices with different aberrations and the flatness of the SLM. Here, a different approach has been taken. Aberrations can be simulated using a summation of Zernike polynomials. Starting an algorithm from a different set of coefficients

in practice means introducing an arbitrary aberration to the system and attempting to correct for it.

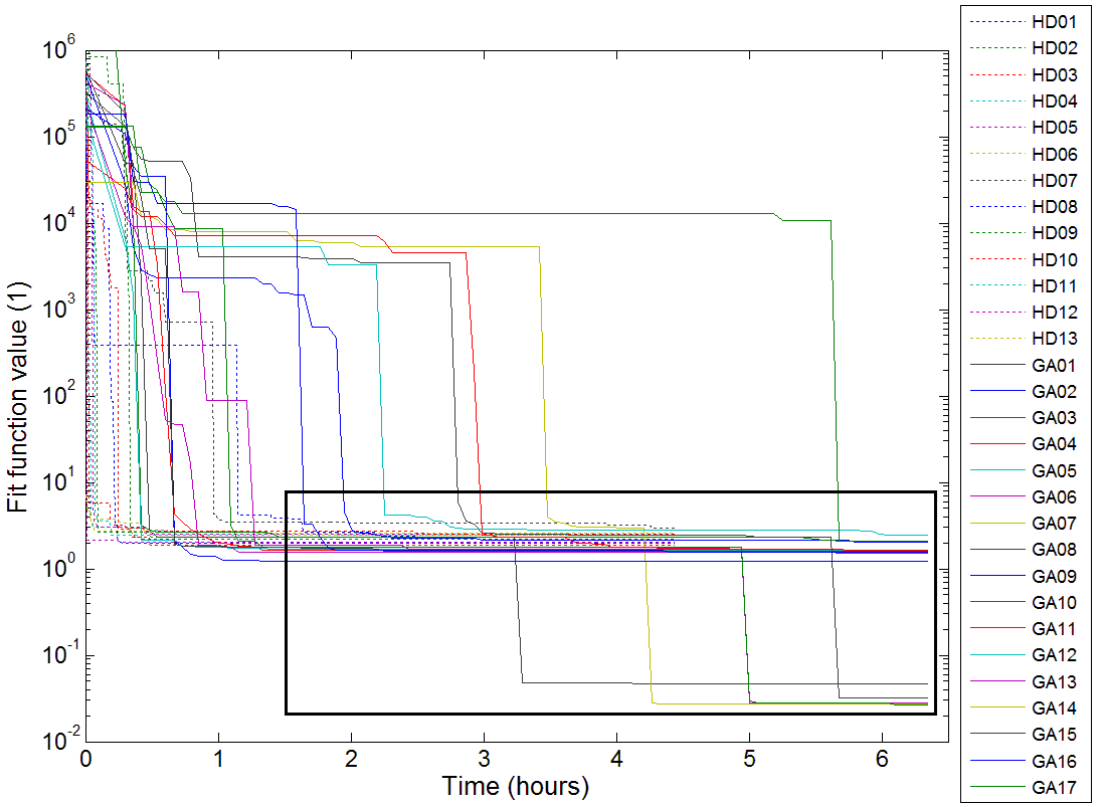
A similar study was performed on the early version of the algorithm and can be found in [67]. The measurements were refined for two reasons, firstly, to validate their correctness, and secondly, the operation of the algorithm was slightly improved, since the publication of previous results.

This procedure has been carried out for the two algorithms: heuristic steepest descent (HD), and the hybrid genetic algorithm (HGA).

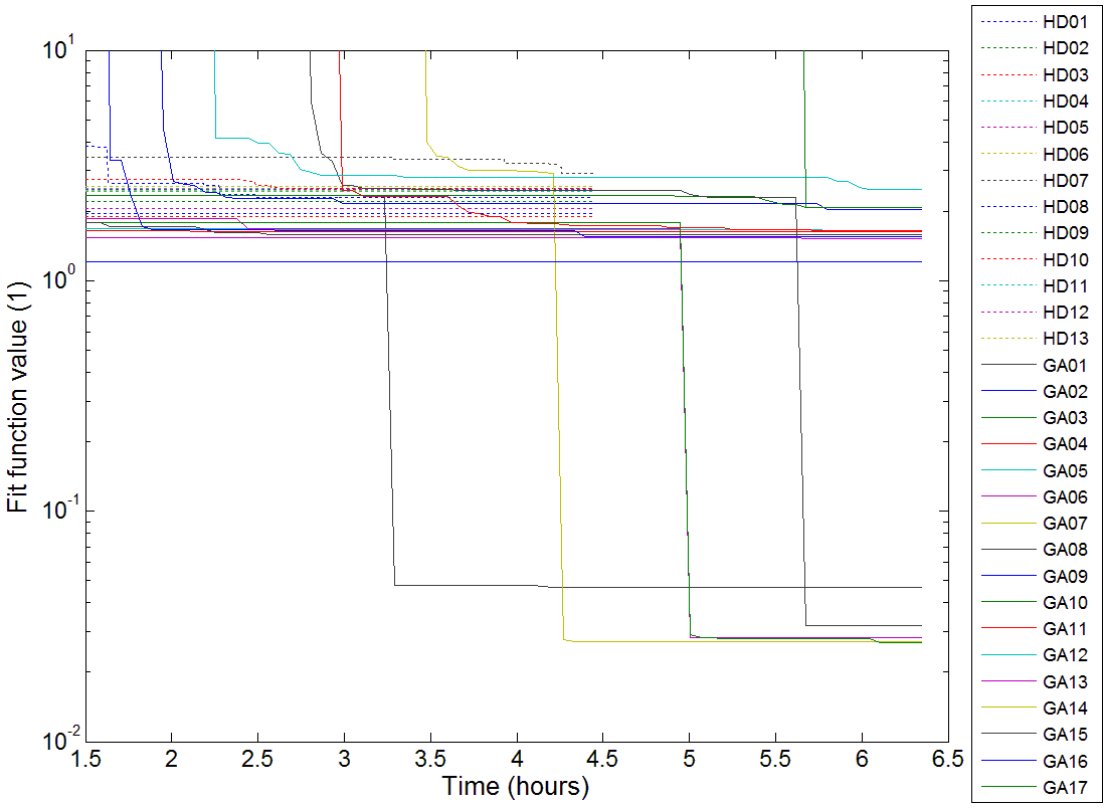
A serious problem that had to be faced was the failure of equipment. The laser was often flickering, as a result of hours of experimental time and the webcam proved unreliable. That degraded the quality of measurmenets. Hence, extra caution has been taken to ensure the reliability and repeatability of measurements. Including:

- Each algorithm has been run 50 times with a different starting point
- The results were later inspected. Whenever an error (such as the failure of the laser or the webcam) was anticipated, the run was discarded
- The remaining runs were then gathered and have the pictures of the replay field with the same coefficients re-taken
- Whenever a new run was being photographed, a reference hologram was taken first, as to have the comparison of light intensity for different runs
- If the two measurements matched, and the light intensity of a known hologram was comparable, the run of the algorithm was qualified as reliable

Only 13 runs of HD and 17 runs of HGA were qualified as error-free. The results are presented in Fig. 3.18. Fig. 3.18a shows all of the runs from start to finish, while Fig. 3.18b is a close-up corresponding to the black rectangle from Fig. 3.18a. The dashed lines represent the different runs of the HD algorithm, while solid lines correspond to HGA algorithm. The summary of the results, both current and previous [67] can be found in Table 3.3.



(a) Performance of different runs of HD and GA algorithms



(b) Performance of different runs of HD and GA algorithms - a close-up

Fig. 3.18 Performance evaluation of Heuristic Descent (HD) and Hybrid Genetic (GA) algorithms

Table 3.3 Performance evaluation of HD and GA algorithms - a statistical summary

Quantity	Early results [67]		Refined results	
	HD	GA	HD	GA
Minimum fit function value	1.96	2.04	1.90	0.03
Average fit function value	2.54	2.38	2.36	1.25
Standard deviation of fit function value	0.34	0.28	0.28	0.86
Average convergence time (hours)	2.6	3.8	1.4	4.7
Standard deviation of the convergence time (hours)	1.5	1.7	1.4	1.5

Various discrepancies can be found by comparing the values. The first difference that becomes apparent is that the minimum value of the fit function for the Genetic Algorithm in the refined measurements is significantly smaller than in the initial measurements (0.03 vs. 2.04). This value is not just an error, but it's solely due to the construction of the fit function. In few runs of the GA, there was overexposure of the green channel, which caused such small values of the fit function.

The convergence times also differ significantly. It is unknown, what caused this discrepancy and, whether it is a trustworthy result, or simply the effect of failing equipment. This measurement should be repeated with a stable laser in order to verify these claims.

3.10 Conclusions

This chapter presents the design and operation of a feedback loop aberration correction mechanism. The history of adaptive-optics is first presented in order to understand the purpose and design choices made. Because of the constraints of holographic projectors, the mechanism designed here employs a blind, sensor-less optimization based only on the far field image displayed by the projector.

Three basic elements of the feedback loop system are a hologram generation algorithm, a fit function mechanism and correction algorithms.

The hologram generation routine is designed to generate quickly multiple holograms of a single spot with different aberration correction parameters. It is based on the Freeman's Pixel to Wrapped Phase Summation. The mechanism includes second-order aberration correction, represented by Zernike Polynomials 3-15.

In order to shorten the execution time, various optimization techniques were used. The one that provided most speed-up was batching the hologram generation. This way, the

preparation for hologram generation (such as allocation of the memory and computation of Zernike Polynomials) is done only once per batch.

A fit function is then used to assess the level of correction and produces a single number, indicating the correction quality. The fit function designed here is suited for an 8-bit CCD device. These devices are preferred over high-quality cameras, since they are inexpensive and very fast in terms of image acquisition. The fit function mechanism is based on two main concepts: peak intensity and spot shape. The peak intensity is the main indicator of the point's quality. However, 256 intensity levels of an 8-bit device is not enough to provide a clear distinction between the points. Whenever two points have the same peak intensity, the point's shape is used for further differentiation. The round shape of a point, where the majority of high intensity pixels are focused in the centre of the pattern is preferred over a physically large point with large side-lobes. This is achieved by a simple thresholding mechanism and the multiplication with the mask, which assigns higher contributions to points further away from the centre. The final fit function combines all of these techniques and outputs a single number, which is a relative measure of the point's quality.

The crucial element of the feedback loop mechanism is the correction algorithm. Its purpose is, once given the fitness values and correction parameters, to decide what will the parameters for the next iteration be. Two correction algorithm types are considered here, steepest-descent algorithms and genetic algorithms.

A traditional steepest descent algorithm cannot be used in this case, because of the fit function's discontinuity at places where the peak intensity changes. Therefore, the novel variant of the algorithm termed Heuristic Descent is introduced. Instead of calculating the gradient, it samples several values for each parameter and selects the one with smallest fitness for the next starting point.

A genetic algorithm, on the other hand, starts with a completely random population of correction candidates. It then iteratively selects two parents from the pool and combines their parameters to form the offspring. To prevent the stagnation, two parents are chosen probabilistically using tournament selection. The difference between a traditional genetic algorithm and the variant presented here is that instead of producing two children out of single crossover, it produces all the possible combinations of parameters. This method maximizes the genetic variety of the offspring with the hope that at least child will have improved fitness compared to parents.

The best qualities of the two algorithms are combined to speed up the convergence times to form the Hybrid Genetic Algorithm. It uses the Genetic Algorithm to produce new correction candidates, while Heuristic Descent routine is used to optimize existing candidates.

The two algorithms, Hybrid Genetic Algorithm (HGA) and the Heuristic Descent (HD) algorithm are then compared against each other. The HD algorithm proved to converge much faster and provided a reasonably good correction. The HGA algorithm was around 50% slower, but proved more reliable with better average fit function values.

That concludes the development of a feedback loop mechanism. It is then used to characterize and correct the aberrations of two holographic projectors. The projectors are optically identical, but have different flatness profiles. One of them has been intentionally misaligned to mimic a realistic automated factory assembly. It was shown that the misalignment can be corrected using this algorithm.

While correcting the two projectors an interesting feature of the algorithm has been postulated. Since the optical elements are usually reasonably good in the centre of the field, the aberrations of that particular point come mostly from the non-flatness of the Spatial Light Modulator used. This hypothesis is tested by changing the front lens and the laser wavelength. The hypothesis implied that the correction should be independent of the front lens used and should scale with the wavelength. Both of these tests were carried out and gave positive results, implying that the hypothesis is indeed true.

This particular feature of the algorithm was published in IEEE Journal of Display Technology in the article entitled "Optimization-Based Adaptive Optical Correction for Holographic Projectors" [67].

Chapter 4

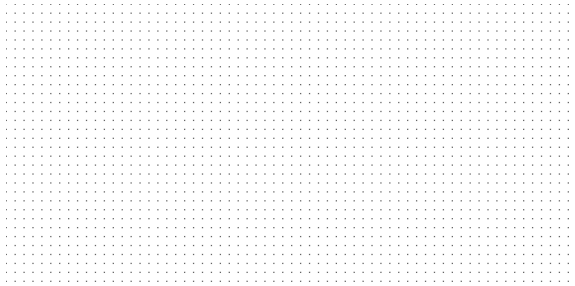
Spatially-varying aberration correction, Piecewise-Corrected OSPR Algorithm

The real time generation of holograms on general-purpose hardware has been unavailable to the public for a number of years. With an introduction of fast hardware and the inception of the One-Step Phase Retrieval (OSPR) algorithm, this goal was achieved for the first time in 2004 [36].

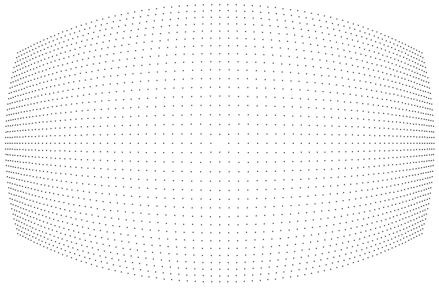
Parallel to that discovery was the construction of Pixel-to-Wrapped Phase Summation (PWPS) algorithm by Jonathan Freeman[56, 57]. PWPS was ingenious in eliminating arbitrary aberrations of optical systems, but suffered from the unfortunate effect of lengthy hologram generation times. In particular the generation time grew linearly with the number of non-zero pixels. For symbology applications, requiring few pixels, the execution times were acceptable. However, for arbitrary video projection, that time could go up to 15 hours per single frame[67, 75].

That figure shrunk to 240 seconds after rewriting the code using highly-parallel General-Purpose General Processing Unit Computing (GPGPU) using nVidia's Compute-Unified Device Architecture (CUDA)[75]. Within the realms of real-time holography, this is still a few orders of magnitude too slow. Therefore, instead of focusing on a more efficient implementation of an old algorithm, the need for a new algorithm was anticipated. Starting from the OSPR algorithm with field-independent correction, this chapter guides the reader through the derivation of an approximated version of PWPS algorithm based on OSPR framework.

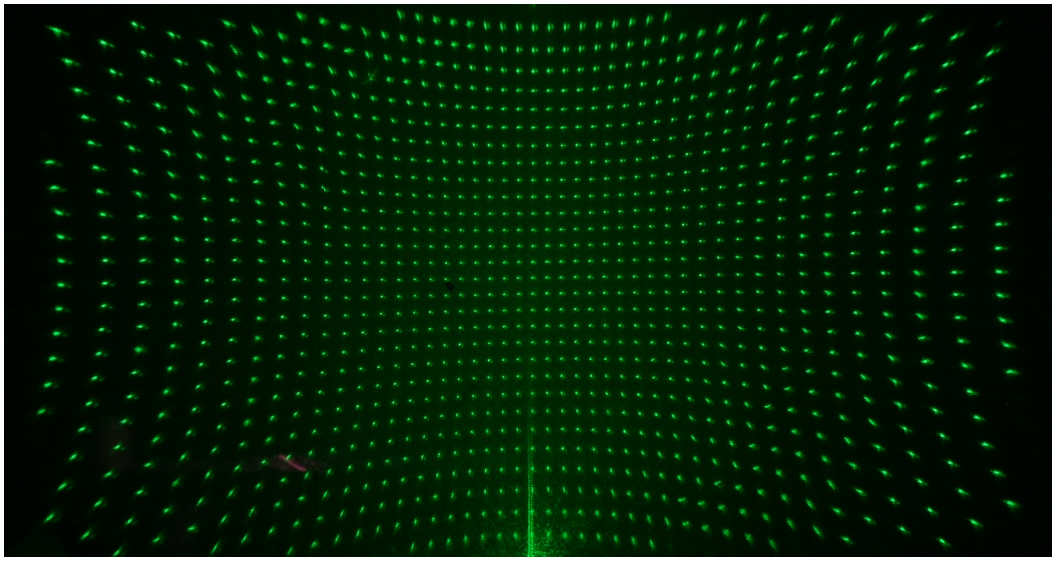
In the following paragraphs, the elimination of the majority of the errors present in holographic projectors is discussed, namely distortions, aberrations and intensity attenuation error, as well as non-uniform sampling.



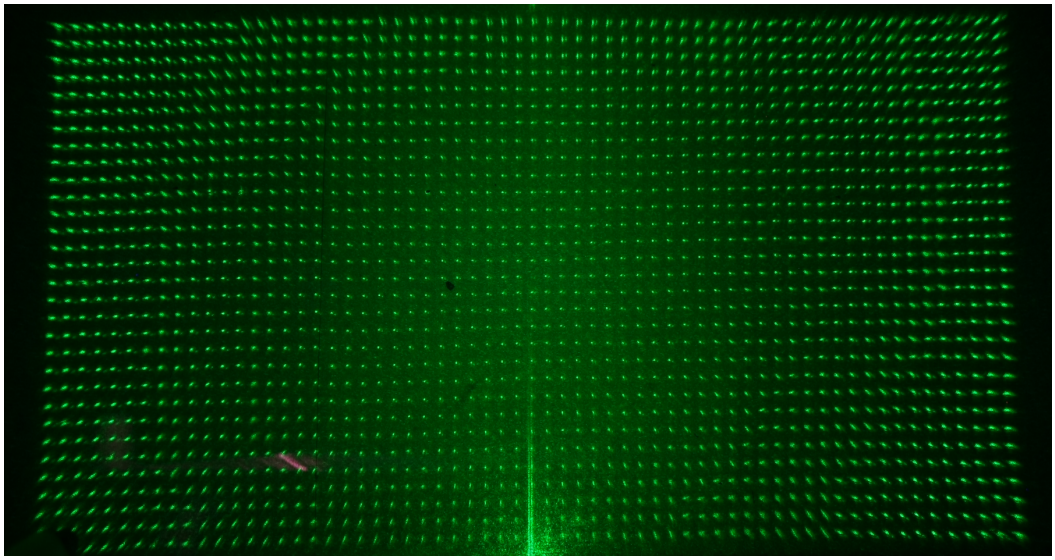
(a) A test image of a grid



(b) An image of a grid with the distortion correction applied



(c) Replay field of the original image



(d) Replay field of the image with distortion correction applied

Fig. 4.1 Distortion correction

4.1 Distortion correction

As discussed in Chapter 1, distortion is the geometric error of the image. Within PWPS, a strong spatial dependence of distortions is not an issue, since all the contributions are decoupled from each other. PWPS handles the distortion correction by changing the output positions of the pixels, which is in practice a non-uniform sampling operation. The image pixels are not any more placed on a regular grid, but instead, the grid is skewed in a way proportional to the distortion of the lens.

The distortion correction from PWPS cannot be ported to OSPR, as it uses a Fourier Transform operation. An FT naturally has a fixed rectangular sampling grid, which cannot be modified. The way to correct for distortion is to apply another counter-distortion before passing the image to the FT routine. This procedure is illustrated in Fig. 4.1. Having a distortion curve measured (described in the following chapter), the correction is straightforward. A sample target image of a grid (Fig. 4.1a) is predistorted to exactly counteract the distortion of the projector (Fig. 4.1b). It can be seen that a significant pincushion distortion seen in Fig. 4.1c is quite well eliminated in Fig. 4.1d.

4.2 Image intensity attenuation

Because of the square shape of SLM pixels, the holographic replay field is modulated by the FT of a square, which is a 2 dimensional sinc function. This error can straight-forwardly be accounted for, once the shape of this function is known. This procedure has already been covered in previous work [40, 76]

4.2.1 Intensity attenuation within PWPS

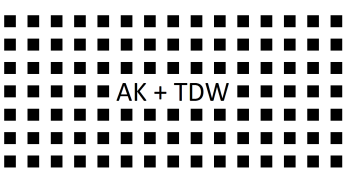
To get a uniform replay field intensity, one has to multiply the input image with the correction term:

$$illumCorr(u,v) = sinc\left(\frac{\frac{M}{2}-u}{M}\right) sinc\left(\frac{\frac{N}{2}-v}{N}\right)$$

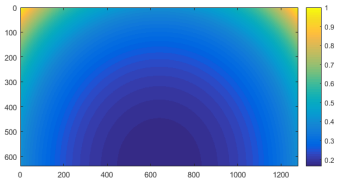
where M is the number of pixels in the SLM.

To illustrate this method in operation, we have simulated the intensity attenuation by assigning each hologram pixel a 2 by 2 pixel block, in a similar way as presented in [76]. After the Fourier Transform is performed, the overall replay field is weighted with an appropriate sinc envelope (Fig. 4.2d). In order to counteract this attenuation, a target image is multiplied

with a mask seen in Fig. 4.2b. Once this procedure is complete, the output image from the projector has an equalized brightness, as seen in the inset Figure 4.2e.

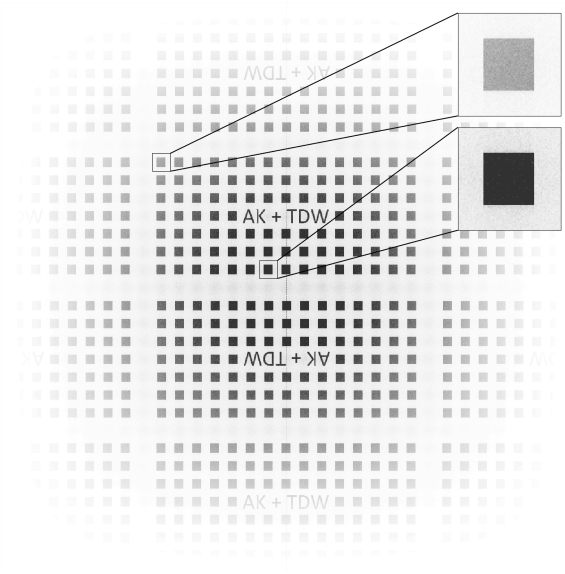


(a) Target test image

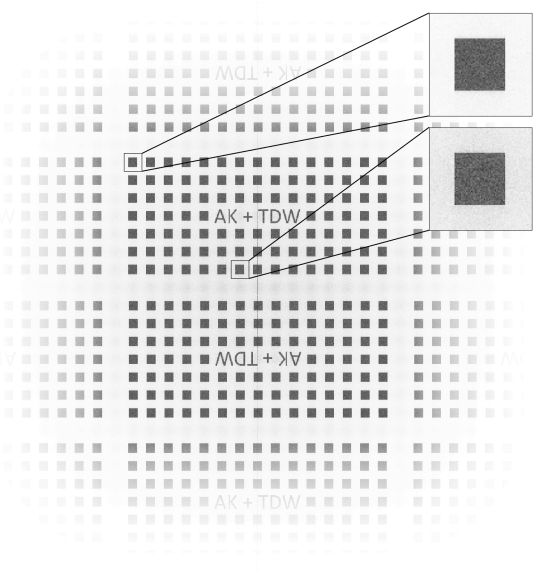


(b) Mask used to counteract intensity attenuation (sinc envelope)

(c) Target image multiplied with the mask



(d) Simulated replay field of the non-compensated image



(e) Simulated replay field of the compensated image

Fig. 4.2 Image intensity attenuation compensation

4.2.2 Intensity attenuation correction within PC-OSPR

Within PWPS, such correction is sufficient to counteract the intensity error of the replay field. However, in the case of a highly-distorted replay field, another contribution to the intensity arises. To explain this phenomenon, we consider a thin ring of pixels (Fig.4.3). When the image gets distorted, a ring at a radius r gets mapped to r'. Since total energy within the ring needs to be conserved, the product of surface area and the intensity has to remain constant. From there, it follows that:

$$r\,dr\,I_{in} = r'\,dr'\,I_{out}$$

Therefore, the input intensity gets attenuated by a factor of: $\frac{r'dr'}{rdr}$.

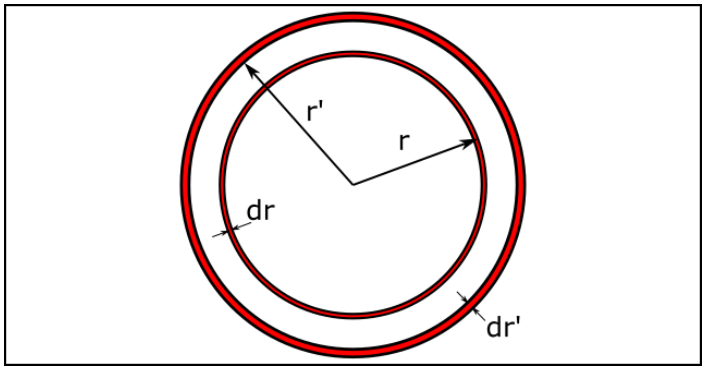


Fig. 4.3 Origin of the intensity attenuation coming from distortion

Given the distortion curve $r'(r)$, the intensity attenuation term is calculated. This curve can be seen in Fig. 4.4a. To form a 2-dimensional correction array for the OSPR algorithm, from every pixel in the replay field, the resultant radius as well as the attenuation factor are calculated (Fig. 4.4b). It can be seen that also at this stage the size of the replay field is restricted to a square. This contribution can be added to a sinc envelope correction for convenience (Fig. 4.4c).

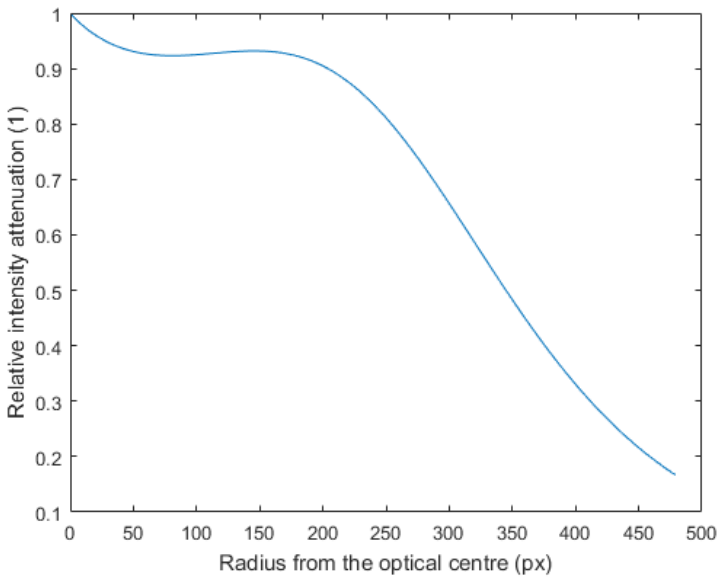
4.3 Aberrations

As discussed in Chapter 2, the aberration phase mask can, in general, be dependent on the spatial coordinates as well as hologram coordinates. For well-designed optics, the spatial dependence is negligible, and therefore, a simple phase mask is sufficient to minimize all of the aberrations. This is not the case for the projector used in this research.

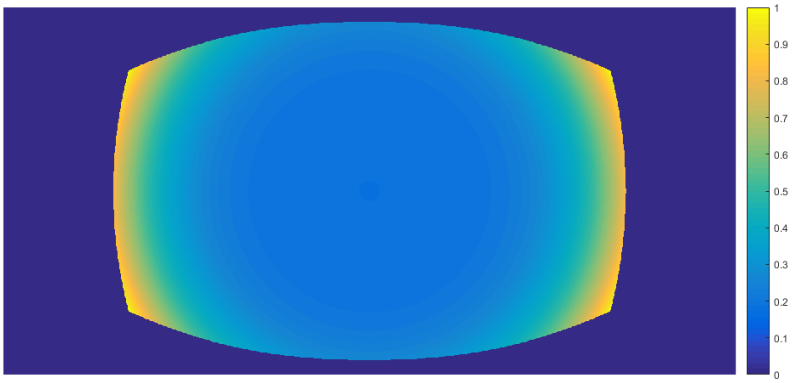
It was noticed that plain OSPR cannot correct spatially-varying aberrations, because all the points are processed simultaneously in a Fourier Transform operation. However, the initial tests revealed that even in the case of highly aberrated replay fields, the aberrations do change rather slowly, as it can be seen in Fig. 4.5. Although the correction was performed in the point, indicated by the dotted circle, the neighbouring points are also fairly well corrected. Due to the approximate cylindrical symmetry of the system, a well corrected region can be enclosed within the ring, boundaries of which are indicated as red circles in Fig. 4.5.

4.3.1 An approximate solution based on Zernike coefficient continuity

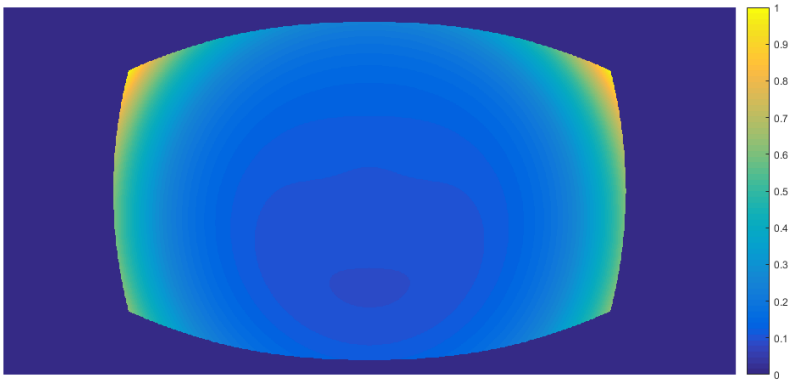
The previous discussion proves that even in the case of a highly aberrated system, such as the ball lens projector, the variation of the aberrations are still quite low in a certain region.



(a) Intensity attenuation as a function of distance from the optical centre



(b) 2D mask correcting intensity attenuation from distortion



(c) 2D mask correcting total attenuation of the replay field from sinc envelope and distortion

Fig. 4.4 Correction of the intensity attenuation coming from distortion

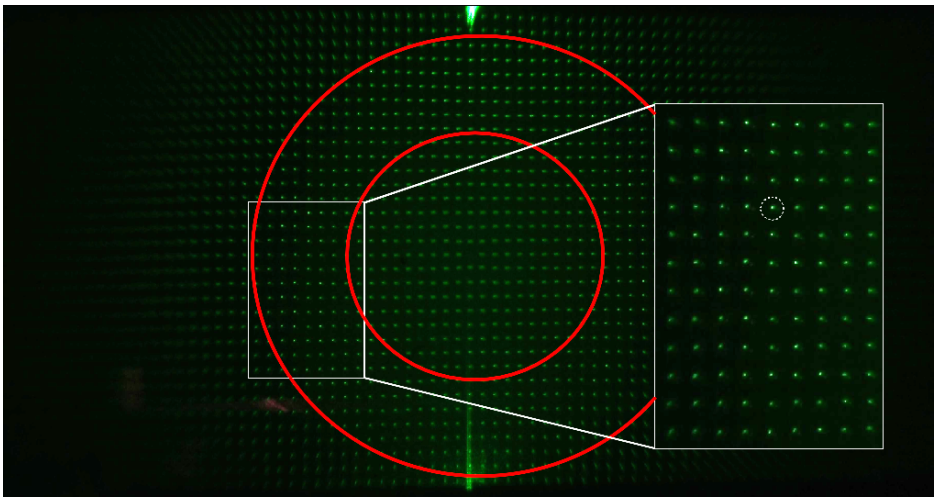


Fig. 4.5 Spatial variation of aberrations

Once the correction is performed on a particular point, there exists a region in which this correction is almost identical.

Once determined a complete set of regions such that each point in the replay field is covered by exactly one region, one can construct an approximate solution to the hologram generation method using One-Step Phase Retrieval algorithm. The formal derivation of this approximation is presented below.

Having determined a set of n phase masks $\varphi_1(x,y)\dots\varphi_n(x,y)$ and the corresponding regions $R_1\dots R_n$, a phase factor can be approximated as:

$$\varphi(u,v,x,y) \approx \begin{cases} \varphi_1(x,y) & \text{if } (u,v) \in R_1 \\ \varphi_2(x,y) & \text{if } (u,v) \in R_2 \\ \vdots & \vdots \\ \varphi_n(x,y) & \text{if } (u,v) \in R_n \end{cases}$$

Using this formula, a summation in Eq. 2.9 can be rewritten in a following manner:

$$\begin{aligned} H(x,y) &= \sum_{u=0}^{u_{max}} \sum_{v=0}^{v_{max}} \sqrt{A(u,v)}\, e^{i2\pi \Phi^{uv}_{pixel}(x,y)}\, e^{i2\pi \varphi(u,v,x,y)} \\ &= \sum_{k=1}^{k=n} \sum_{(u,v)\in R_k} \left\{ \sqrt{A(u,v)}\, e^{i2\pi \Phi^{uv}_{pixel}(x,y)} \right\} e^{i2\pi \varphi_k(x,y)} \end{aligned} \tag{4.1}$$

It is clear that the term inside the curly brackets coming from PWPS routine is equivalent to a simple Discrete Fourier Transform, with only pixels from region R_q present. Therefore:

$$H(x,y) = \sum_{k=1}^{k=n} \mathscr{F}^{-1}\left\{ M_k \sqrt{A(u,v)}\, e^{i2\pi\,\vartheta(u,v)} \right\} e^{i2\pi\,\varphi_k(x,y)} \tag{4.2}$$

where $\vartheta(u,v) \in [0,1]$ is a random variable distributed uniformly and mask M_k has been introduced to filter only the points that belong to the region R_k:

$$M_k = \begin{cases} 1 & \text{if } (u,v) \in R_k \\ 0 & \text{otherwise} \end{cases}$$

This important derivation proves that PWPS can be approximated to practically any precision using this variant of OSPR algorithm. This method was termed Piecewise-Corrected OSPR. An additional advantage of using OSPR-type algorithm is that there exists a number of improvements of OSPR algorithm, such as Adaptive OSPR (AdOSPR) and AdOSPR with Liu-Taghizadeh (L-T) optimization[40]. Using the same framework, the PC-OSPR algorithm can be extended to PC-AdOSPR and PC-AdOSPR-LT.

4.3.2 Piecewise-Corrected OSPR

The idea behind the OSPR algorithm is that time-averaging of the consecutive frames is a powerful noise reduction technique. Due to the usage of very high-frequency ferroelectric Spatial Light Modulators, with frequencies in the range of kilohertz, such holograms can be displayed much faster than the reaction time of the eye. The PC-OSPR algorithm suited for spatially-varying aberrations is presented in Algorithm 4

4.3.3 Piecewise-Corrected OSPR with feedback: Adaptive PC-OSPR algorithm

OSPR reduces noise $\propto \frac{1}{N}$ where N is the number of frames. Even better noise reduction can be achieved by including additional compensation. Once the quantization noise in the holographic replay is known, it can be compensated for in the next time-sequential frames. This approach was termed Adaptive OSPR and proved to reduce the noise as $\frac{1}{N^2}$ [40, 44].

In order to extend PC-OSPR for the adaptive case, the phase correction needs to be undone to get back the original unaberrated image after quantization. The way to do so is to

first correct for aberrations using a phase mask, quantize the hologram and only then apply the phase correction with an opposite sign.

The full algorithm is presented in Algorithm 5.

4.4 Resolution improvement

A very strong distortion of the lens has an effect on the resolution of the image. After the target has been pre-distorted, it can be noticed that towards the edge of the replay field, multiple pixels in the undistorted image map to a single pixel in the output. In order to better understand and quantify this phenomenon, we have plotted the function $\frac{d}{dr}r'$ in Fig. 4.6. It can be seen that this function is approximately 1 until around 200 and then grows rapidly. So, while at the edge of the replay field, almost 3 neighbouring pixels are averaged to one during the distortion compensation procedure. That leads to the dramatic loss of detail further from the centre of the field.

Algorithm 4: Piecewise-Corrected One-Step Phase Retrieval Algorithm (PC-OSPR)

$I(u,v)$: Input target image
N : Number of OSPR frames to generate
$H_{j,quant}(x,y)$: Output j-th binary hologram

Calculate the target field for the first iteration:

$$T(u,v) = \sqrt{I(u,v)}$$

for $j \leftarrow 1$ **to** N **do**

  Add random phase $\vartheta_{j,rnd}(u,v)$ to the target:

$$T(u,v) = T(u,v)\, e^{i2\pi\,\vartheta_{j,rnd}(u,v)}$$

  $H_j \leftarrow 0$

  for $k \leftarrow 1$ **to** n **do**

    Construct a k-th subimage by multiplying the target with an appropriate mask:

$$T_k(u,v) = T(u,v)\, M_k(u,v)$$

    Perform an inverse Fourier Transform:

$$H_{j,k}(x,y) = \mathscr{F}^{-1}\left\{T_k(u,v)\right\}$$

    Apply phase correction and accumulate the result:

$$H_j(x,y) = H_j(x,y) + H_{j,k}(x,y)\, e^{-i2\pi\,\varphi_k(x,y)}$$

  end

  Quantize the hologram to one of the complex SLM states:

$$H_{j,quant}(x,y) = \texttt{QuantizeHologram}(H_j(x,y))$$

end

Algorithm 5: Adaptive Piecewise-Corrected One-Step Phase Retrieval Algorithm (AdPC-OSPR)

$I(u,v)$: Input target image
N : Number of OSPR frames to generate
$H_{j,quant}(x,y)$: Output j-th binary hologram
F : Total visual field

$F \leftarrow 0$

Calculate replay field scaling factor:

$$S_I = \sum_{u,v} I(u,v)$$

Calculate the target field for the first iteration:

$$T(u,v) = \sqrt{I(u,v)}$$

for $j \leftarrow 1$ **to** N **do**

Add random phase $\vartheta_{j,rnd}(u,v)$ to the target:

$$T(u,v) = T(u,v)\, e^{i2\pi\, \vartheta_{j,rnd}(u,v)}$$

$H_j \leftarrow 0$

for $k \leftarrow 1$ **to** n **do**

Construct a k-th subimage by multiplying the target with an appropriate mask:

$$T_k(u,v) = T(u,v)\, M_k(u,v)$$

Perform an inverse Fourier Transform:

$$H_{j,k}(x,y) = \mathscr{F}^{-1}\left\{T_k(u,v)\right\}$$

Apply phase correction and accumulate the result:

$$H_j(x,y) = H_j(x,y) + H_{j,k}(x,y)\, e^{-i2\pi\, \varphi_k(x,y)}$$

end

Quantize the hologram to one of the complex SLM states:

$$H_{j,quant}(x,y) = \texttt{QuantizeHologram}(H_j(x,y))$$

$T_{rec} \leftarrow 0$

for $k \leftarrow 1$ **to** n **do**

Undo the phase correction and reconstruct the resultant image:

$$T_{k,rec}(u,v) = \mathscr{F}\left\{H_{j,quant}(x,y)\, e^{i2\pi\, \varphi_k(x,y)}\right\}$$

Crop out the reconstructed image corresponding to the subimage k and add a total contribution:

$$T_{rec}(u,v) = T_{rec}(u,v) + T_{k,rec}(u,v)\, M_k(u,v)$$

end

Calculate a total reconstructed field:

$$F(u,v) = F(u,v) + \left|T_{rec}(u,v)\right|^2$$

Calculate an adaptive compensation for the next frame:

$$T(u,v) = \begin{cases} \sqrt{(j+1)\, I(u,v) - \frac{F(u,v)\, S_I}{\sum F(u,v)}} & \text{if} \quad (j+1)\, I(u,v) > \frac{F(u,v)\, S_I}{\sum F(u,v)} \\ 0 & \text{otherwise} \end{cases}$$

end

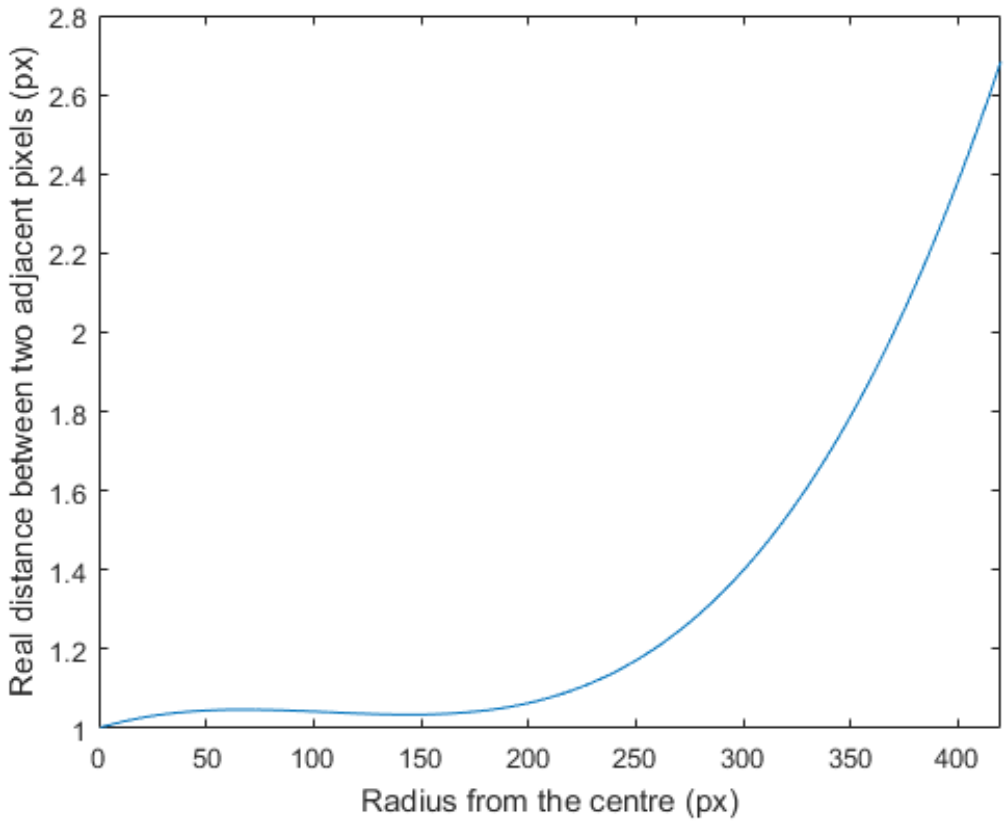
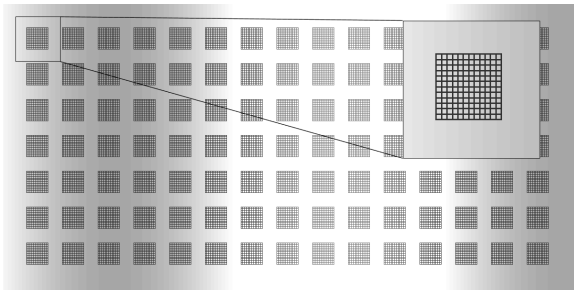
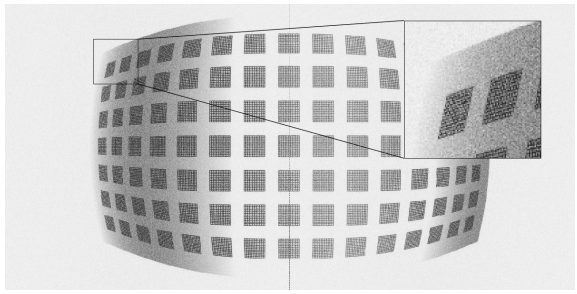


Fig. 4.6 Number of pixels in the input image mapping to one pixel in the output, as a function of radius

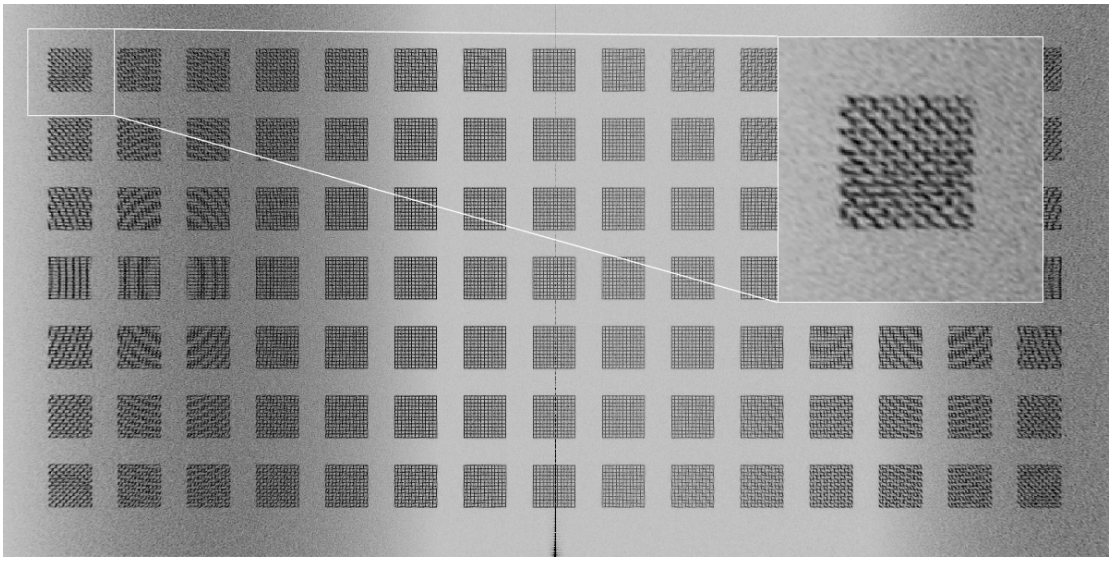
In order to counteract this, one can artificially increase the resolution of the Fourier Transform performed. This is achieved by up-scaling the image by a factor of 2 before the distortion correction is applied and performing an FT at a higher resolution. This procedure, however, increases the noise in the replay field. To illustrate this phenomenon, we have designed a target test image, composed of a set of grids (to judge the resolution), with a slowly-varying sinusoidal background (to judge the noise properties) [Fig. 4.7a]. Using a real distortion curve from the projector, the distortion-correcting hologram was constructed and then, its replay field was simulated[Fig. 4.7b]. To visually judge the effect on the image, the distortion correction was digitally undone for a better visual inspection. The image with native resolution can be seen in Fig. 4.7c. It can be seen that there is a noticeable loss in resolution in the outer edges of the image. While in the doubled-resolution image seen in Fig. 4.7d, this error is eliminated, but the the background noise increases.



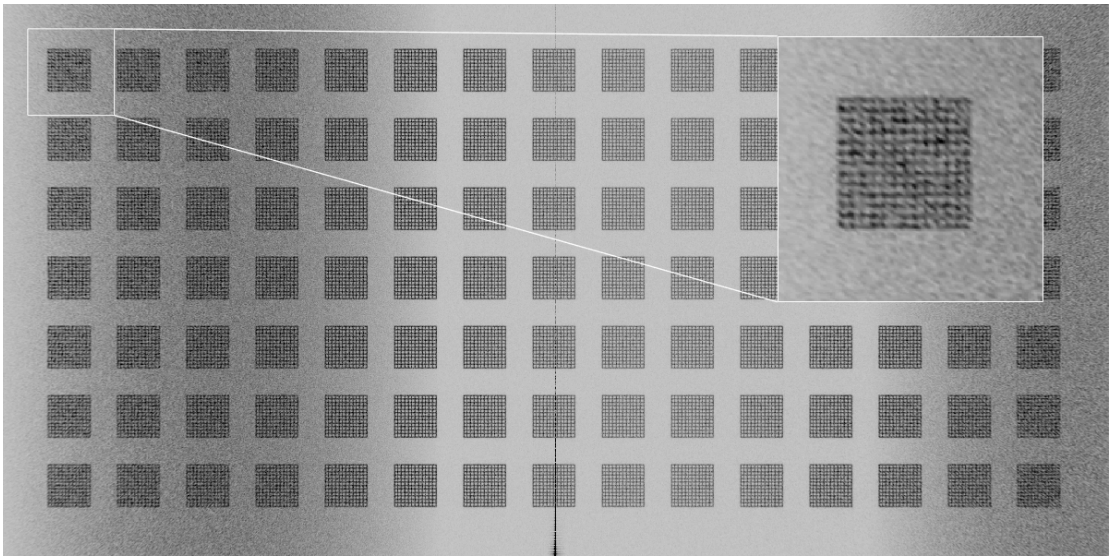
(a) Target test image



(b) Reconstruction of a distortion-corrected hologram



(c) Simulation of a holographic replay with distortion present - native resolution



(d) Simulation of a holographic replay with distortion present - doubled resolution

Fig. 4.7 Resolution of the hologram vs. noise

4.5 Summary of the correction

The correction of various errors was discussed in the preceding sections. By this point, it is assumed that all the errors of the projector have been appropriately characterized. The characterization of these errors is a separate, lengthy topic, which will be covered in the following chapter. Here, the transition from the characterized errors to the fully-corrected holographically-projected images in real time is given. The summary of a full correction is shown in the Fig. 4.8. Each correction element of such system can be switched on independently of all the other elements in order to test the particular effect on the image.

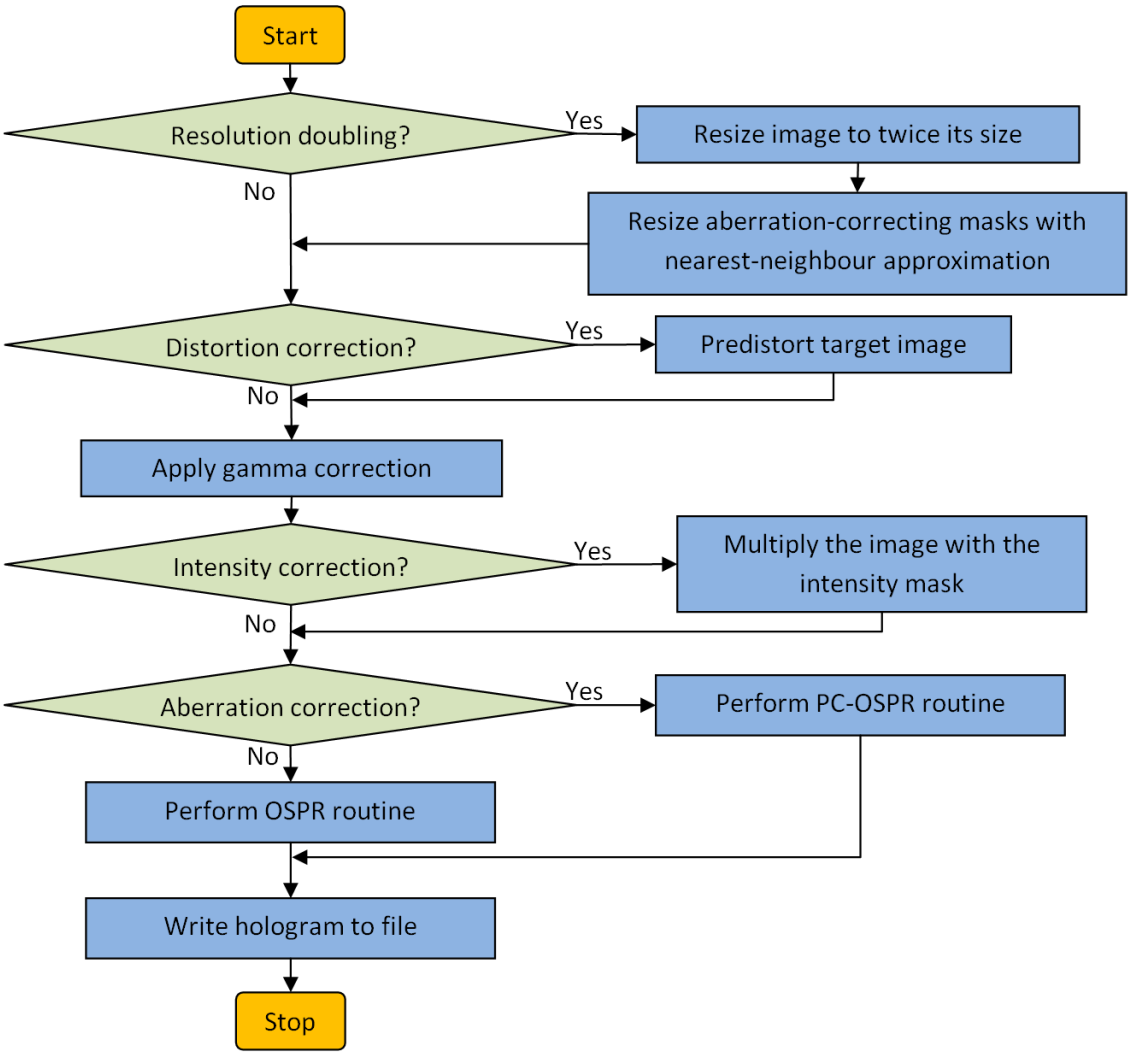


Fig. 4.8 Flowchart, showing a full, modular correction

4.6 Results

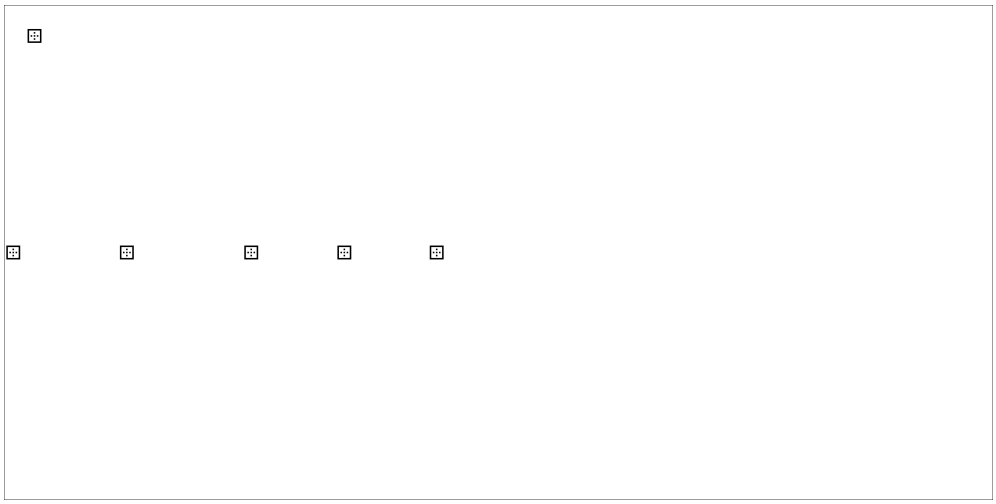
For a matter of clarity, the images shown in this section were converted to black and white by discarding red and blue channels. The intensity was also inverted for a better perception on white background. Therefore, darker colours indicates higher light intensities.

4.6.1 Aberration correction

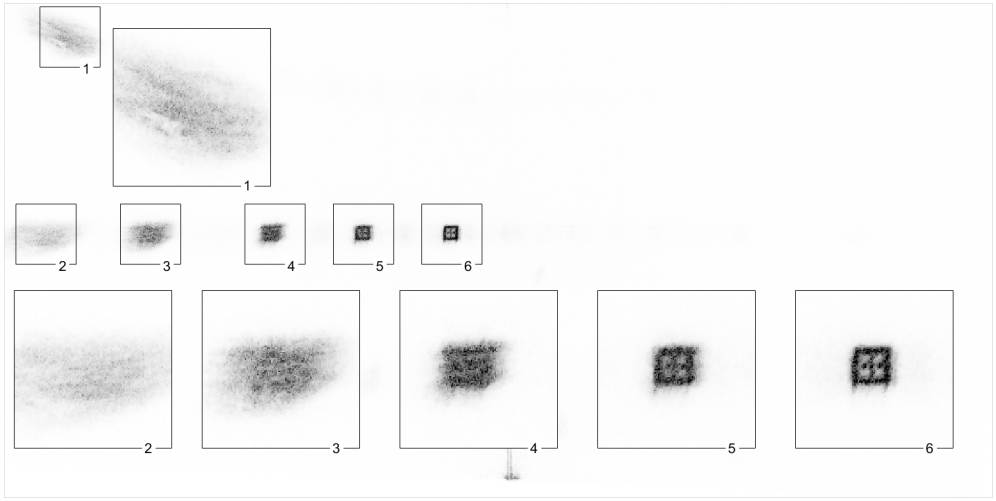
Aberrations of the Projector II have been characterized using a feedback loop mechanism. Because of approximate rotational symmetry, correction for 5 different points along the radius from the optical centre were found. After the further visual inspection, it appeared that these corrections are insufficient to correct the pixels in the top left and bottom right corner, so an additional correction in that region was performed. These 6 points proved satisfactory to correct the majority of the replay field. The 6 chosen correction points are indicated in Fig. 4.9a. The test shape is the same as the one used in Chapter 3: a square with 5 pixels in the centre. The uncorrected replay field can be seen in Fig. 4.9b and the corrected one in Fig. 4.9c. The insets 1-6 are used in order to better visualize particular corrections. It can be seen that correction for points 3-6 is nearly perfect, while points 1 and 2 are suboptimal, but nonetheless, improved significantly.

4.6.2 Aberration region assignment

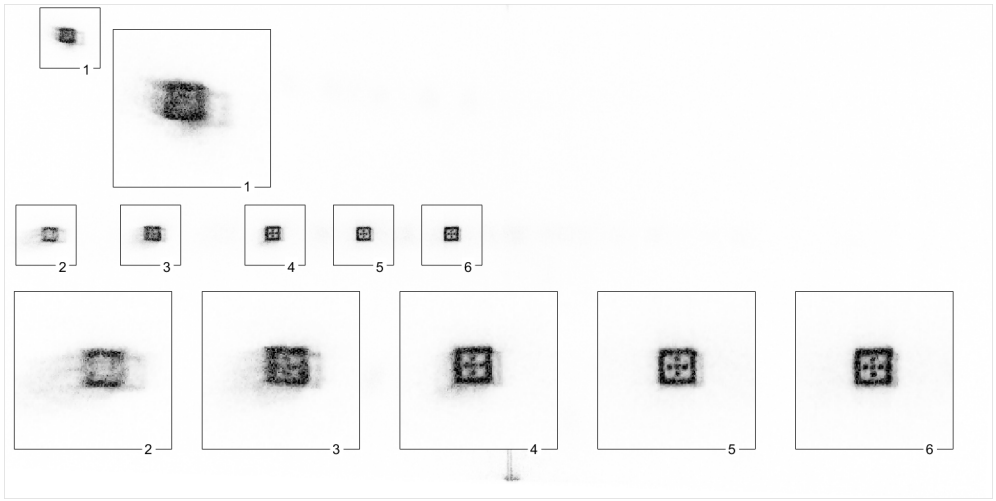
Once the corrections are made, each correction has to be assigned an appropriate mask. A resultant set of masks is presented in Fig. 4.10. Different shades indicate different masks. One can see an approximate rotational symmetry, which breaks further from the optical centre of the image. To present the operation of the mechanism, we multiply the target image [Fig. 4.10b] with one of the masks [Fig. 4.10c]. That region will have one of the phase corrections applied. The uncorrected image can be seen in Fig. 4.10d and then the same region after correction can be observed in Fig. 4.10e. It can be seen that the image is indeed improved significantly.



(a) Position of the corrections in the replay field

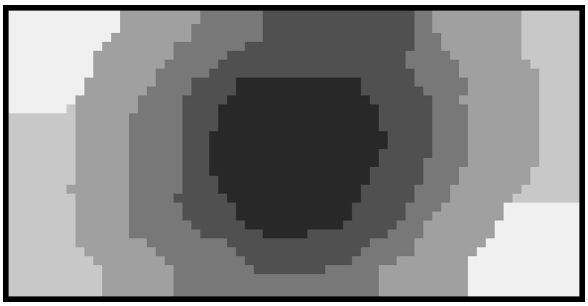


(b) Uncorrected replay field

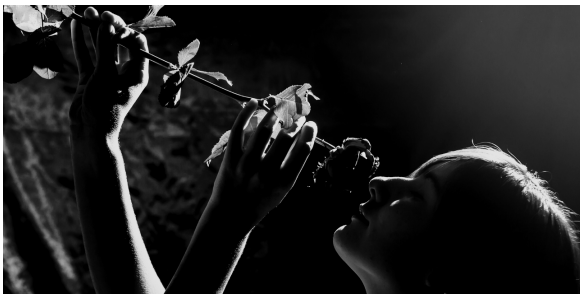


(c) Adaptive-Optical correction

Fig. 4.9 Adaptive-Optical Aberration Correction (color removed for clarity)

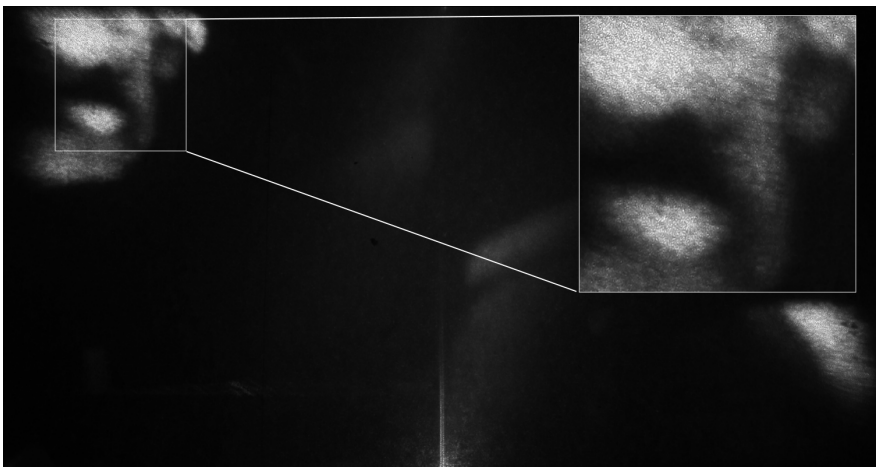


(a) A set of assigned masks

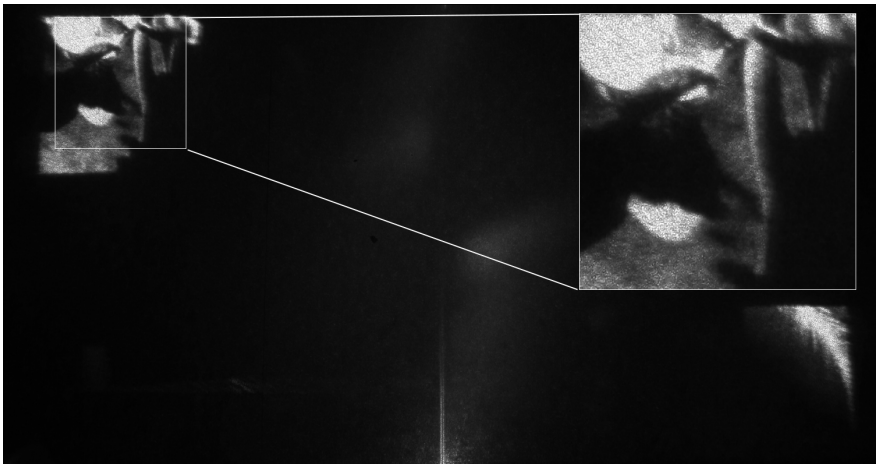


(b) Target test image

(c) Target multiplied with one of the masks



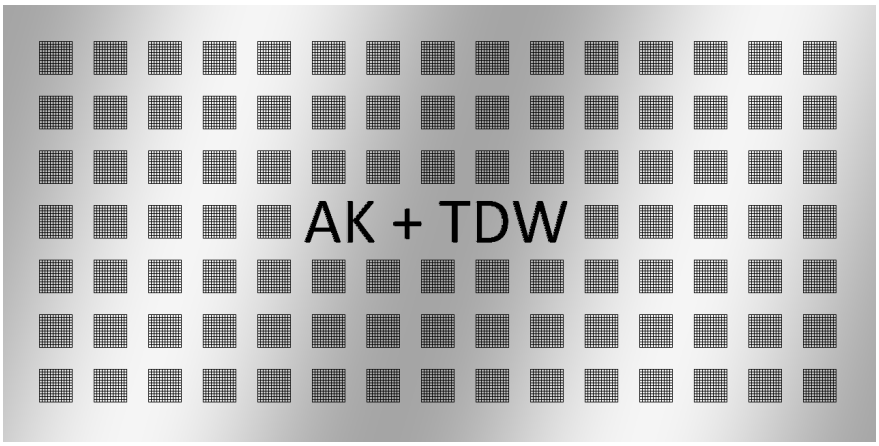
(d) An uncorrected region



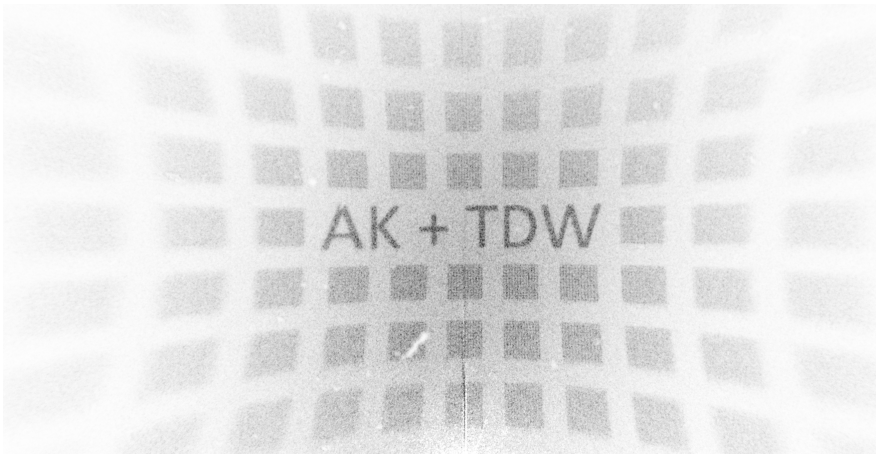
(e) A corrected region

Fig. 4.10 Aberration-correcting masks

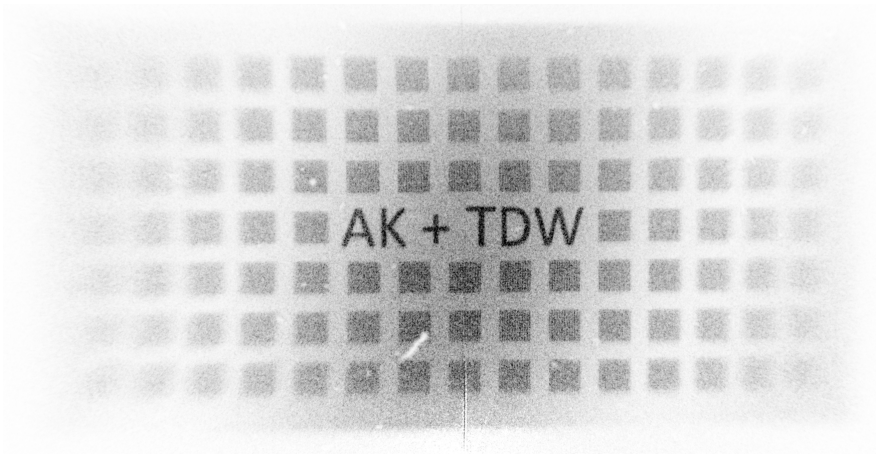
4.6.3 Correction steps



(a) A target test image

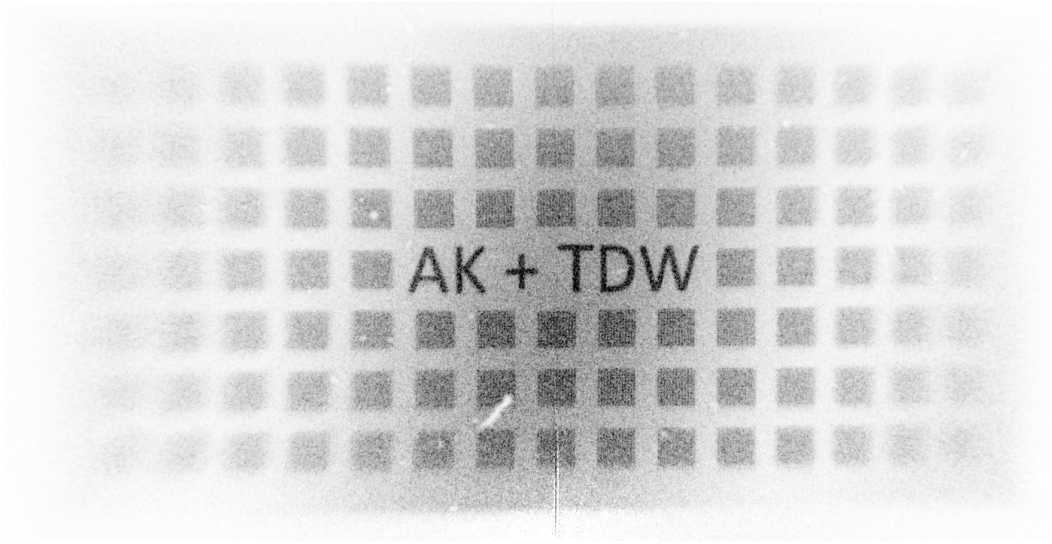


(b) An uncorrected image

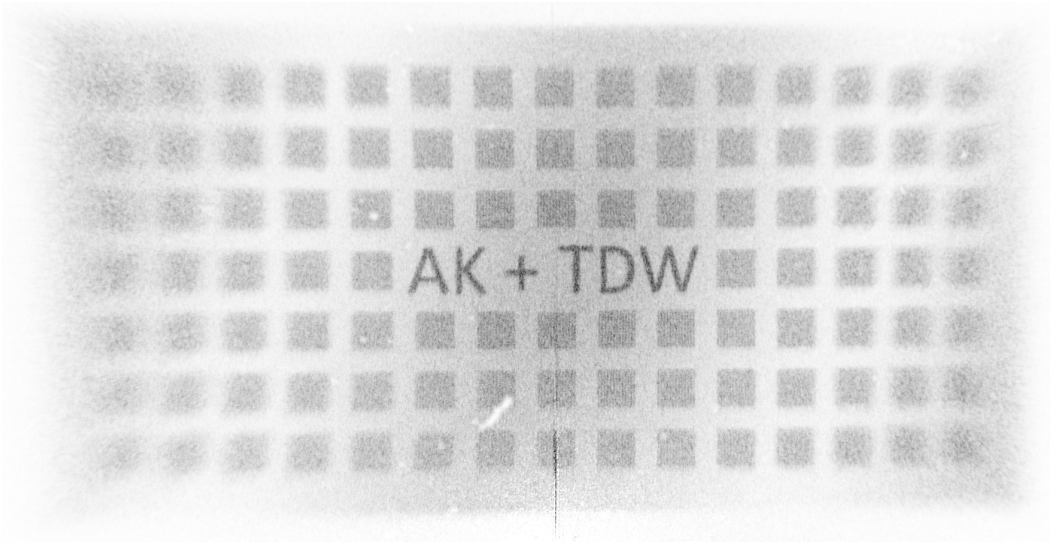


(c) Distortion corrected

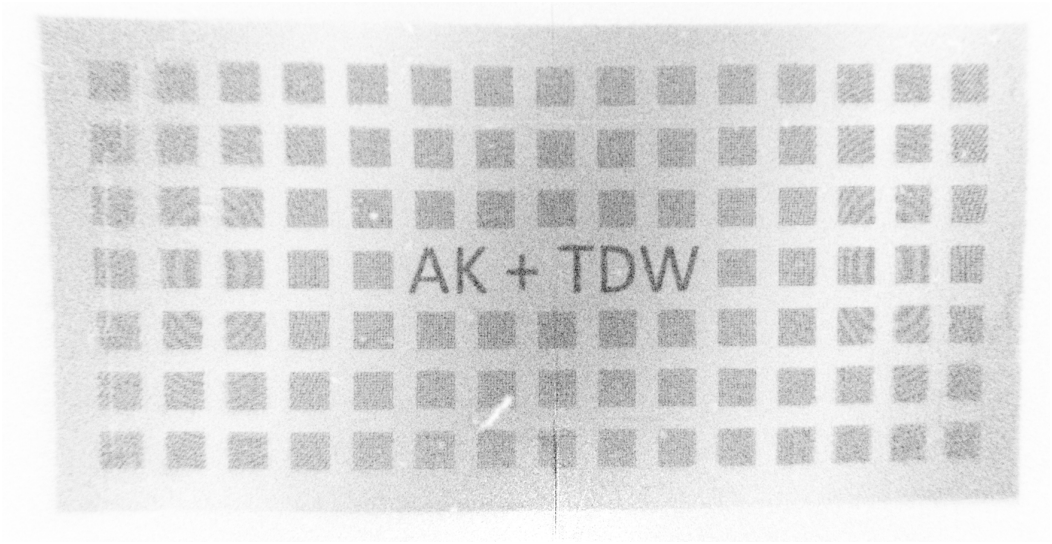
Fig. 4.11 PCOSPR flow at native resolution: distortion correction



(a) Distortion corrected



(b) Distortion and intensity error corrected



(c) A full correction

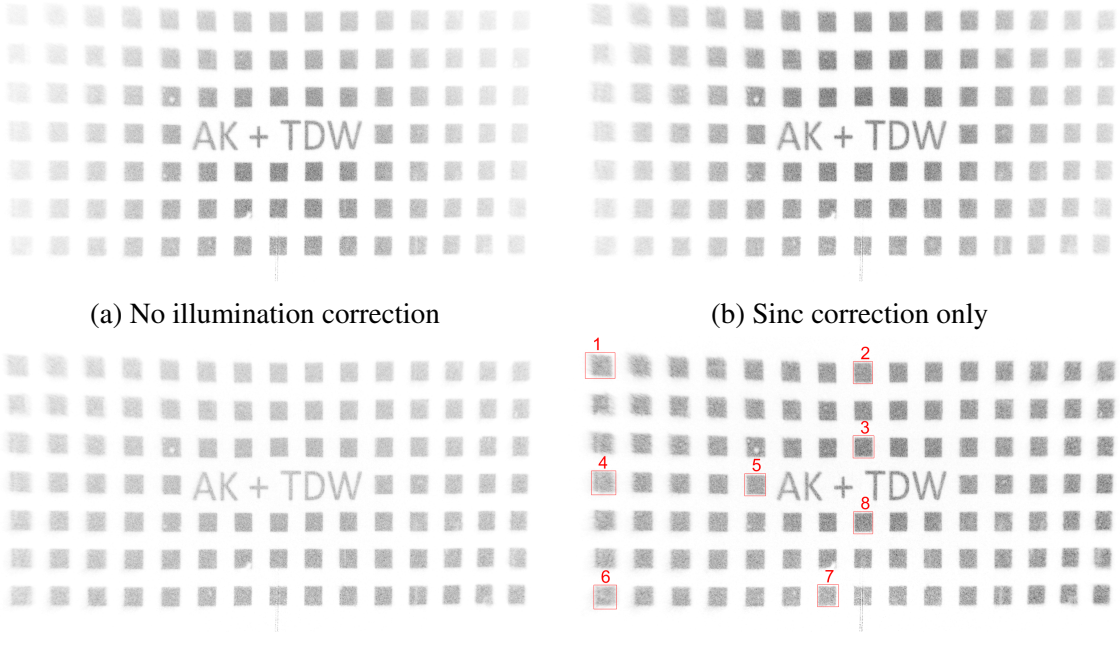
Fig. 4.12 PCOSPR flow at native resolution: illumination and aberration correction

In Fig. 4.11 the result of a distortion correction is presented. The test image seen in Fig. 4.11a was designed specifically to test the noise and resolution properties of the projector. It can be seen that certain degree of error remains [Fig. 4.11], especially in the top centre of the image, indicating that the assumption about the circular symmetry of the projector begins to break down.

The images seen in Fig. 4.12 show further two steps of the correction process: the image intensity correction [Fig. 4.12b], as well as the piecewise aberration correction [Fig. 4.12c]. Each of the corrections discussed is superimposed on the top of all previous ones (as indicated by the flowchart in Fig. 4.8). It can be seen that in the final image [Fig. 4.12c] the majority of errors are eliminated.

4.6.4 Intensity correction

Two contributions to the intensity error in the case of OSPR-type algorithm were postulated. One of them coming from the sinc envelope of the replay field. Here, the prediction will be compared to the experimental findings.



(a) No illumination correction

(b) Sinc correction only

(c) Dist correction only

(d) Both corrections

Fig. 4.13 Intensity correction assessment

In Fig. 4.13, different intensity corrections are compared: an image with no correction applied [Fig. 4.13a], sinc correction only [Fig. 4.13b], distortion correction only [Fig. 4.13c] and both corrections combined [Fig. 4.13d]. To assess the uniformity of the illumination,

Table 4.1 Intensity attenuation correction

	Average intensity of a square (arb. units)								Average	STD
	1	2	3	4	5	6	7	8		
No correction	1.00	6.80	9.94	1.87	7.98	1.66	8.43	12.36	6.25	4.26
Only sinc	1.69	9.20	10.02	2.28	7.86	1.73	5.97	10.12	6.11	3.73
Only dist	3.00	6.72	7.97	4.66	7.88	5.03	8.50	9.87	6.70	2.30
Full correction	4.36	7.97	8.57	5.21	6.84	4.83	5.35	8.52	6.46	1.73

a number of squares was selected and have their total intensity calculated. The results can be seen in Table 4.1. It can be seen that indeed, the fully-corrected image shows the highest uniformity among all the corrections by having the smallest standard deviation of the intensity (1.73) among all of the corrections. This result therefore confirms the prediction made previously.

4.6.5 Adaptive PC-OSPR

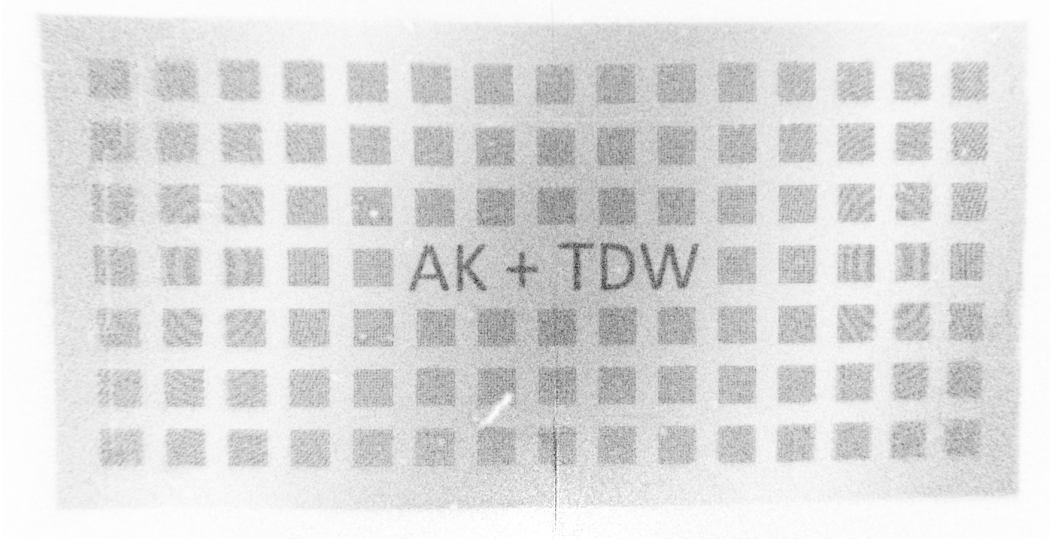
The same corrections were applied in the case of the Adaptive version of PC-OSPR algorithm. However, not to make this thesis unpleasantly long, only the comparison of the final images is presented in Fig. 4.14. It can be seen that the Adaptive version of this algorithm does improve the image. However, this improvement is not as significant as it might be predicted by the theory, because of the significant speckle of the laser and the imperfect aberration correction in the corners of the image.

4.6.6 Resolution improvement

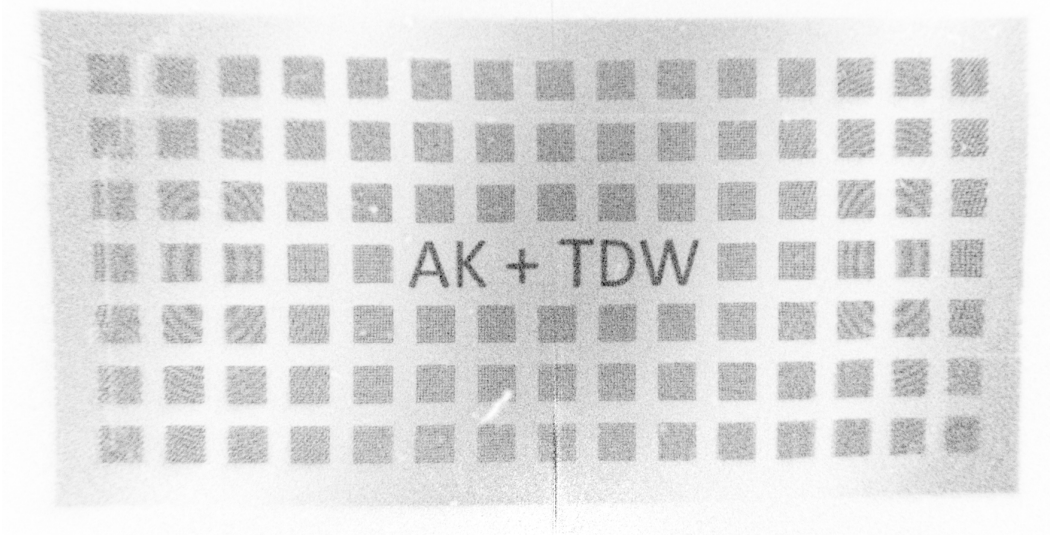
In Fig. 4.14 the resolution comparison is presented. It can be seen that the results largely match the simulations [Fig. 4.7]. Again, the resolution improvement in the corners cannot be clearly visible, because of the imperfect aberration correction and significant speckle.

4.6.7 Performance on a real-life image

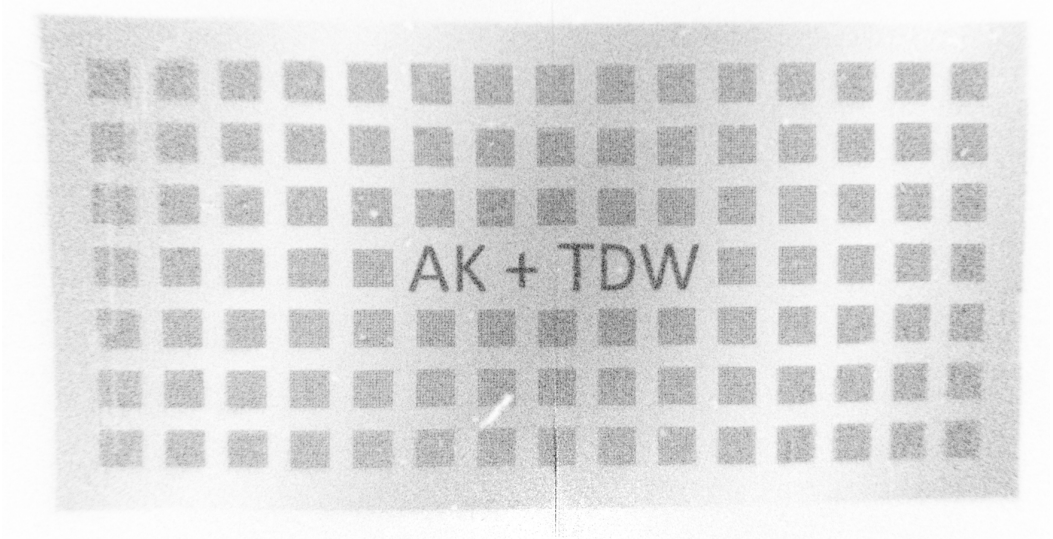
The properties of the algorithm have already been demonstrated. However, the purpose of a holographic projector is to project real images. This is demonstrated in Fig. 4.15. It can be seen that indeed the image is very well corrected. The slight blurring of the image in the corners, although visible in the test image, does not significantly degrade the output image.



(a) A full PC-OSPR correction



(b) An adaptive PC-OSPR correction



(c) An adaptive PC-OSPR correction at a doubled resolution

Fig. 4.14 PC-OSPR - AdPC-OSPR - AdPC-OSPR at a double resolution: comparison

(a) An uncorrected frame

(b) PC-OSPR at a native resolution

(c) Adaptive OSPR at a doubled resolution

Fig. 4.15 Real life performance of the PC-OSPR algorithm

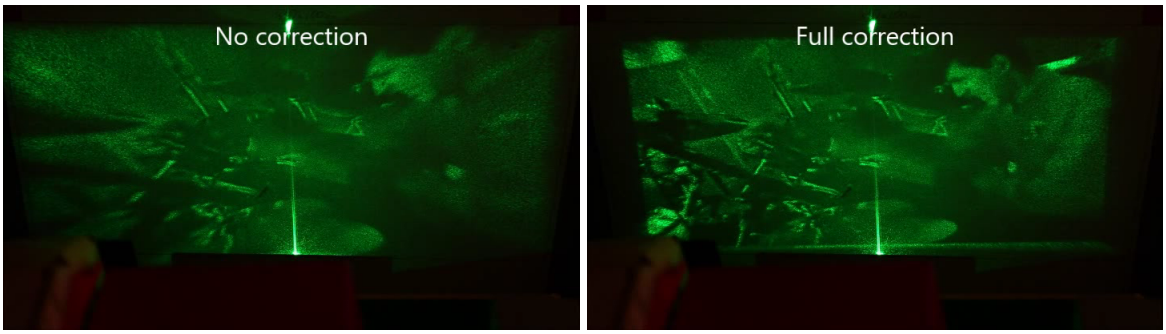
4.7 Real-Time Operation

The algorithm has also been rewritten using general purpose Graphical Processing Unit programming (GP-GPU) [77]. Various software optimizations were implemented to fully utilize the parallel processing capabilities of the GPU. It has to be decided, what framework to use in order to display the image on the SLM. The OpenGL [78] was chosen, because of its simple interoperability with CUDA. Initially, OpenGL utility toolkit (GLUT [79]) was used as a wrapper for creating a window. However, GLUT library did not allow enough control over the window to display it on a specific monitor, therefore the program was rewritten using the GLFW library [80]. It proved slightly slower, but had enough flexibility to achieve everything that was necessary for the purpose of this project.

The final version of the algorithm is presented below. Certain adjustments were made to increase the execution speed (such as moving various pieces of the code outside of the main loop).

- Initialize GLFW library, create a window on a secondary monitor
- Initialize screenshot routines
- Set initial values of the modular correction switches
- Load predefined correction information from a file
- Precalculate phase masks from Zernike coefficients
- Main program loop:
  - Take a screenshot of a specified monitor
  - Apply gamma correction
  - If distortion correction flag is set, correct distortion using a linear approximation
  - If aberration correction flag is set, correct aberrations
  - Display an image on the SLM
  - Process events, read the keyboard input and adjust correction flags accordingly. If ESC button has been pressed - exit the loop.
- Cleanup the memory
- Destroy the window

Executing the above set of operations allowed to create a prototype of a dynamic, real-time holographic projector employing a general-purpose PC with a mid-range GPU (nVidia GTX 760). The prototype was able to achieve a frame-rate of up to 12FPS. The results were reported in [81] and two frames from the real-time hologram generation can be seen in Fig. 4.16.



(a) Movie frame, uncorrected RPF (b) Movie frame, corrected RPF

Fig. 4.16 Real-time PC-OSPR YouTube stream of Dire Straits, the best band that ever existed[1]

4.8 Conclusions

This chapter presents and summarises Piecewise-Corrected One-Step Phase Retrieval algorithm. Its main advantage lies in the fact that it combines the correcting power of Pixel-To-Wrapped Phase Summation (PWPS) with the hologram generation speed of One-Step Phase Retrieval (OSPR).

Distortion correction, discussed first, is performed in a different way in PWPS- and OSPR-type algorithms. In PWPS, it is in practice a non-uniform sampling operation. OSPR only allows an uniform sampling grid of a Fourier Transform. The PC-OSPR algorithm then has to conform to a regular sampling grid by pre-distorting the image before it is passed to the Fourier Transform operation.

A severe distortion influences the output intensity of the image. Whenever the output image is stretched onto a greater surface area, the image intensity drops. Fortunately, the algorithm already corrects for the uneven intensity distribution coming from the pixel shape. This additional contribution, coming from diffraction, can easily be quantified and added to the pixel shape correction. The study of the RPF with different intensity masks, proven that indeed, the correction coming from both contributions give the most even replay field intensity.

Aberration correction is the crucial element of the algorithm. In PWPS, every single point is corrected independently which always ensures precise elimination of aberrations. In reality, the variation of aberrations is a smooth, slowly-varying function. If we assume that aberrations are constant within some very small region, we can represent the summation

[1] According to some sources

of PWPS with the series of Fourier Transforms with a phase correction. We can imagine a situation when every single point is assign its own aberration-correcting regions, the holograms generated with this PC-OSPR algorithm are going to be identical to the ones generated using PWPS. In reality, these regions can be made much bigger. PC-OSPR is then only an approximate solution to PWPS.

By appropriately choosing the number of regions and their size, it is possible to eliminate the aberrations to the extent that the imperfections are too small to be noticed by a human eye. In the projector presented here, 6 such regions have been proven successful to generate high-quality images. By increasing the number of regions, the quality can be improved, but the time of hologram generation will grow linearly with the number of regions.

The next problem originates from the severe distortion of the system. While pre-distorting the image, some of the finer details are lost towards the edges leading to the loss of resolution. To solve this, we proposed increasing the resolution of the Fourier Transform operation. This method proved to work well in eliminating artefacts like Moire fringes from the image, but slightly increased the noise level and doubled the computation time.

To further increase the image quality, a technique termed Adaptive-OSPR introduced previously by Cable [40] is investigated. We found that it only slightly increases the quality at a highly increased computational time, and hence, not suitable for real-time applications.

All of these techniques are then implemented using GP-GPU programming in order to achieve a real-time hologram generation. Given a highly-optimized implementation, it was possible to run the aforementioned algorithm on a mid-range GPU (GTX 760) at a frame-rate of 12FPS. The quality of the corrected image was highly improved showing no significant errors.

Chapter 5

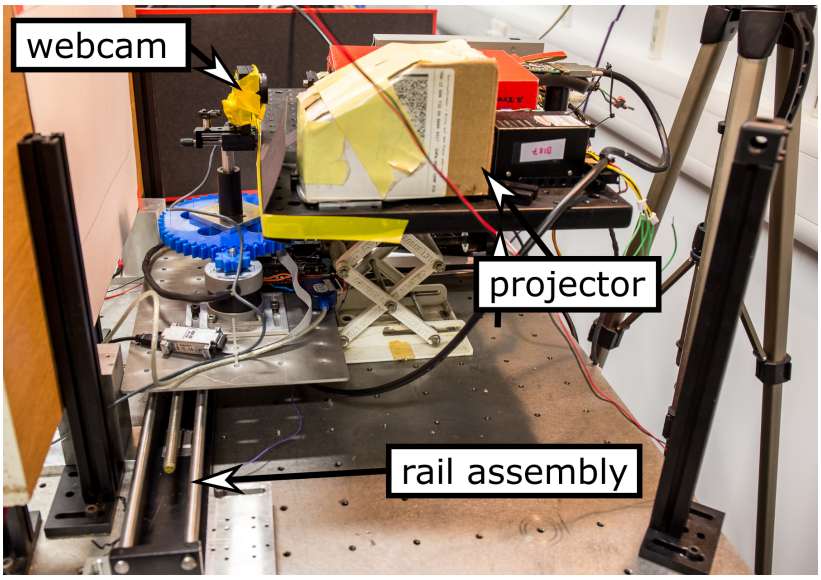
An automated testbed for factory-assembled holographic projectors

In the previous chapter, ways of correcting various errors in holographic projectors were presented, including distortion, aberration and image intensity error. The aim of this chapter is to establish a set of methods to characterize these errors, so they can be corrected using the presented formalism. These three chapters attempt to construct an automated testbed for holographic projectors. Every single projector can be placed on such a testbed that will characterize all of its imperfections and assign a set of correction information. This information will be encoded into non-volatile memory of each projector, allowing it to display highest-quality image throughout its lifetime.

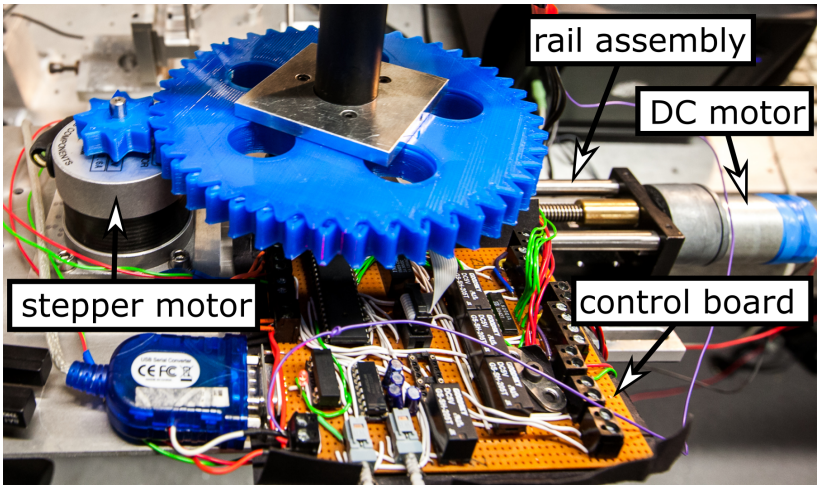
5.1 Experimental setup

The setup used in characterization of the projector is a composition of two separate setups. Setup I is identical to the one presented in Chapter 3. It is used only for aberration correction, whenever a magnified view of a specific part of the replay field is needed. A webcam provides a relatively small resolution ($640px \times 480px$), but is able to acquire images rapidly (15 FPS). Depending on the scaling, that area corresponds to 5-10 pixels of the RPF in the horizontal direction. The second part of the setup, identical to the one developed in [65] is used to capture a full replay field with a dSLR. The resolution of a single image is much higher ($15-20Mpix$), but the time to acquire, transfer and process the image is much longer. Depending on the laser intensity, it can be as long as several seconds.

5.1.1 Setup I



(a) Overview of Setup I



(b) ATmega microcontroller board

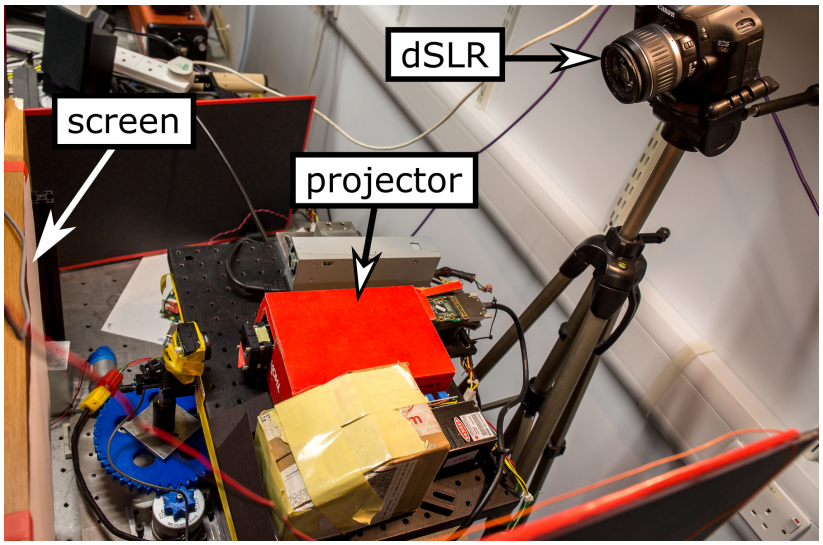
Fig. 5.1 Setup I

The setup from Chapter 3 was adapted for the spatially-varying case by placing the webcam on mobile rails, seen in Fig. 5.1a. This way, the webcam, instead of facing the optical centre of the projector, can now be moved to face multiple positions in the RPF. To facilitate the rotation of the camera's head, a stepper motor connected to the 3D printed gearbox was constructed [Fig. 5.1b].

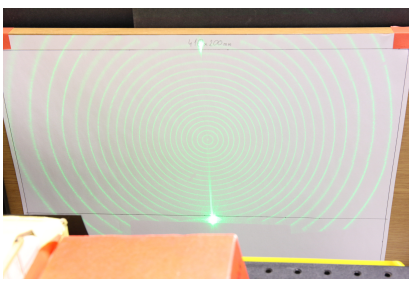
An entire mechanism is controlled by a custom-made ATmega microcontroller board [Fig. 5.1b], which applies voltage of right polarization to the DC motor and drives the stepper

motor. The current setup can only measure and characterize points along a single axis. The movement in the second axis is currently achieved by the manual placing of the camera. This step can easily be automated whenever more funds are available.

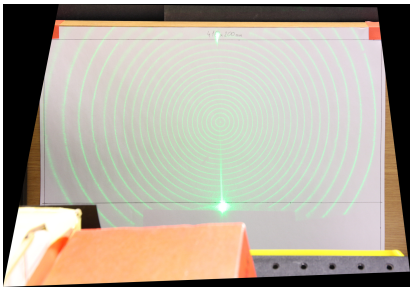
5.1.2 Setup II



(a) Setup II



(b) Image acquired by the camera



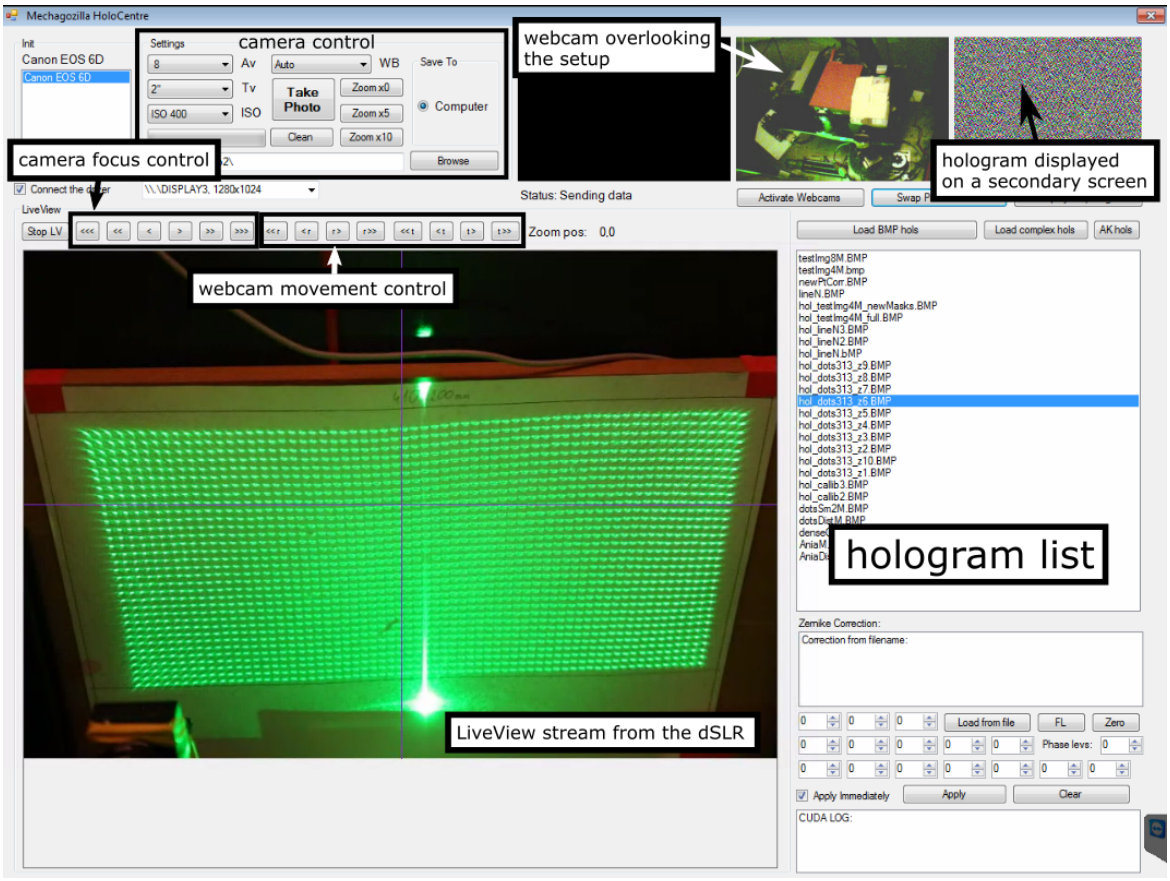
(c) Image after callibration

Fig. 5.2 Setup II

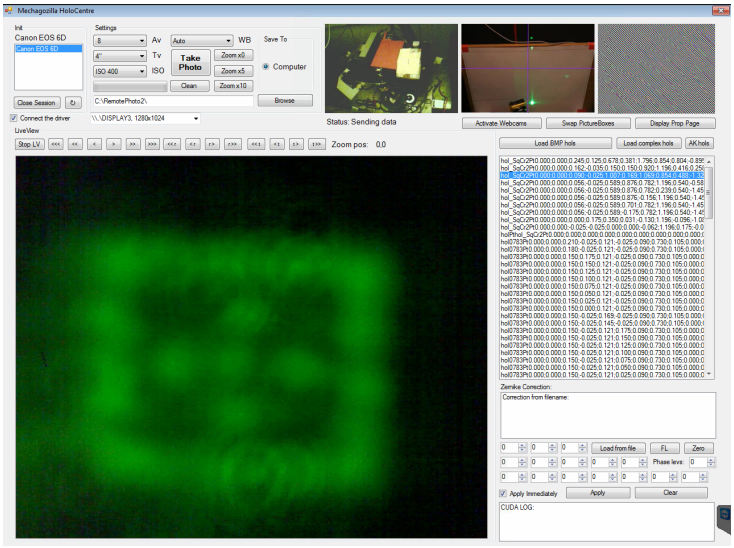
The second setup is identical to the one used in [65]. The projector is placed in front of a flat screen, which is overlooked by the dSLR camera [Fig. 5.2]. Because of the camera's placement, the image acquired exhibits a noticeable parallax error. In order to work around this limitation, a calibration method was designed.

A dark rectangle of known dimensionality, which bounds the replay field was drawn on the screen. The algorithm then recognizes the sides of the rectangle and calculates an appropriate projective transformation to get back the real image displayed by the projector. This process is illustrated in Figs. 5.2b - 5.2c.

5.1.3 Remote control capabilities of the setup



(a) Setup II configuration



(b) Setup I configuration

Fig. 5.3 Remote control software

All of the experiments can be fully controlled remotely from any location using TeamViewer [82]. Several pieces of software were developed for the purposes of monitoring and remote diagnosis. For instance, a program with a working name of "Mechagozilla HoloCentre", which is an expanded version of "Canon SDK Tutorial" [83], can simultaneously control the webcam movement, a dSLR camera, hologram display and overlook the setup from multiple webcams. The main components of a program are indicated in Fig. 5.3a. The user can swap the photoboxes to enlarge a given stream: LiveView stream from the dSLR and webcams.

The two setups can be used interchangeably: whenever Setup I has to be used, the webcam is automatically moved to face the projector [Fig. 5.3b]. When Setup II needs to be used, the camera is moved away, allowing the dSLR to capture the entire screen [Fig. 5.3a].

5.2 Distortion measurement

The majority of optical systems are cylindrically symmetric, as is the projector used in this work. For this type of distortion, the dependence of the real radius versus paraxial radius is sufficient to characterize and correct it.

Accurate distortion measurement is the key to a precise correction. Various researchers corrected the distortion of a lens by calibrating it, based on a rectangular grid [48, 84]. Here, an even more straight-forward method is proposed. With the assumption about spherical symmetry of the field, the effective method to measure the distortion is to display a set of concentric rings around the optical centre. Assuming that the camera is well calibrated, the real radii of the rings can be retrieved.

In the input image seen in Fig. 5.4a rings with radii in steps of $20px$ were displayed. In the replay field (seen in Fig. 5.4b), 24 rings were recorded. The centre of the replay field is selected manually. For each angle from 1 to 360 degrees, the program radially samples the pixels in the image, retrieving the intensity at a particular position along the line.

A sample measurement of this type is presented in Fig. 5.4c, which corresponds to the red line in the Fig. 5.4b. This curve is then smoothened with a Gaussian kernel and the found maxima correspond to rings' positions.

A set of 360 measurements for all angles is then seen in Fig. 5.4d. These measurements are then averaged out and fitted to a curve of the type:

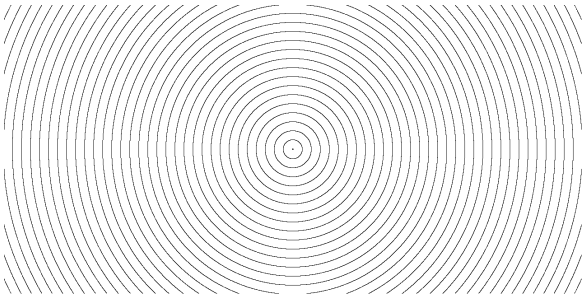
$$r_{cam}(r_{img}) = p_4 {r_{img}}^4 + p_3 {r_{img}}^3 + p_2 {r_{img}}^2 + p_1 r_{img}$$

The measured positions r_{cam} are in camera coordinates, and, to translate them to replay field coordinates, one employs the assumption that the curve is tangential to $r' = r$ as $r \to 0$.

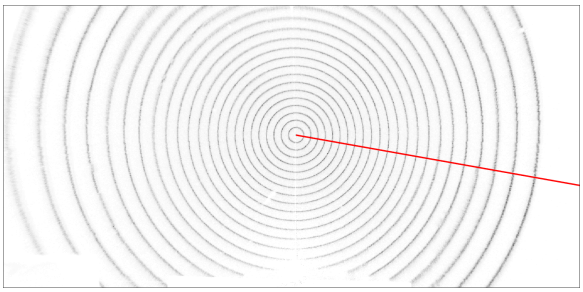
Therefore, the rescaled distortion curve in RPF coordinates will be of a form:

$$r'(r) = \frac{p_4}{p_1} r^4 + \frac{p_3}{p_1} r^3 + \frac{p_2}{p_1} r^2 + r$$

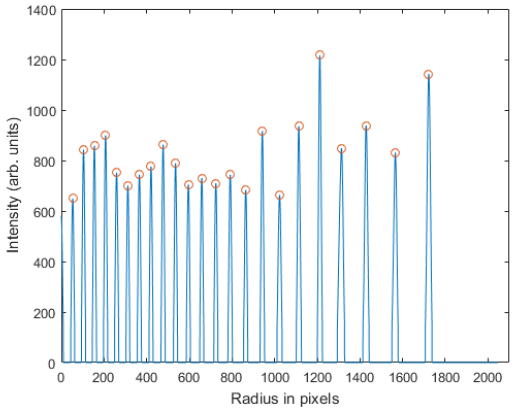
This curve contains all the information about the distortion of the projector and is sufficient to perform a correction.



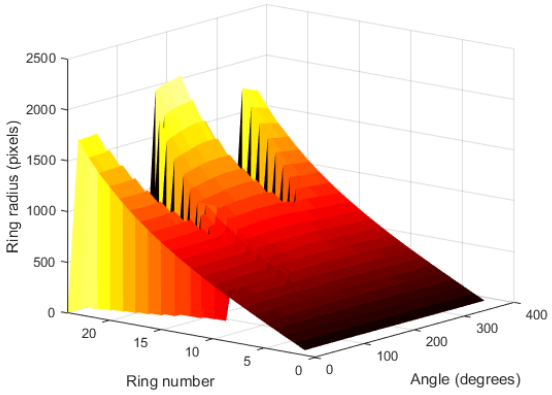
(a) Target image used to measure distortion



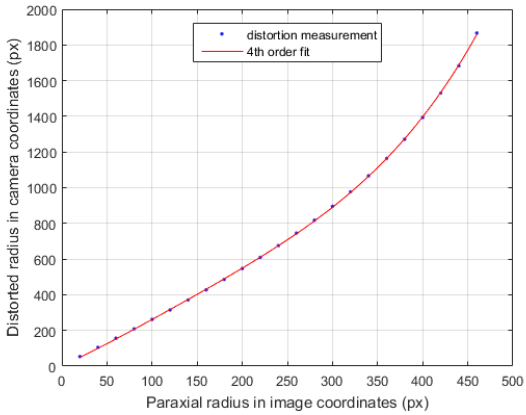
(b) Distorted replay field recorded by a camera



(c) Distortion of the projector along a single angle



(d) Distortion for all of the angles



(e) Fitting experimental points into a curve

Fig. 5.4 Distortion correction

5.3 Aberration Correction

5.3.1 Different ways of obtaining the aberration parameters

Zemax simulation

One way of obtaining the aberration correction parameters is to simulate the system using a ray-tracing software (such as ZEMAX [54]). Freeman described this procedure in [56, 57]. The source of light is first expanded and collimated by lenses in telescope configuration. Then it is modulated by the SLM (either in transmissive or reflective configuration). After the SLM, the reverse telescope is used to expand the field of view. Fig. 5.5 shows an example of such modelled system.

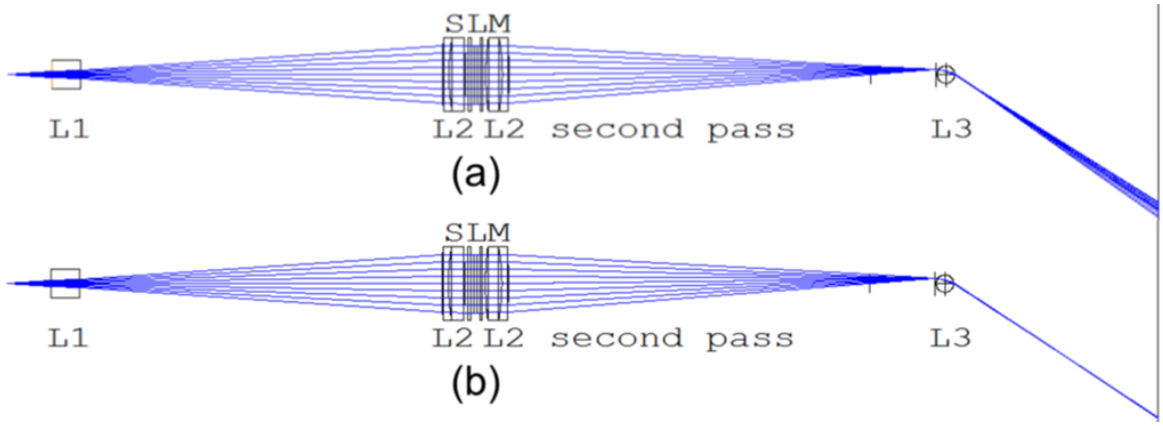


Fig. 5.5 Aberration correction of a holographic projection simulated in ZEMAX
(a) A projector with no correction applied (b) A corrected projector
L1, L2 and L3 indicate lenses [1].

In ZEMAX, an SLM is modelled as a grating with variable pitch. The aberration correction is effectively a Zernike Fringe Surface directly on the top of that grating. Different field positions can be achieved by changing the pitch of the grating. The lenses can be inserted into such model either by defining optical surfaces' parameters , or even more easily, by inserting a lens model from a file. Most of the lens manufacturers include their designs in the format that can straight-forwardly be inserted into the ZEMAX model.

It has to be noted that this model represents sequential ray-tracing. As the light beam reflects off the SLM and passes through the lens L2 in the opposite direction, this is indicated in the model by another lens, which is a mirror image of L2 (indicated in Fig. 5.5 as "L2 second pass"). The model can be made more realistic, by inserting a mirror in the place of

[1] The presented model was largely influenced by a previous ZEMAX design, constructed by Freeman [56]

the SLM, then the two lenses would ideally overlap. For the matter of clarity, however, this procedure was not used.

For each position, the Zernike coefficients can be set as an optimization target. The global optimization tool then chooses such configuration of the coefficients that minimize the spot size. The result of this procedure is the set of coefficients, which can be interpolated, based on continuity for every field position. Within the PWPS routine, this is exactly needed to calculate a corrected hologram. For every pixel, the corrective phase mask is calculated using the summation from Eq. 2.3 and added to the overall contribution. Within PC-OSPR, a set of discrete regions has to be calculated.

Adaptive-optical correction

Adaptive-optical correction is naturally quantized. Hence, the procedure of obtaining aberration-correcting phase masks as well as correction regions is different from ZEMAX.

The correction is performed by applying procedures described in Chapter 3. Two methods are presented there: a robust Hybrid Algorithm (HGA) and a simplified Heuristic Steepest Descent (HD). The first is very time consuming, but provides best correction, while the second one is much faster, but only finds the global optimum when it is reasonably close to the starting point.

First, the correction is performed on the point at the optical centre. To estimate, where the next correction should be found, a grid of pixels is displayed with a given correction. The next point is then chosen as the one for which the points are not any more sharp due to the spatial variation of aberrations. The same procedure continues until an entire replay field is reasonably well corrected.

The characterization of the projector is an iterative process, where with every iteration further improves the optical quality. Initially, a lengthy procedure has to be employed. Later, the results are refined. In the refinement process, an algorithm already has a set starting point, which is the current correction. Therefore, instead of using the time-consuming HGA, HD can be used instead.

5.4 Assignment of the aberration regions

In principle, the aberration regions can be assigned for the ZEMAX correction, as well as the adaptive-optical correction, as it was demonstrated in [75]. However, previous experiments with PWPS method proven that for the projector II, coefficients obtained from ZEMAX are not any more optimal, since the projector's construction is far from ideal [81]. Therefore, the

region assignment for ZEMAX, although in principle possible, is omitted here, as it will not lead to a well-corrected image.

The result of the feedback loop is already a discrete number of phase masks. The task is then to find, for every single RPF position, the particular phase mask among a given set, that minimizes the aberrations. The algorithm needs to be designed in a way to be sufficiently error-resistant and to make minimum assumptions about projected image. For clarity of the explanation, this complicated algorithm is split into a set of smaller subtasks:

- Capture the RPF of a grid of single points with each correction mask applied
- Recognize all of the points, assign a fit function value to each
- Map the points at the same RPF positions onto each other
- For each of the mapped points, decide which phase mask minimizes the fit function
- Recognize the replay field position of each point
- Create a set of index tables that indicate for every RPF position, which mask minimizes the aberrations
- Detect and correct errors based on continuity

For a matter of precision, the projected grid was chosen to be regular in the screen domain (and, because of distortion, irregular in the hologram domain). For that precise reason, the distortion correction step has to be applied first.

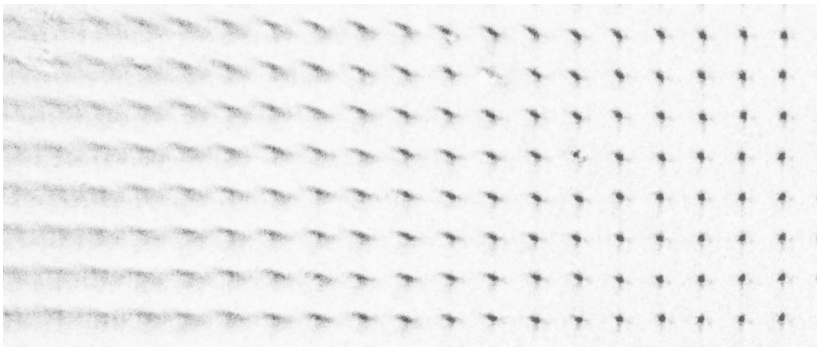
5.4.1 Image preparation and processing

To maximize the effectiveness of the algorithm, the image has to be sufficiently high quality and free of artefacts. To ensure this is the case, target images are generated using 24-frame OSPR to provide sufficient elimination of speckle. The camera aperture is set such that an entire screen remains in focus. The exposure is set such that there isn't any clipping in RAW format. Once the image is acquired, a following processing is done:

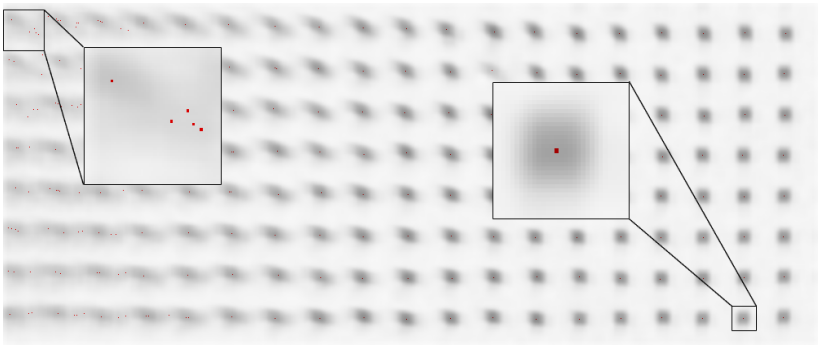
- Image is developed using DCraw, an open-source software for developing RAW files [85, 86], into a 16-bit TIFF
- Background frame(a picture of the screen without any hologram displayed) is subtracted to eliminate the artefacts, such as the zero order and various reflections
- Image is demosaiced, only leaving pixels corresponding to green colour
- Appropriate projective transformation is applied
- Image is cropped and rotated

All of these steps have been automated. Only the image crop has to be selected manually first, before it is applied to all of the images in a batch.

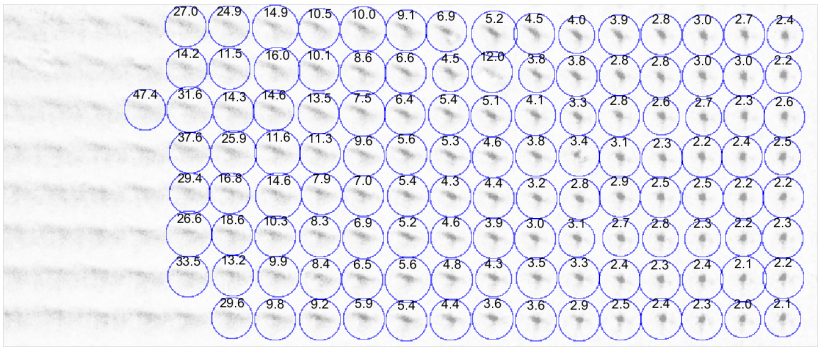
5.4.2 Single point recognition



(a) A part of the callibration image



(b) Points smoothened with a Gaussian kernel with the local maxima of the image found



(c) A set of final points recognized by the algorithm

Fig. 5.6 A mechanism recognizing and rating the points

To precisely recognize the position of each of the single points in the RPF, the image is first convolved with a Gaussian kernel with a reasonably large radius. After this process, the structure of each point is smeared into a large blob. The local maxima of that smeared image proven to coincide quite well with the centre of the pattern. This procedure is demonstrated in Fig. 5.6. Fig. 5.6a shows a part of the calibration image. The convolution of the image and the Gaussian filter is seen in Fig. 5.6b. Insets show the positions of the local maxima. The right inset shows a decently well-corrected point with a single red spot indicating a found

local maximum. The left inset, on the other hand shows a blurry point. In this case, multiple local maxima are found, which do not necessarily coincide with point's structure.

The procedure of eliminating false positives is combined with finding a boundary of each point. The appropriate region of interest will be the one that fully contains a point and has very small intensity on the outer edges. The flow of the algorithm is presented in Algorithm 6

Algorithm 6: Point recognition algorithm

I : Input image
(x_0, y_0) : Position of a found local maximum
rMin : The output radius of the point

```
1  maxI ← I(x0, y0)
2  for rq ← 1 to 100 do
3  |   currentI = MaxIntensity(rq, rq +3) ;
4  |   if currentI < maxI then
5  |   |   maxI ← currentI;
6  |   else
7  |   |   if rq > 10 and currentI < maxI × 1/10 then
8  |   |   |   rMin ← rq;
9  |   |   |   Point found
10 |   |   else
11 |   |   |   Noise found
12 |   |   end
13 |   end
14 end
```

The working of this algorithm is presented in Fig. 5.7. The function *MaxIntensity* first separates points that are contained within the ring of radii $(r_q, r_q + 3)$ and then calculates the maximum intensity within that region. One ring of this type for $r_q = 10$ is seen in Fig. 5.7b. For every value of $r_q = 1px \ldots 100px$, the algorithm calculates the maximum intensity within a ring, which can be seen in Fig. 5.7c attempting to look for such a value of r_q where the intensity stops decreasing. In the case of the mentioned measurement, this value is found to be 30 and the assigned boundary of the point can be seen in Fig. 5.7d.

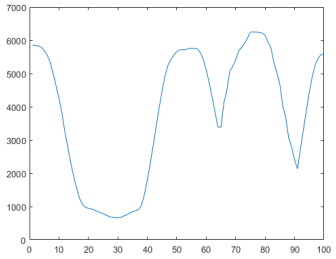
On the other hand, the point presented in Fig. 5.7e is a false positive. The local maximum indicated by MatLAB does not correspond to the highest intensity within the point, which can be seen in Fig. 5.7f. This point is discarded as a false positive and not processed any further.

The same procedure is carried out for every local maximum found in the image. Each point, which is not ignored as a false positive gets its fitness calculated using the procedure described in Chapter 3.

As an output, the position of the point, the radius, and the FF value is recorded in the array. A set of recognized points is presented in Fig. 5.6c together with an assigned boundary and its fitness.

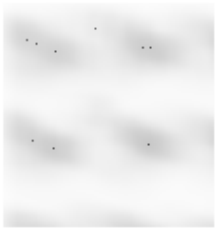
(a) Point 1: image cropped around a given local maximum

(b) Point 1: Single ring of pixels at radius 10

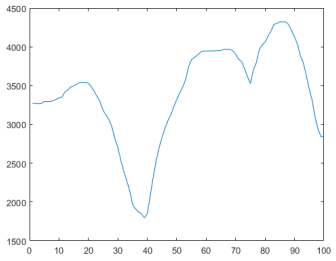


(c) Point 1: Plot of maximum intensity for radii 1 to 100

(d) Point 1: An assigned boundary



(e) Point 2: image cropped around a given local maximum

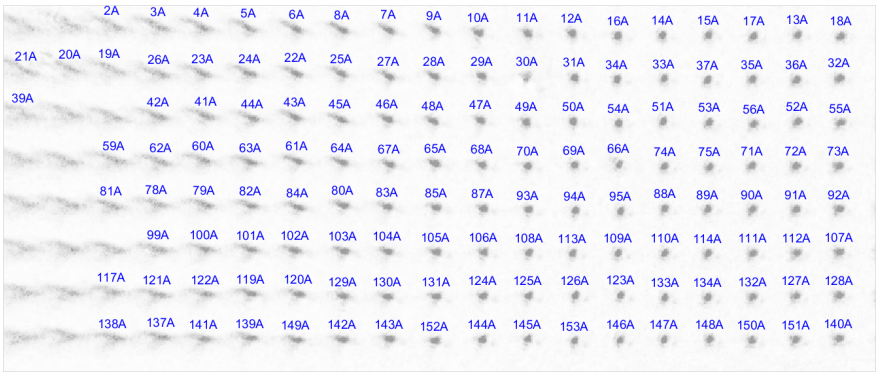


(f) Point 2: Plot of maximum intensity for radii 1 to 100

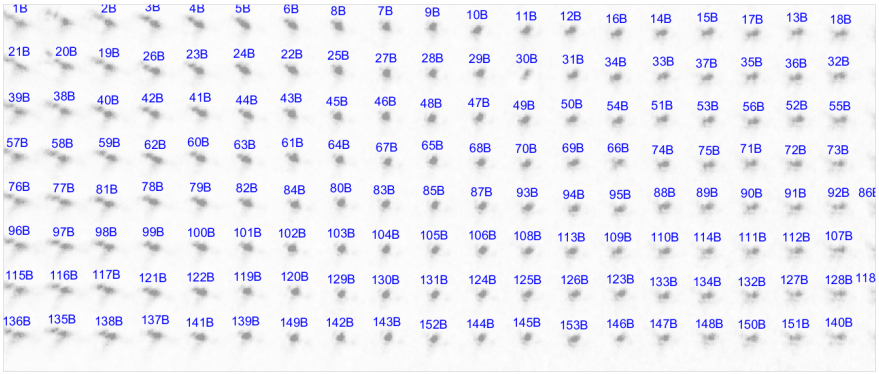
Fig. 5.7 Point boundary recognition

5.4.3 Matching same points from different corrections

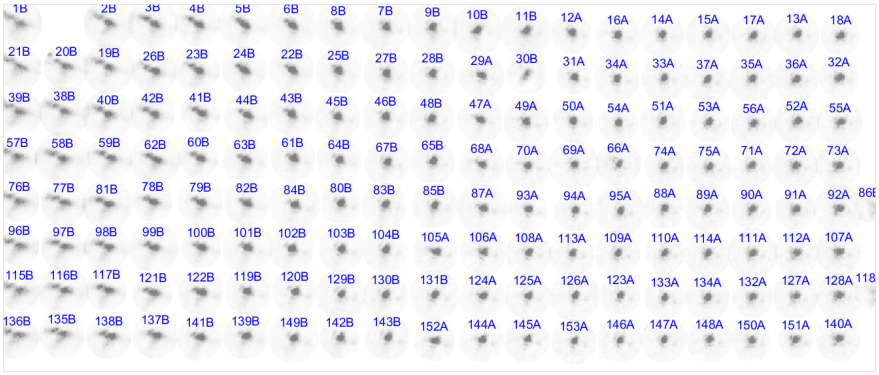
The next step in defining the aberration-correction region is to identify the same points in different images. An additional difficulty is the fact that different aberration-correcting phase masks might slightly change the overall tilt of the image, therefore shifting the points' positions. To work around this problem, the list of all the recognized points from all images is created. A sample point from the image has its distance to all of the points compared. If the closest point found in the list is closer than the average value of the both radii, the correspondence is found. If no point in the list matches a given entry, a new entry, indicating a yet unrecorded point, is created.



(a) Image with a phase mask 1 with all of its points recognized



(b) Image with a phase mask 2 with all of its points recognized

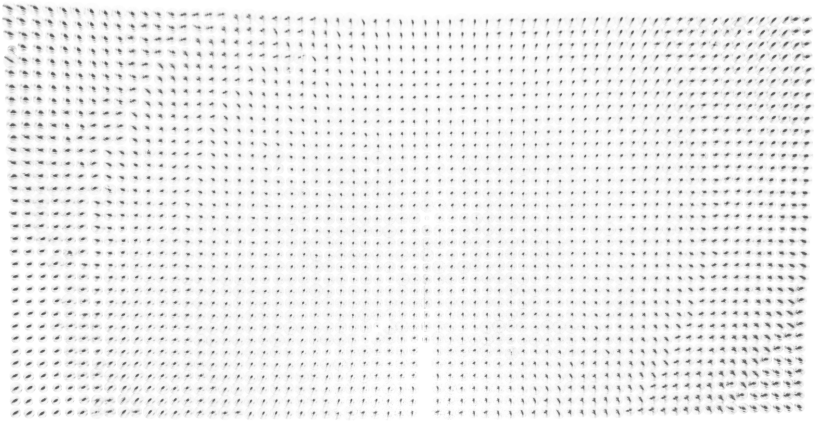


(c) Best points from two images matched onto each other

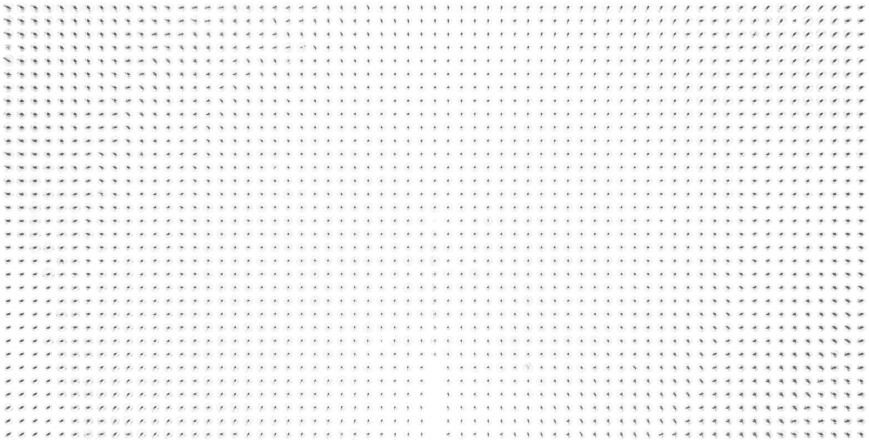
Fig. 5.8 Matching points from different corrections

5.4.4 Finding the correspondence to the RPF coordinates

The previous sections all act on a list of single spots found in the image. In order to construct an appropriate mask, positions of these spots need to be translated from camera coordinates into the RPF coordinates. A human eye can precisely recognize, which points lie on a single line. The computer program need to be taught how to achieve that. The solution proposed here is based on iteratively building a map of single spots, based on its neighbours. This



(a) Best points from all corrections in camera coordinates



(b) Best points from all corrections translated into RPF coordinates

Fig. 5.9 Translating points' coordinates to RPF coordinates

procedure is introduced to account for a non-perfect distortion correction. Although the pixels on the larger scale might suffer from distortion, this effect will almost be negligible on the small scale in the neighbourhood of surrounding spots. The procedure is the following:

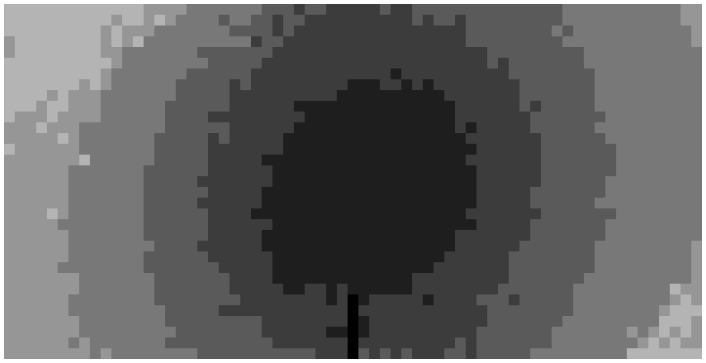
- Manually select the position of the spot at the optical centre
- Find the smallest circular region around a given spot that contains not more than 4 neighbours
- Based on relative positions, recognize spots on the right, left, top and bottom
- If diagonal spots are found - ignore
- Put the found spots in the appropriate place of the spot array
- If a particular point has all of his neighbours found, this position in the array is assigned as complete and will not be processed any further
- For every position in the array that has not been marked as complete, repeat steps 2-7

5.4.5 Error detection and correction

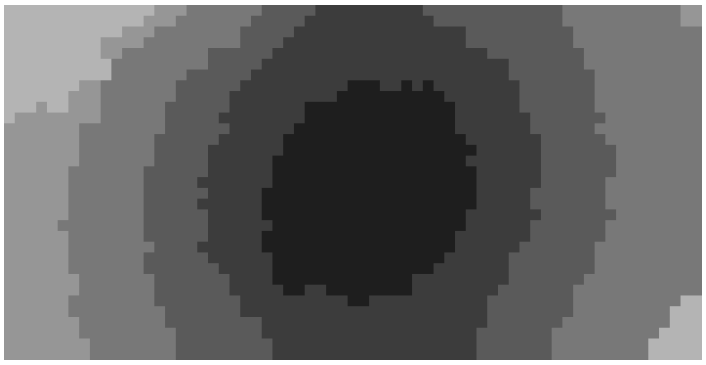
As it can be seen in Fig. 5.10a, the algorithm sometimes struggles with judging the aberrations of points. This effect is most visible at the mask boundaries, where both corrections have comparable quality. In order to fix these errors, we will employ the argument of masks' continuity. Processing points one by one, the algorithm calculates all of the neighbours of a particular point. A few different scenarios are considered:

- A point is surrounded by all points from same correction number - no error detected
- A point is surrounded by all points from a different correction number - an error straight-forward to fix
- A point is surrounded by points having few different correction numbers - a difficult case
- A point does not have an assigned correction number

The algorithm considers each RPF spot and decides, whether the point is at a correct place. If the error is found, the algorithm assigns a new correction number. The procedure is iterative and finishes once no further errors are detected. The resulting fixed masks can be seen in Fig. 5.10.



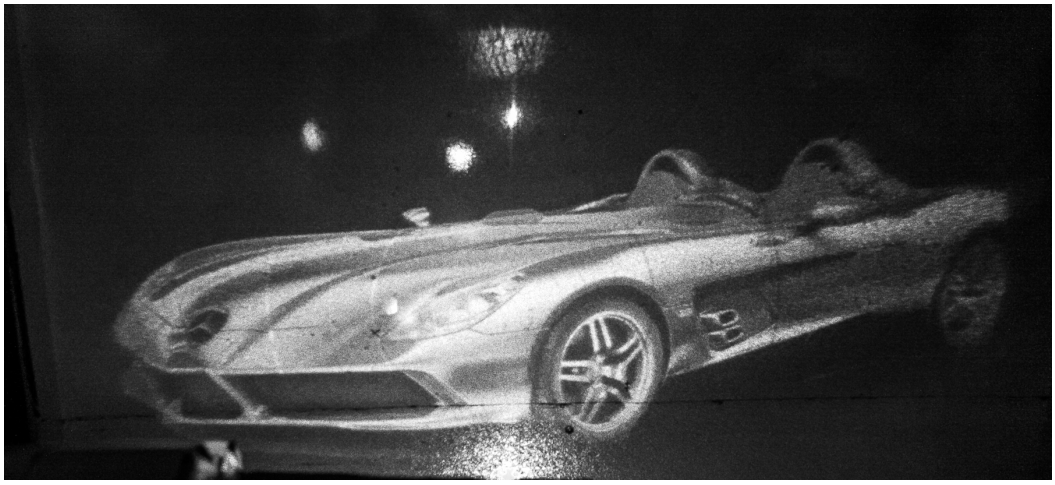
(a) Image of corrected points selected from the images



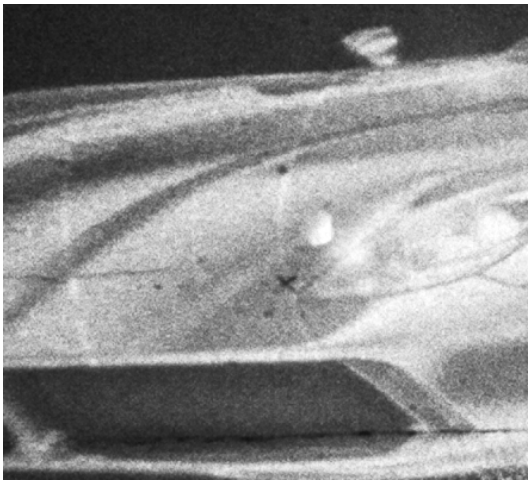
(b) Image of corrected points translated into RPF coordinates

Fig. 5.10 Repairing noisy masks

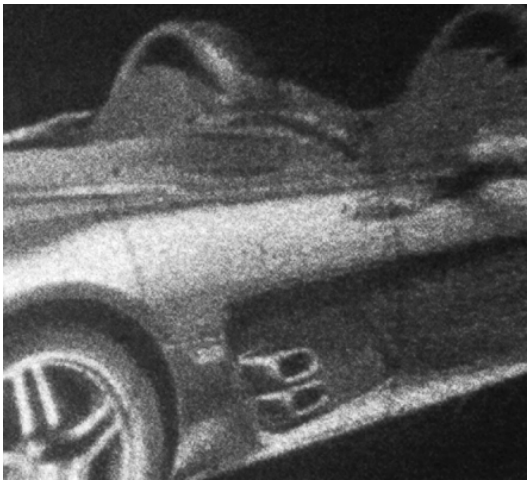
5.5 Tilt correction



(a) Replay field with a tilt mismatch visible



(b) Tilt mismatch: overlapping regions



(c) Tilt mismatch: disjoint regions

Fig. 5.11 Tilt mismatch between the adjacent masks

The phase masks found by the adaptive-optical corrections are not necessary perfect corrections. Rather, they are the best approximations of a perfect correction using a limited number of Zernike Polynomials weighted by a Gaussian illumination profile. Therefore, it sometimes happens that the overall image tip and tilt is different for neighbouring masks. An example of this phenomenon can be seen in Figure 5.11. This error is relatively easy to eliminate by observing a single pixel at the boundary and adding first and second Zernike Polynomials (tip and tilt) to one of the masks. A point at a boundary can be displayed with both of the phase-correcting phase masks. The algorithm then locates the centre of the point (the same

way as described in Chapter 3) and adds a sufficient amounts of tip and tilt to one of the masks until both the points coincide.

5.6 Summary of the corrections

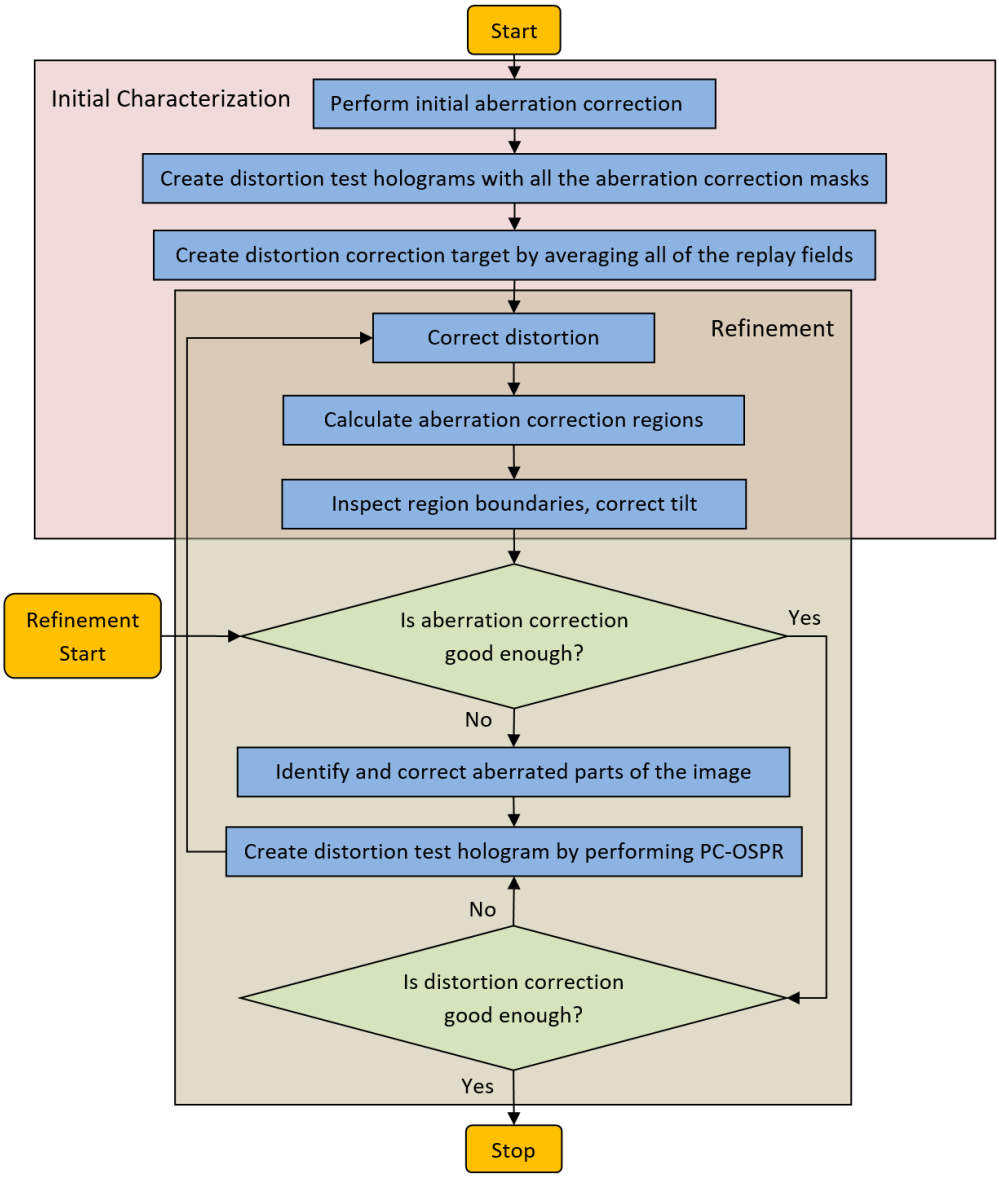


Fig. 5.12 Full correction flowchart

The projectors characterized by the mechanism were found to be relatively sensitive to the aberration changes. Introduction of an additional component into the setup (such as a polariser) or slight change of the SLM tilt resulted in a noticeable change in the projector's

optical performance. Therefore, after every time the projector‘s layout has been changed, it has to be recalibrated for maximum performance.

Whenever a new type of projector needs to be characterized, the lengthy procedure that doesn't make any assumptions about it has to be performed. The procedure is split into few stages, indicated in Fig. 5.12.

First, the aberration correction is performed in a number of RPF positions using the adaptive-optical feedback loop mechanism running a hybrid genetic-steepest descent algorithm (HGA). Having the aberration corrected, the circular distortion target is projected using all of the correcting phase masks. The images are then averaged out in order to initially reduce the image blur. On that target the distortion correction is initially performed. The assignment of aberration-correcting regions is the next and final step.

Once the above correction is complete, its quality is inspected and iteratively improved if necessary. The improvement of the aberration correction is performed by refining existing correction points, and if this proves insufficient, correcting the aberrated regions. The first iteration of the distortion correction will almost certainly be imperfect, since the first distortion target is an average over all of the targets with different corrections. To improve it, a normal PC-OSPR procedure is employed to create a target will be appropriately corrected for aberration. After this procedure is carried on, the accuracy of the correction will improve with every iteration, and eventually, will be sufficient enough for the viewer not to notice any errors.

Whenever a projector has already been calibrated, but its optical layout has changed, by the insertion of an additional component, or a new device with similar properties is being calibrated, the lengthy procedure of initial calibration can be shortened. The current mechanism allows for this by starting the recalibration procedure straight from the refinement step.

5.6.1 Correction output

The output of the projector's correction is the link between the testbed and the algorithm generating holograms. In the Table 5.1, we present a list of variables that are necessary to fully correct the projector. This list is then written into a file and is used by MatLAB as well as CUDA implementations of PC-OSPR routine.

An entire illumination correction for PC-OSPR is contained by the variable $illumCorr$. That includes sinc envelope, distortion illumination correction and any further adjustments.

$dist$ is the distortion curve presented previously. For a matter of simplicity, we calculate the value of the found polynomial for pixel distances $1 \ldots max_r$. The value of max_r needs

Table 5.1 Projector correction information

Variable Name	Type	Size	Explanation
SLMx SLMy	Integer		Size of the SLM
IMGx IMGy	Integer		Size of the input image
illumCorr	Array of floats	(IMGx, IMGy)	Correction factor for image intensity error
dist	Vector of floats	(800)	Distortion curve of a projector
N	Integer		Number of aberration-correcting regions
ACmasks	Array of integers	(IMGx, IMGy)	Array specifying the correction region of every pixel
phaseMasks	Array of floats	(SLMx, SLMy, N)	Corrective phase mask for every region
zernCorr	Array of floats	(15, N)	(optional) Zernike coefficient for every mask

to be greater than the maximum radius of the replay field. For the projector used here, the maximum RPF radius is 715.6, therefore the limit was hard-coded to 800 pixels.

The aberration correction masks (ACmasks), which have the same size as the input image, indicate, which mask has to be used to correct a particular position of the replay field. Each entry in ACmasks is a number between 0 and N.

zernCorr are the Zernike coefficients of every corrective phase mask. The second-order approximation is used, therefore values of $a_1 \dots a_{15}$ have to be recorded. An additional tilt correction is incorporated into the first two coefficients a_1 and a_2. For MatLAB, the recalculation of the corrective phase masks is a lengthy procedure, therefore the precalculated phase masks are also included in *phaseMasks* variable. Each of the N mask is the same size as the SLM.

The values *SLMx*, *SLMy*, *IMGx*, *IMGy* are currently hard-coded to $(1280, 1024)$ and $(1280, 640)$ respectively, but can straight-forwardly be adjusted whenever a new SLM is purchased.

5.7 Conclusions

This chapter summarizes the automated testbed for factory-assembled holographic projectors. All the methods, suited for a previously developed Piecewise-Corrected OSPR are presented.

This chapter serves as a link between Chapters 3 and 4. The aberration correction algorithm developed in Chapter 3 is used here as a tool to find an optimal corrective phase mask for a given spot. After such correction is performed in a number of replay field positions, the mechanism is used to find the optimal region boundaries.

The task is, for every RPF position to find a particular correction number that works best in that region. This is performed by displaying a grid of pixels with multiple corrections and algorithmically deciding for every spot, which correction minimizes the fitness function. For that reason, a mechanism, which accepts a picture of the replay field and separates it into single points is designed. Then, a spot from one correction is matched with the same spot from other corrections and they have their fitness compared. That procedure is performed for every RPF position.

This method proved to work reasonably well, but sometimes led to noise around region boundaries, when both corrections were comparable. This noise is then eliminated by the error correction mechanism, which, assuming continuity of region boundaries, decides whether the particular point fits to its neighbourhood. After the correction, the mask boundaries are greatly improved.

Once the boundaries are assigned, it often happened that there was still some misalignment between the masks leading to imperfect stitching. This is often caused by the implicit tilt in the phase masks. In order to eliminate this error, an additional tilt correction is performed after the region boundaries are assigned. As it can be seen in Chapter 4, sometimes the error still remains. However, given more time and effort, it should be exactly eliminated in the future.

Distortion measurement is characterized somewhat independently of other corrections. Several authors corrected distortion based on the rectangular grid. However, in this case the system is to a good extent rotationally-symmetric. Therefore, it proved sufficient to measure distortion based on the image of concentric circles. Starting from the centre of the field and sampling a single line outwards, a number of peaks in the image space is noted. With the knowledge that the circles are equally-spaced in the input image, and under the assumption that the distortion curve should be tangential to $y = x$ in the limit of the radius going to zero, the distortion curve can easily be calculated. Although simple, we found this method to be more than sufficient for rotationally-symmetrical projectors.

The experimental setup, forming the testbed, consists of two separate setups. One of them is used for aberration correction and looks at a smallest scale of a single point by projecting image directly onto a surface of a webcam. The other is used for distortion correction and aberration region assignment and consists of a screen overlooked by the dSLR. The two setups can be used interchangeably. The webcam, mounted on motorized rails can be pointed at any given position in the replay field. Whenever the second setup has to be used, the webcam can be moved out of the field of view and the image then gets projected on the screen. We found this mechanism especially useful and convenient. The full automation of the webcam's movement also made remote operation and monitoring possible.

Following all the characterizations, the final parameters are gathered into a single data structure. This structure then fully characterizes a given projector and can be passed to PC-OSPR algorithm, allowing it to generate holograms, leading to a high-quality image suited to the imperfections of the particular projector.

All of these methods are automated to a great extent. There are still certain manual tasks that need to be performed, but it is postulated that, once the project is carried on the commercial scale, full automation is easily achievable.

The nature of the correction methods, being generic enough can be ported to other types of holographic projectors. Chapter 6 proves that they can be applied to a Nematic Liquid Crystal devices and it is certain that they can also be applied to devices comprising DMD modulators.

A great advantage of all the presented methods is the fact that they characterize projectors only on the output projected image, and hence do not require any projector modifications.

Chapter 6

Holographic projector designed for photo printing and maskless lithography

The work described in this chapter is the result of a collaboration with Dr Phillip Hands from Edinburgh University and Trevor Elworthy from LumeJET photo printing company. The objective of the study was to determine whether holographic projection can be used in high-quality photo printing and maskless holographic lithography. The work described in this chapter is the direct application of the research described in the proceeding chapters.

6.1 Objective of the research and the overview of the project

The image projection system developed by LumeJET consisted of an array of Light Emitting Diodes (LEDs) [87]. The emitted light field was then concentrated using a specially-designed condenser, decreasing the pixel size. Such an image was scanned over the photosensitive paper to achieve resultant high-quality prints. As ingenious as the system was, it had a number of disadvantages:

- The number of LEDs was relatively small, and after the field has been shrunk, the area illuminated at once was of the order of few millimetres
- Printing process required precise alignment of the scanning mechanism and was very sensitive to lateral shift and vibrations
- A large LED linewidth caused a small, but noticeable crosstalk (overlap in-between adjacent inks)
- Due to a large number of very precise components, such printing systems were bulky and expensive

All of these problems can be dressed when the LED projection system is replaced with a holographic laser projector. Such devices are able to display as many pixels as the pixel count of the SLM, which is on order of millions. Lasers have much narrower bandwidths than LEDs, and hence can be tuned to precisely hit the absorption peak of particular inks, eliminating the crosstalk completely. Far field projection systems, such as all of the projectors described in this work are very insensitive to focal plane shifts. When the far field is reached, the image stays in focus, only slightly changing in size.

In Chapter 4, the objective was to construct a system that would project an image with acceptable quality in real time. In this chapter, on the contrary, the objective is not at all the speed, as the calculation of the holograms can happen offline before the actual photo printing. The research question can then be rephrased: "given practically infinite resources, what is the best image quality that can be achieved?".

After the initial tests and construction of Demonstrator 1, it occurred that the holographic projectors, while offering a huge advantage over a previous system, still cannot fully replace the LED system in the wide-angle printing configuration. The project then turned to the investigation of holographic projectors for use in holographic maskless photolithography.

Another, yet not fully exploited feature of holographic projectors is the ability to straightforwardly correct various optical errors, both, inherent in the optics and introduced in the manufacturing process. Therefore, the cost of such devices can be reduced dramatically.

6.2 Literature review of maskless holographic lithography

Multiple researchers studied maskless holographic photolithography. A large number of publications on the topic was reviewed, among which, three particular PhD dissertations were selected and studied in great detail as containing the most novel approaches to the subject. In this section, these approaches will be listed, compared, and will serve as a benchmark for the system developed here. In order to discriminate the pros and cons of the approaches, several factors will be taken into account:

- laser wavelength
- SLM resolution
- resolution and size of the replay field
- size of projected structures
- hologram generation algorithm
- digital correction included
- ability to construct 3-dimensional structures

6.2.1 Nathan J. Jenness, Duke University, 2009

Jenness used holographic lithography to perform patterning of photopolymers, proteins and for micro-manipulation of Janus nanoparticles. The laser wavelength was selected to be 532*nm* rather than in the UV region, because of live biological cell patterning.

Optical system

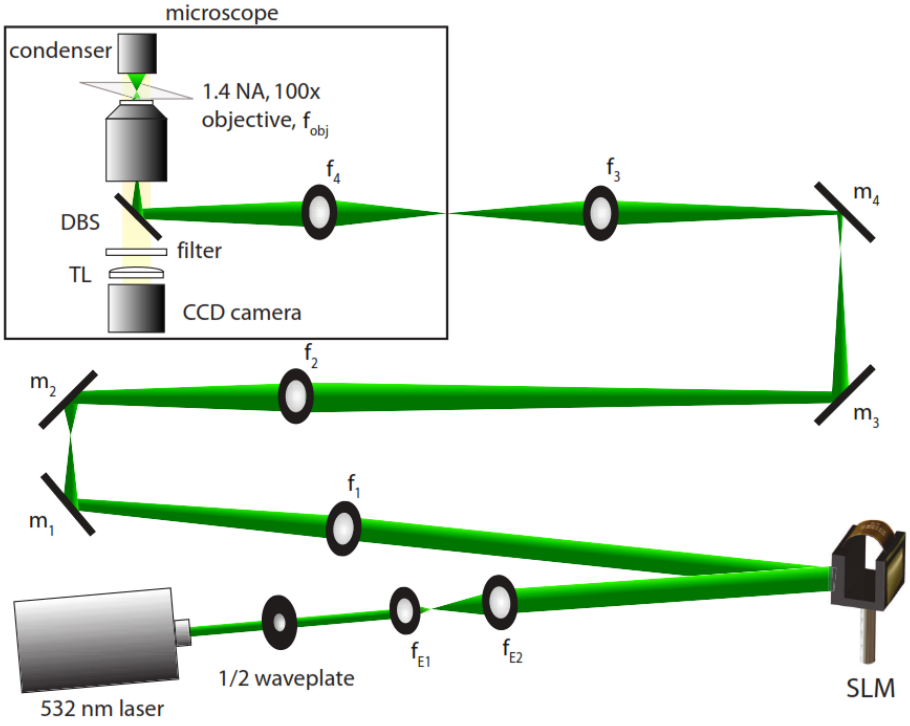


Fig. 6.1 Experimental setup used for maskless holographic lithography [38]

The optical system used by Jenness is shown in Fig. 6.1. The design is relatively complex, containing 7 different lenses and a microscope's objective. The usage of the microscope largely influenced the design, as it dictated the focal lengths of lenses f_3 and f_4 as well as the number of lenses. The system operation required a careful alignment procedure.

The SLM used in the experiment was a nematic Holoeye LC-R 2500, capable of performing an 8-bit phase modulation of $1024px \times 768$ pixels at a frequency of 72Hz [88].

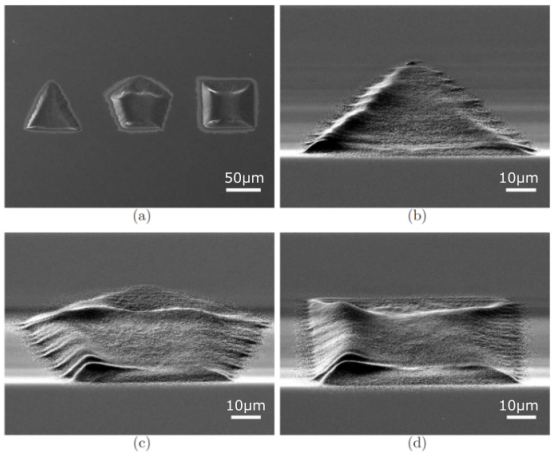
Replay field size

The patterning area is quite small, because of the microscope objective's limited field of view ($56\mu m \times 42\mu m$). The claimed resolution of the patterning is $330\,^{nm}/_{px}$ and the "usable" replay field size was $150px \times 175px$, again dictated by the aperture of the microscope.

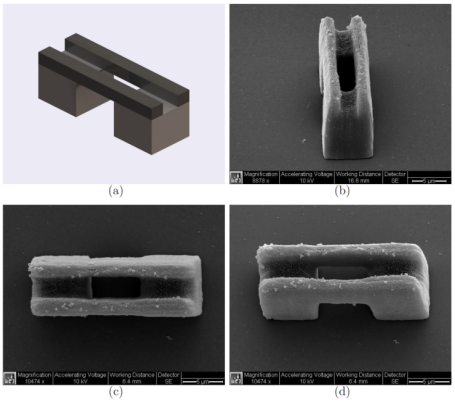
Hologram generation algorithm

The algorithm used for hologram generation was a traditional Gerchberg-Saxton algorithm. Although the SLM resolution was $1024px \times 768px$, this area was limited to $512px \times 512px$ to shorten the computational time and ensure a roughly uniform illumination profile.

Jenness also employed time-averaging of 10, 30, and 50 frames to improve the quality of the image (as seen in Fig. 6.2d).

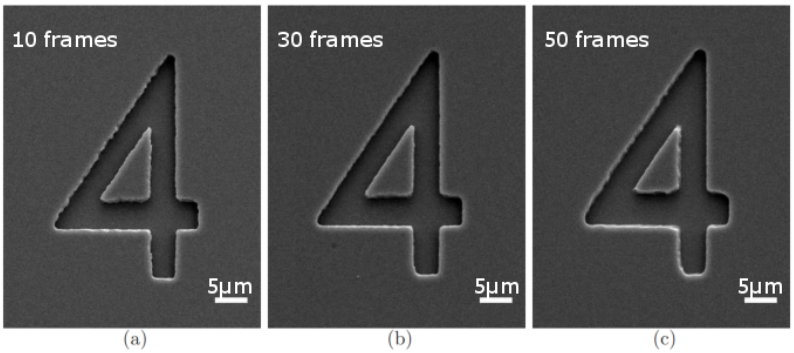


(a) Simple structures



(b) A complicated 3D structure

(c) An image



(d) Example of time averaging

Fig. 6.2 Jenness: examples of projected structures [38]

Digital correction

What Jenness refers to as aberration correction is just an adaptive intensity compensation applied to the target image before the hologram generation routine (similar to pixel shape correction presented in Chapter 4).

Projected structures

Jennes produced 2D as well as 3D structures, some examples can be seen in Fig. 6.2.

6.2.2 Christoph Bay, University of Cambridge, 2011

Optical system

The optical setup employed by Bay, presented in Fig. 6.3, was significantly simpler than the one used by Jenness. After the SLM, he only uses a single Fourier-transforming lens to project the hologram onto the sample plane. Additional modules were constructed to measure the intensity of light and observe the projected pattern on a camera. The laser wavelength was in the UV range ($402nm$).

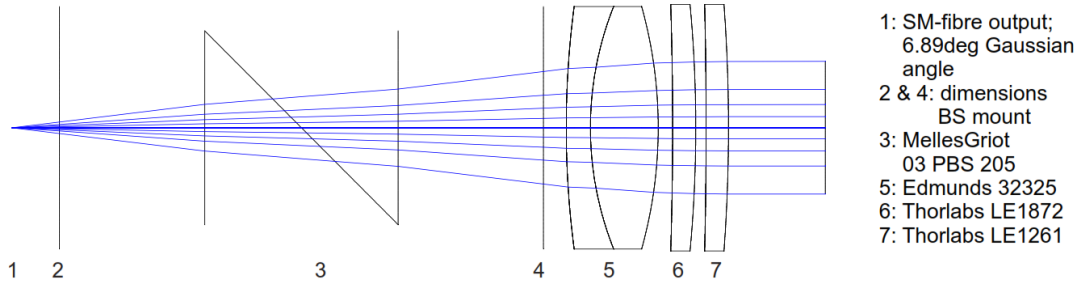
The SLM employed was the Holoeye HEO 1080P providing full-HD resolution of $1920px \times 1080px$ [88], but only the central area containing $1080px \times 1080px$ was used for hologram display.

Replay field

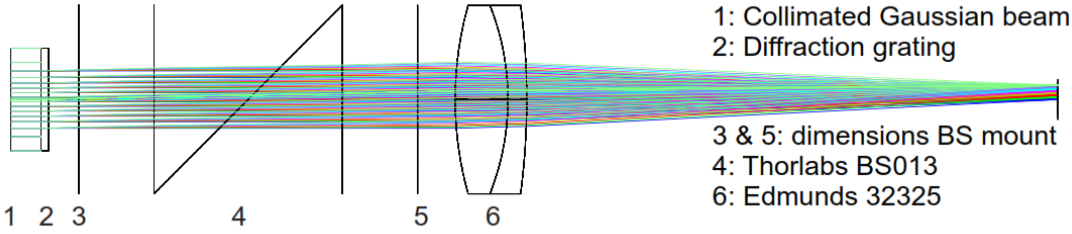
The airy disc radius in the system was $2.69\mu m$. The original replay field was $3.7mm \times 3.7mm$. However, because a significant zero order spot and various artefacts, it had to be constrained to $1.7mm \times 1.7mm$ with a resolvable pixel size of $3.7\mu m$. Bay did investigate the super-resolution algorithm using $3\times$ oversampling. However, he concluded that the improvement it brought was not satisfactory at a highly increased computation time.

Hologram generation algorithm

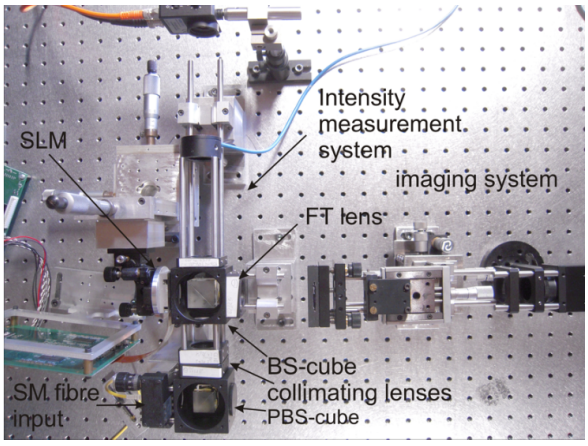
The algorithm used by Bay was so far the most advanced. He utilized a variant of Gerchberg-Saxton with Fienup feedback. On the top of that, he employed time-averaging with the OSPR algorithm with feedback of up to 100 frames.



(a) Zemax ray-trace of Bay's beam-expanding setup



(b) Zemax ray-trace of Bay's projection setup



(c) Bay's setup assembled

Fig. 6.3 Optical setup used by Bay [89]

Digital correction

The optical performance of the system was very good, due to a careful ZEMAX design (Figs. 6.3a-6.3b). The aberrations of the system across the field were not significantly bigger than the airy disc and did not require any further correction. To ensure an aberration-free image, Bay corrected the SLM flatness using interferrometric measurements [Fig. 6.4], in the same way as Freeman [56]. The effect of this procedure can be seen in Fig. 6.5.

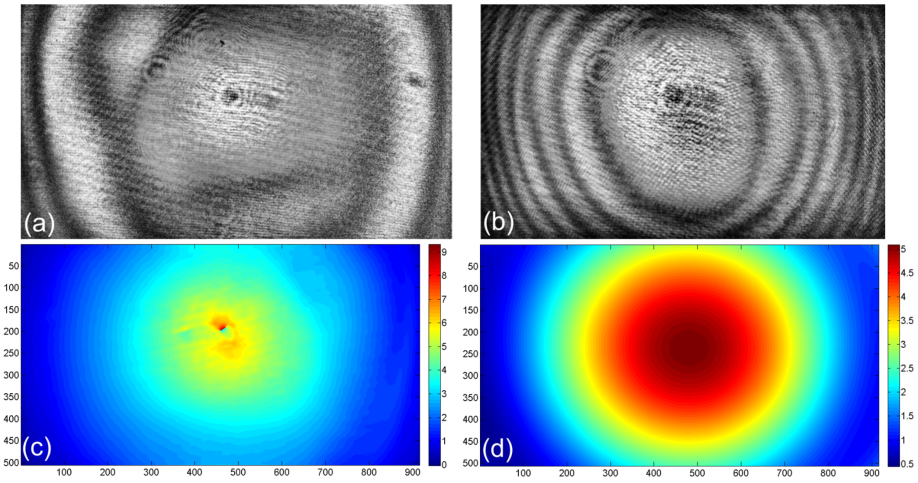


Fig. 6.4 (a) Flatness of the SLM measured with an interferrometric technique, (b) Pattern displayed to measure the curvature of the SLM, (c) An unwrapped phase mask, (d) The fit of the unwrapped mask to Zernike Polynomials $1-15$ [89]

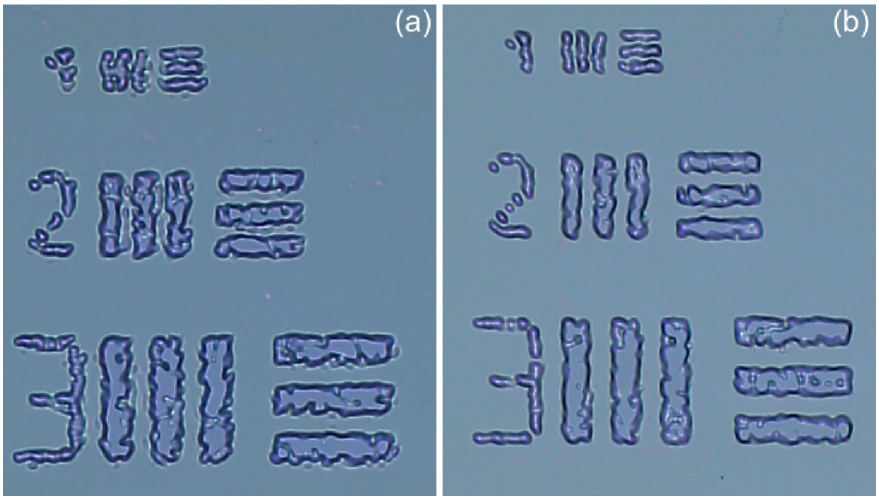


Fig. 6.5 The effect of flatness correction on the projected image: (a) before correction, and (b) after correction [89]

6.2.3 Daniel R. McAdams, University of Pittsburg, 2012

McAdams developed an optical system for two photon dynamic maskless holographic lithography.

Optical system

The optical system used by McAdams resembles that used by Jenness. It contains 3 lenses following the SLM, including the microscope objective . He inserted a shutter and a neutral density filter to precisely control the light intensity and the illumination time. The SLM is a nematic Holoeye LC2002 device.

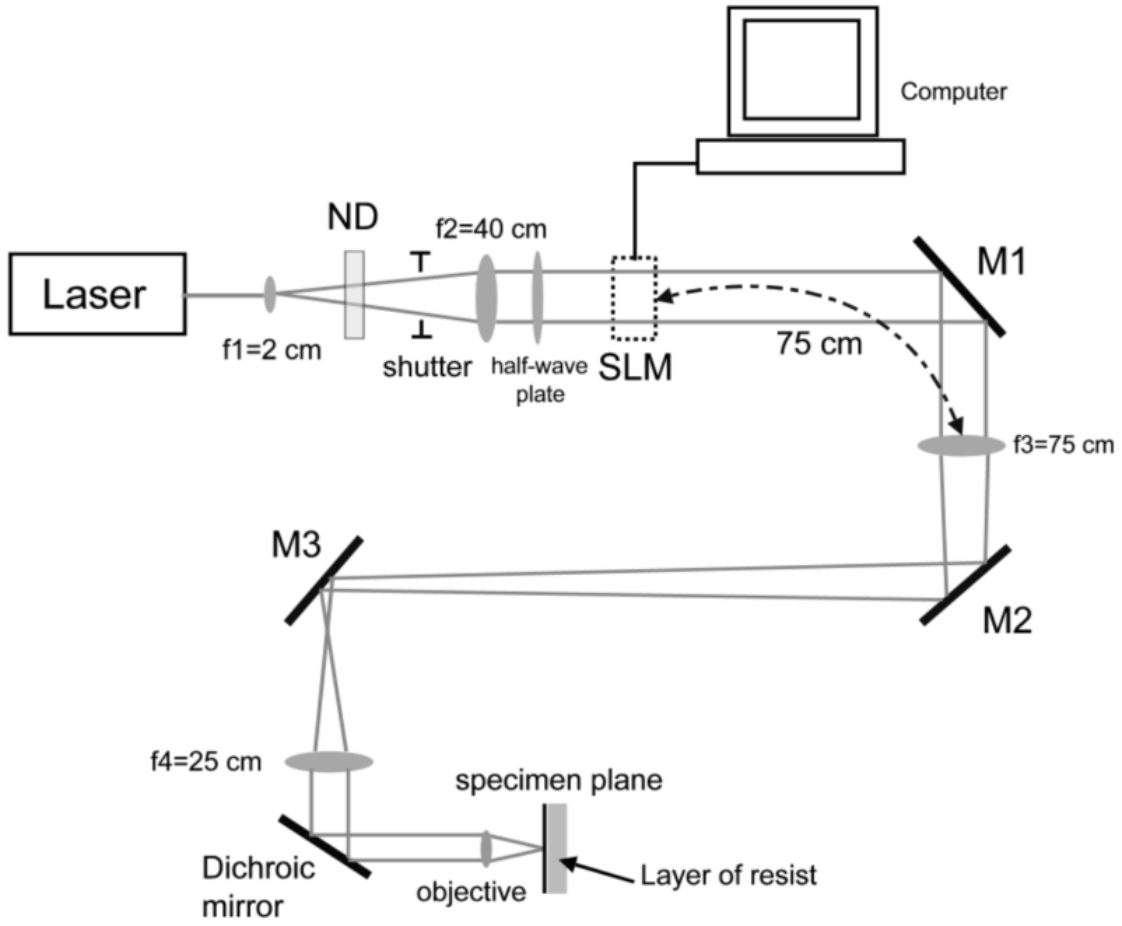


Fig. 6.6 McAdams: Experimental setup used for maskless holographic lithography [20]

Replay field

The theoretical calculations reveal that the maximum patterning area for this system can be 84x84 um. Given the 512x512 sampling of the Gerchberg-Saxton algorithm, the theoretical limit to the resolution is 160nm. Achieving this resolution is prevented by the resist, which has a smallest curable voxel size on the order of 300nm. Nonetheless, this is a rather impressive result.

Another limit to the patterning area is the fact that the camera used in the experiments is of the size 42x56 um. Because of the feedback-loop correction routines performed on the system, the aperture of the camera is the real limiting factor. The layout of the camera and the RPF can be seen in Fig. 6.7

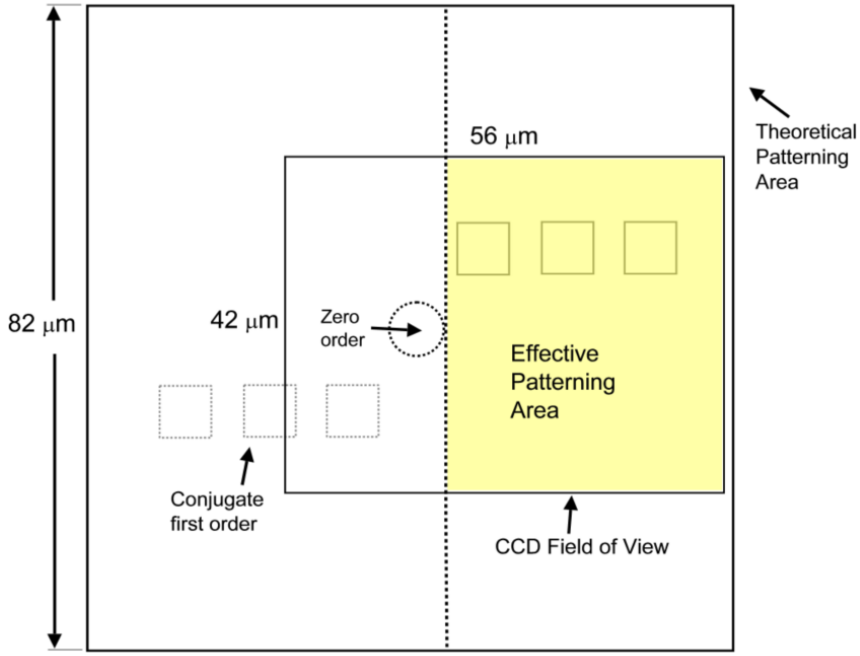


Fig. 6.7 Theoretical and the actual patterning area [20]

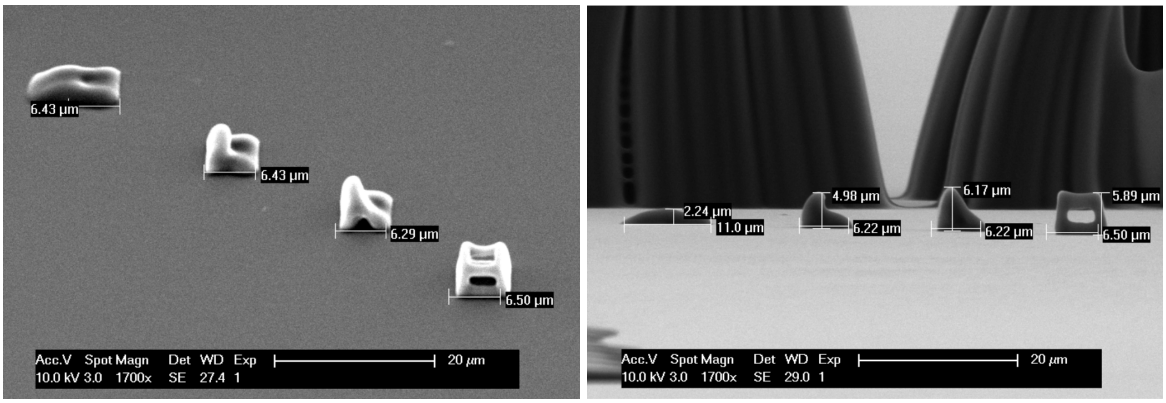
Hologram generation algorithm

McAdams again employed the Gerchberg-Saxton algorithm. He projected 2D as well as 3D structures. An interesting feature of the algorithm employed by him was the parallel 3D GS patterning. Instead of projecting separate layers of the structure, the algorithm employed by him [90] attempted to constrain the light field in 3-dimensions.

In a similar way to the other researchers, McAdams performed time-averaging of holograms using 5, 10, and 20 frames.

Digital correction

McAdams performed an adaptive aberration correction. Although significantly simpler to what has been presented here, it proved successful in correcting a relatively small wavefront errors that he dealt with. The effects of this procedure can be seen in Fig. 6.8.



(a) Perspective view (b) Side view

Fig. 6.8 McAdams: Adaptive Aberration correction - the amount of correction increases from left to right [20]

Projected structures

McAdams presented a number of high-quality 2D and 3D structures. A particularly interesting example is a free-standing three-legged structure seen in Fig. 6.9.

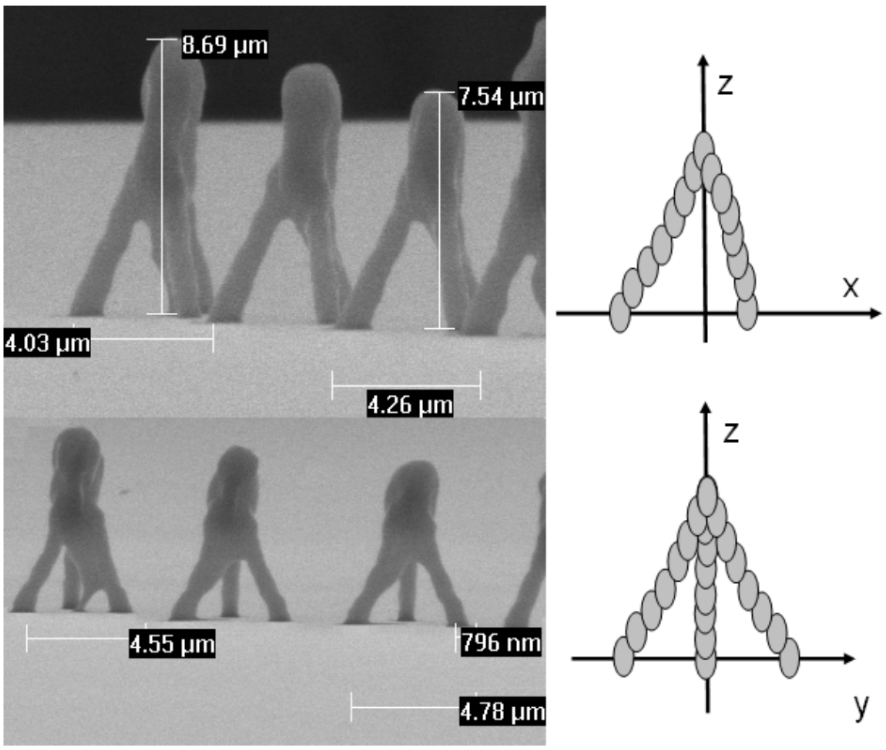


Fig. 6.9 McAdams: structures produced with 3D holographic lithography [20]

6.2.4 Comparison of the approaches

Quantity			Jenness	Bay	McAdams
Laser wavelength		(nm)	532	402	532
Diffraction limit		(nm)	N/A	2690	160
Resolution	SLM	$(px \times px)$	1024×768	1920×1080	800×600
	FT		512×512	1080×1080	512×512
RPF	size	$(um \times um)$	N/A	3700×3700	82×82
	resolution	$(px \times px)$	N/A	512×512	512×512
Image	size	$(um \times um)$	56×42	1700×1700	42×56
	resolution	$(px \times px)$	150×175	512×512	350×250
Hologram generation algorithm			Gerchberg-Saxton	Gerchberg-Saxton with Fienup Optimization	2D and 3D Gerchberg-Saxton
Frame averaging			up to 50 frames	up to 100 frames	up to 20 frames
Advanced correction			Adaptive Pixel Shape compensation		
Aberraction compensation			N/A	Advanced ZEMAX design	Adaptive aberration correction

6.3 Design considerations

All of the lithography systems presented previously were examples of reasonably good, optimized designs. The optical performance was preserved either by using specialized optical components (a microscope objective by Jennes and McAdams) or a careful ZEMAX design (Bay). This thesis takes a reverse approach: the design here is very simple and rudimentary. An advanced digital correction is later employed to prove that this imperfect system can be fully corrected and display a high-quality pattern.

The design procedure has been split into few stages. First, the properties of the system were calculated from basic theory using a spreadsheet. When the approximate focal lengths were calculated, ZEMAX ray-tracing software was employed to simulate an approximate diffraction limit of the system using off-the-shelf optical components. Unlike other ZEMAX designs, the target of the optimization was not to minimize the aberrations of the system, but to arrive at the smallest diffraction limit. That was achieved by incorporating a virtual aberration correction inside the ZEMAX model. Once assembled, the setup did show a

considerate amount of aberrations, but the advanced aberration-correction methodology developed through this thesis was used to fully correct the system.

6.3.1 Calculations

A simplistic model of a holographic projector was employed to first calculate the approximate requirements for the lenses. The approximate optical design is shown in Fig. 6.10.

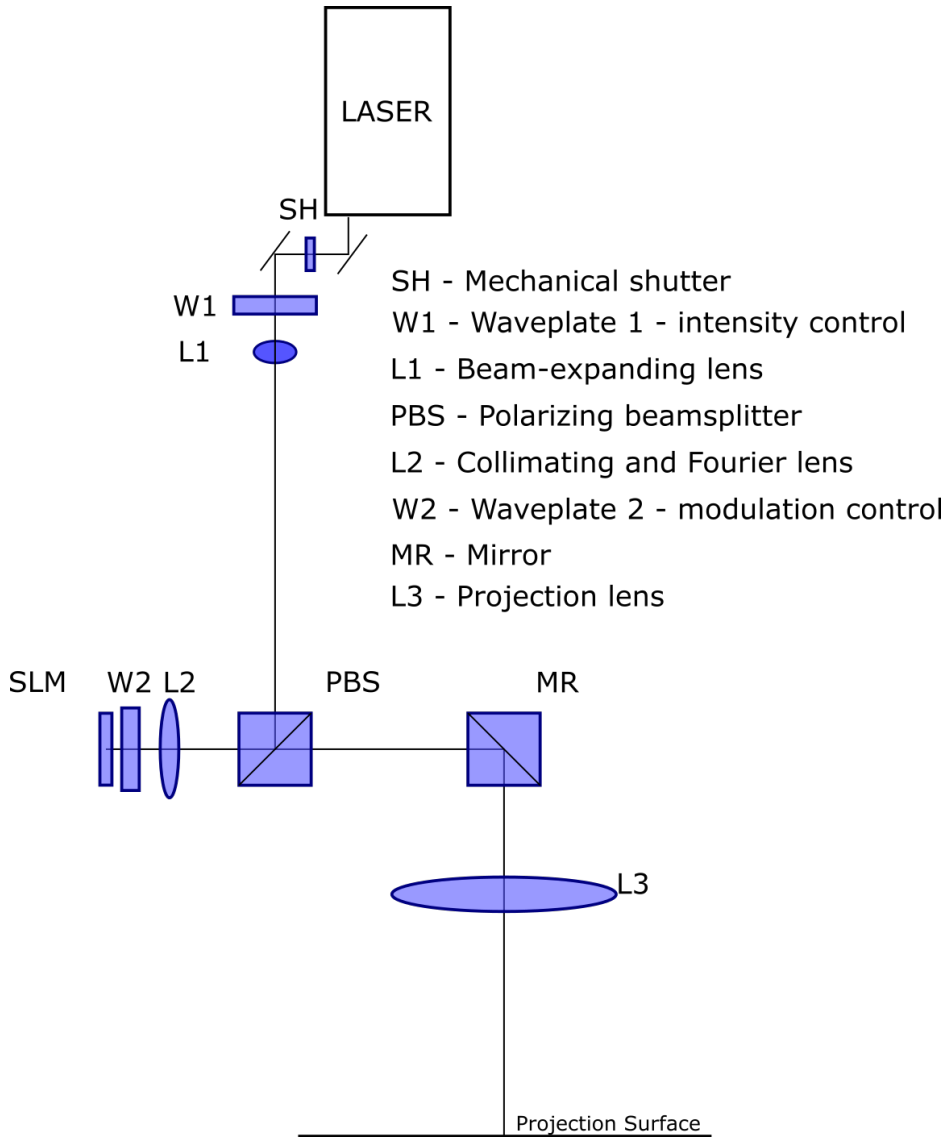


Fig. 6.10 A simplistic projector layout

SLM illumination

The laser provides a Gaussian beam, which needs to be expanded in order to fill the microdisplay. Due to the apodisation of the hologram, the structure of the pixel is defined by the width of the Gaussian at the SLM [19]. As Dr Hands once said, "throwing away photons is cheap", therefore we have decided to significantly expand the beam to achieve an almost flat profile at the microdisplay at a cost of losing a significant portion of light.

The beam gets expanded and collimated by lenses L1 and L2 in the Keplerian telescope configuration. The width of the beam will then be scaled by the factor of [91]:

$$w_{SLM} = w_{laser}\frac{f_2}{f_1}$$

where w_{SLM} is the beam width at the SLM plane, and w_{laser} is the width of the beam exiting the laser, which for the laser used here is equal to $0.9 \pm 0.09mm$ [92, 93]. The light distribution in the SLM plane will then be [91, 92]:

$$I(r) = \exp\Big[-2\frac{r^2}{{w_{SLM}}^2}\Big]$$

In order to calculate the difference between the centre of the SLM and the edge of the SLM, one substitute the half of the SLM's diagonal in the place of r. For the HOLOEYE PLUTO SLM used in this research, the exact value was $r_{max} = 8.8mm$. Putting all of the constants depending on the laser and the SLM, one arrives at the equation:

$$I_{edge} = I(r_{max}) = \exp\Big[-191.7\frac{f_1^2}{f_2^2}\Big]$$

This relationship has been plotted in Fig. 6.11. It can be seen that as the ratio of focal lengths $\frac{f_2}{f_1}$ increases, the intensity at the corner of the SLM rises. If one sets the edge illumination percentage to be, for instance 90%, the focal length ratio needs to be greater than 42.7.

System aperture after the SLM

The beam then illuminates the SLM, gets modulated in the reflection mode, and, after passing again throught the lens L2 forms a replay field at a focal distance. The size of the replay field at this plane is equal to [19]:

$$\alpha_0 = \frac{2f_2\lambda}{\Delta}$$

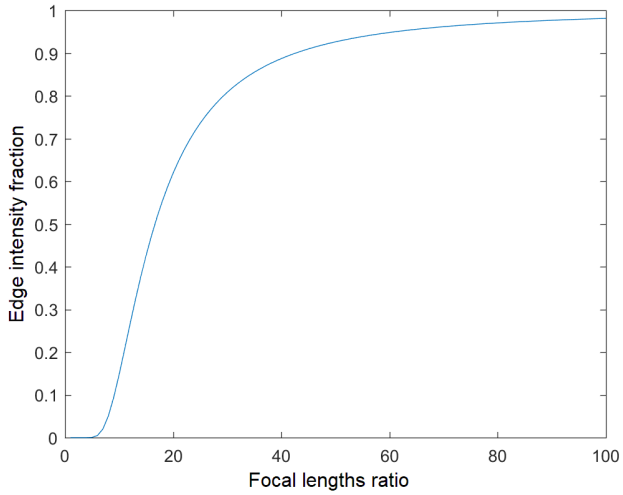


Fig. 6.11 Edge intensity depending on the focal lengths ratio

At this point it should be made sure that the apertures of the system are big enough not to clip any part of the replay field.

Image formation

Lenses L2 and L3 together form a telescope, which produces a demagnified image of the hologram in the focal plane of the lens L3.

The hologram then undergoes a rapid free-space diffraction and reaches the Fresnel field right before the image plane. This configuration was used in order to substantially increase the depth of focus. The holographically-projected images in the Fraunhofer region only change in size and not in structure. On the other hand, holograms that form in the focal plane of the lens very quickly go out of focus with increasing distance. Fresnel region is an intermediate case. Images still change in structure, but this change can be described precisely with a Fresnel Transform. In reality, it can be thought of as a Fourier Transform multiplied with a quadratic phase factor (third Zernike Polynomial).

The distance necessary for the fre-space propagation to go into the Fresnel region can be defined as [94]:

$$l_{fr} = 0.63\sqrt{\frac{D^3}{\lambda}}$$

where D is the maximum linear dimension of the diffracting structure (which in this case is half of the small hologram's diagonal).

The distance between the focal plane of the lens and the photosensitive paper (R) has to be greater than the Fresnel distance l_{fr}. Once this criterion is met, one can calculate the

overall size of the image. Given the size of the demagnified hologram and the propagation distance, the approximate size of the replay field can be calculated employing the formula:

$$\alpha_0 = \frac{2R\lambda}{\Delta_2}$$

where Δ_2 is the feature size of the demagnified hologram. Assuming that the RPF is quantized evenly in (M,N) points, one arrives at the pixel spacing in x- and y-dimensions:

$$x_{res} = \frac{\alpha_0}{M}$$

$$y_{res} = \frac{\alpha_0}{N}$$

6.3.2 Spreadsheet calculations

In order to find an approximate focal lenghts for the system, a spreadsheet was created. The parameters taken as an input are: $\lambda, \Delta, M, N, R, f_2, f_3$. From there, the spreadsheet calculates the Fresnel and Fourier Distances and the pixel spacings in the image plane. Various focal lengths (of lenses available in the lab) are then tested until the desired pixel size is achieved. The calculations are performed in MS Excell and the picture of the spreadsheet can be seen in Fig. 6.12.

It should be emphasized that these considerations are only approximate and the equations are solved with brute-force approach.

System params		
f2	250	mm
f3	50	mm
R	200	mm
Fresnel distance	63	mm
Fourier distance	11676	mm
Fresnel + lens	113	mm
FOV (input)	33	mm
FOV (output)	133	mm
Pix spacing X	69	um
PIX spacing Y	123	um

SLM/laser params					
Pix pitch	8	um	=	8E-06	m
M	1920	pix	=	0.0154	m
N	1080	pix	=	0.0086	m
D/2	1101	pix	=	0.0088	m
D/2 telescope	1.762	mm	=	0.0018	m
Wavelength	532	nm	=	5E-07	m
Pix pitch 2	1.6	um		2E-06	m

Fig. 6.12 Field of view and pixel spacing calculations

6.3.3 ZEMAX simulation

Once the focal lengths are settled for, the physical system, comprising real lenses is modelled. Lens models that can be straight-forwardly inserted into ZEMAX are freely available on the manufacturers' websites (Thorlabs [95] and Edmund Optics [96]). Once the model (an example shown in Fig. 6.13) is created, the constraints, imposed by the the projection chamber, are accounted for.

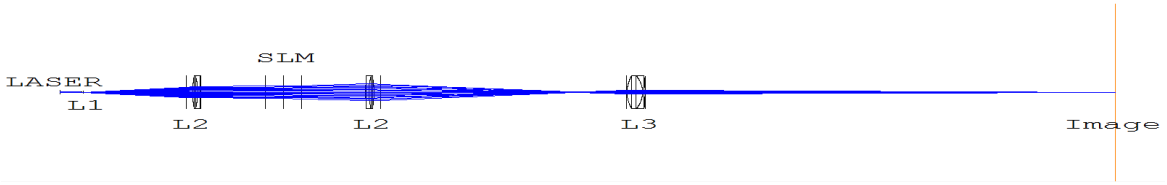


Fig. 6.13 ZEMAX model of a holographic projector

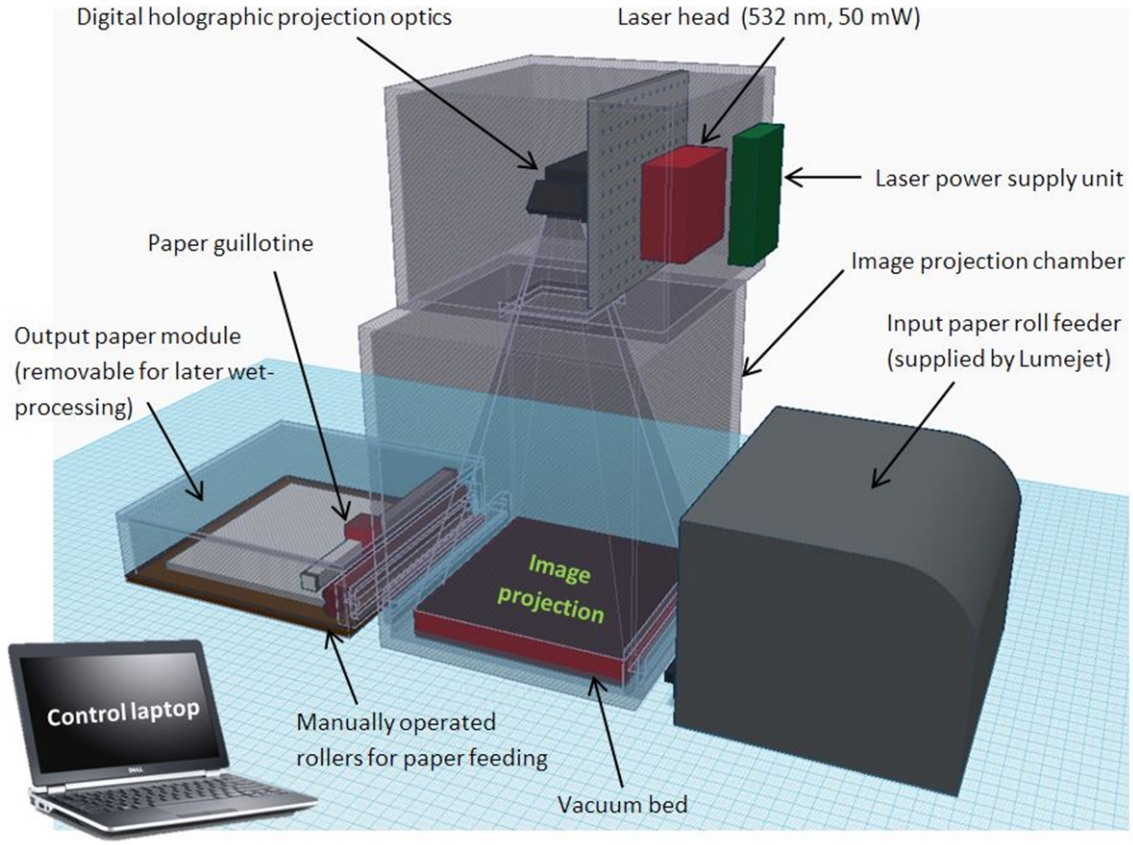
The optimization of this system then is split into two separate stages:

- **Focusing:** The optimization variables are set to be the lens positions, and the optimization target is the spot size. The global optimization tool of ZEMAX is used to set the lens positions for which the spot size is the smallest. That procedure is equivalent to focusing the front lens of the system.
- **Aberration correction simulation:** Once the system is in focus, it is necessary to check whether the diffraction limit matches the figure already calculated in the spreadsheet. For this purpose, the Zernike Fringe surface is inserted in the place of the SLM. The Zernike coefficients $a_3 \dots a_{15}$ are set as an optimization target and the global optimization tool is again used to find such combination of the Zernike coefficients that minimizes the spot size. Once the simulation converges or gives a reasonably small spot size, the diffraction limit is inspected and compared with the pixel spacing of the system.

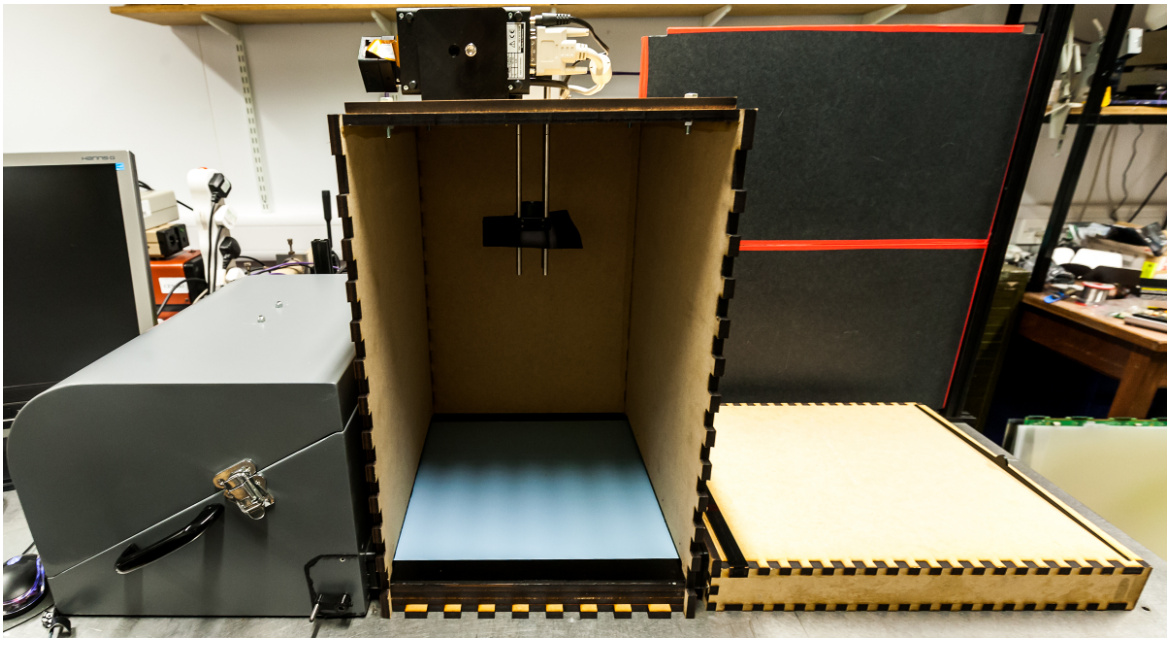
If the two figures match, the system is constructed using the given optical components.

6.4 Experimental setup

The setup has changed considerably through the project. However, its major elements, such as the projection chamber, number of lenses, a Spatial Light Modulator and the laser were the same throughout. The schematic view of the projector's major elements is shown in Fig. 6.14 and the detailed list of components used is presented in Table 6.1.



(a) Projector overview, outlining the elements [97]



(b) A picture of the projector

Fig. 6.14 Holographic projector designed for photo-printing and maskless lithography

Table 6.1 A list of components used in assembling the demonstrator projectors

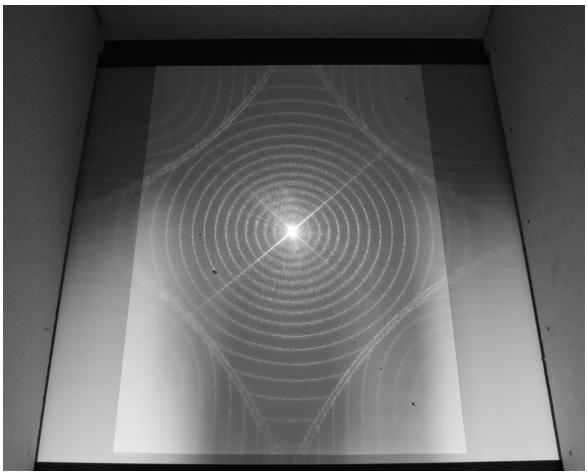
Component	Type	Specification	Part number	Used in projector
Laser	Quantum GEM	50mW CW laser		1, 2, 3
SLM	HOLOEYE PLUTO	Nematic, 1920x1080		1, 2, 3
L1	FibrePort Collimator	$f_1 = 15mm$		1
L1	Aspheric	$f_1 = 4mm$	A240TM-A	2, 3
L2	Edmund Optics Achromat	$f_2 = 225mm$	47-646	1
L2	Thorlabs Achromat	$f_2 = 250mm$	AC254-250-A	2
L2	Thorlabs Achromat	$f_2 = 300mm$	AC254-300-A	3
L3	Thorlabs Achromat	$f_3 = 25mm$	AC254-25	1
L3	Thorlabs Achromat	$f_3 = 50mm$	AC254-50-A	2
L3	Thorlabs Achromat	$f_3 = 100mm$	AC508-100-A	3
SH	Thorlabs Optical Beam Shutter		SH1	3

6.5 Adaptive-optical correction

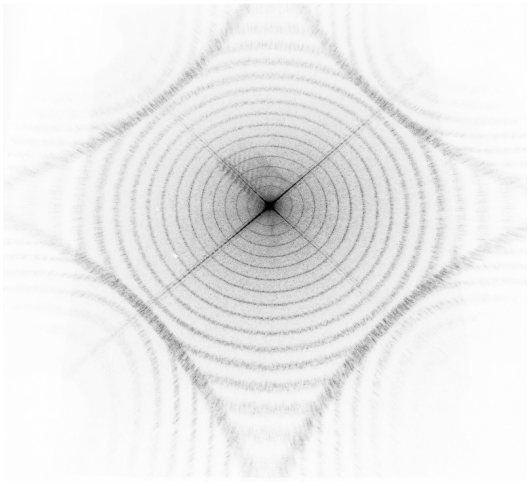
The correction procedure was carried out in the same way as described in Chapters 3-5. The corrections applied were: aberration correction, distortion correction and brightness equalization (sinc envelope correction). Because of the severe time constraint, the images taken are not well organized and some corrections are applied independently of others.

6.5.1 Distortion correction

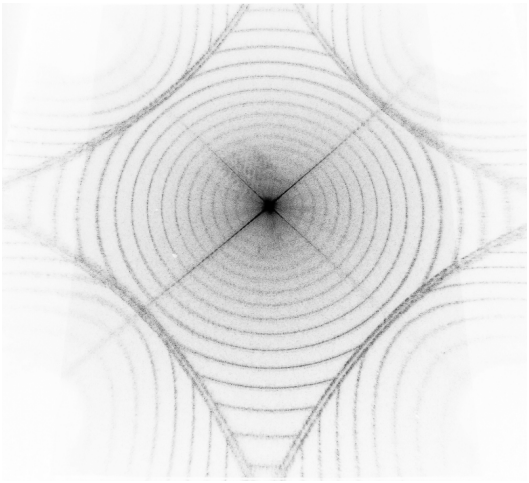
Demonstrator 1 in particular suffered a significant amount of distortion, because of a very wide field of view. Fig. 6.15a shows the inside of a projection chamber with a displayed image. The A4 page was put for scale as well as for image calibration. It can be seen that the distortion grows as the distance from the optical centre increases. This effect is visible especially at the borders of the image, where the square boundary lines resemble an almost parabolic shape. To further enhance the target image, two frames with different aberration correction were averaged: an uncorrected image, which is reasonably sharp closer to the optical centre [Fig.6.15b] and the image corrected at the edge of the replay field [Fig.6.15c]. To eliminate the error coming from the fact that the camera was pointing at an angle, an appropriate projective transformation was applied, and the zero order was subtracted to further enhance the quality of the target. The resultant distortion target can be seen in Fig. 6.15d and the retrieved distortion curve in Fig. 6.15e.



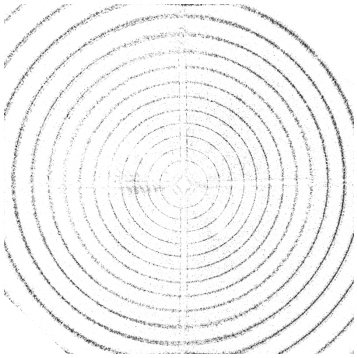
(a) Field of view of Demonstrator 1 with a distortion target displayed



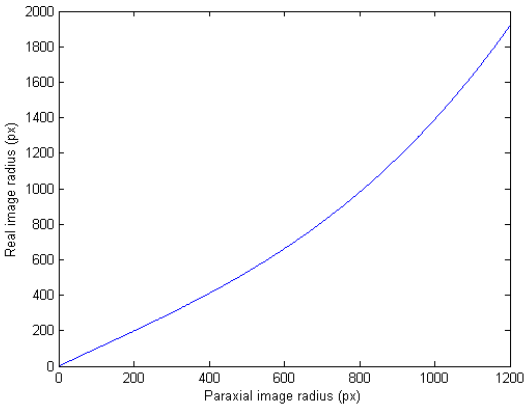
(b) Distortion target
no aberration correction



(c) Distortion target
edge aberration correction



(d) An average distortion target



(e) A retrieved distortion curve

Fig. 6.15
The process of distortion correction
Colours have been inverted in Figs. (b) - (d) for the sake of clarity

6.5.2 Aberration correction

Aberration correction was carried out in the same way as described in Chapter 3. The hologram generation methodology, however, had to incorporate the fact that the Spatial Light Modulator used continuous rather than binary phase modulation. Adapting the program to the new SLM device required a relatively easy modification of the quantization piece of the code. The program ran slightly faster, because only one full-phase hologram had to be calculated rather than 24 OSPR frames.

Initial experiments were carried out using a webcam as a feedback device. In later experiments, where a precise correction was required, the webcam was replaced with a Canon 400D dSLR in a LiveView mode laid face-up on the projection surface. In order to integrate the dSLR into the existing setup, a separate picture acquisition module was written. The new module initialized the dSLR through the Eos Digital SDK [73], set the zoom of the LiveView window to $10x$ and acquired the pictures one by one, exactly as the previous setup. Again, no screenshots of the module in operation remained. However, a computer running it can be seen in Fig. 6.16b and the entire setup can be seen in Fig. 6.16

6.6 Diffraction limit breaking

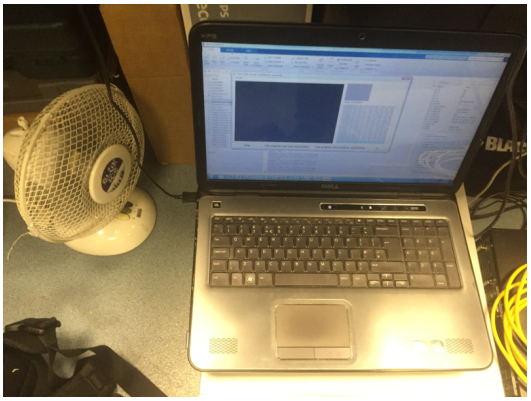
One of the crucial elements of the system that, to author's best knowledge, has not been thoroughly explored before, was breaking of the diffraction limit in holography. Cable describes this procedure in his thesis, mentioning two-fold increase in the resolution, but does not explicitly present breaking of the diffraction limit of the system. In this work, we have modelled the system to have a particular diffraction limit, which was then surpassed by the specially-designed hologram. It is achieved by employing a highly-supersampled version of the OSPR algorithm with Gerchberg-Saxton optimization.

6.6.1 Design of the algorithm

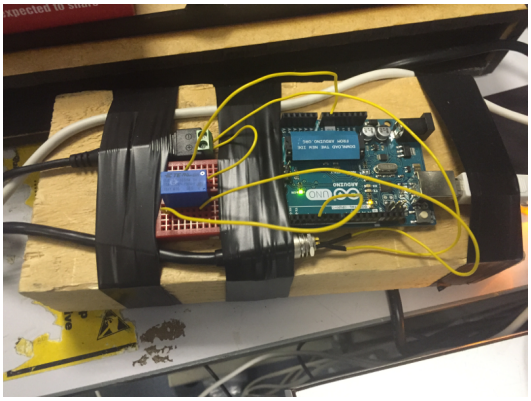
While constructing the hologram generation algorithm, the supersampling of the replay field was employed. This approach was inspired by Adaptive OSPR with Liu-Taghizadeh optimization described by Cable [40].

For the purpose of supersampling, a set of constraints of the IFTA need to change. The algorithm allowing for a generalized constraints is presented in Algorithm 7. It can be seen that it bounces back and forth between the image and hologram planes, imposing generalized constraints in both: the image and hologram planes. The particular constraints suited for this problem are going to be discussed consequently.

(a) Experimental setup at the time of Demonstrator 3



(b) Computer running the feedback loop optimization



(c) Arduino microcontroller, controlling the shutter

Fig. 6.16 Feedback loop

Algorithm 7: Generalized Iterative Fourier Transform Algorithm

H : Hologram to be generated
I : Input target image
N : Number of iterations

1 Add uniformly distributed random phase $\vartheta_{rnd}(u,v)$ to the image:

$$\mathsf{T}(u,v) = \sqrt{I(u,v)}\, e^{i2\pi\, \vartheta_{rnd}(u,v)}$$

for $\mathsf{q} \leftarrow 1$ **to** N **do**

2 Perform an inverse Fourier Transform:

$$H(x,y) = \mathscr{F}^{-1}\left\{T(u,v)\right\}$$

3 Apply the constraint to the hologram:

$$H_{constrained}(x,y) = \texttt{ConstrainHologram(}H(x,y)\texttt{, }x\texttt{, }y\texttt{)}$$

4 Perform a forward Fourier Transform:

$$T_{rec}(u,v) = \mathscr{F}\left\{H_{quantized}(x,y)\right\}$$

5 Apply the constraint to the image:

$$T(u,v) = \texttt{ConstrainImage(}T_{rec}(u,v), I(u,v), u, v\texttt{)}$$

6 **end**

The revised set of constraints (as outlined in Chapter 2) is:

- **Hologram constraint**
  The SLM can only display a fixed number of hologram pixels. Therefore, all the pixels that laying "outside" of the SLM will have the amplitude forced to zero, while the pixels "inside" of the SLM will have the amplitude quantized to the profile defined by the laser illumination and phase quantized to 256 phase levels that can be represented by the SLM:

$$ConstrainHologram(H(x,y),x,y) = \begin{cases} B(x,y)\frac{H(x,y)}{|H(x,y)|} & \text{if } \begin{cases} \frac{M_{FFT}-M}{2} \le x < \frac{M_{FFT}+M}{2} \\ \frac{N_{FFT}-N}{2} \le y < \frac{N_{FFT}+N}{2} \end{cases} \\ 0 & \text{otherwise} \end{cases}$$

Where (M,N) is the resolution of the SLM, (M_{FFT},N_{FFT}) is the resolution of the Fourier Transform and $B(x,y)$ is the illumination profile of the SLM.

- **Image constraint**
  In the similar way to AdOSPR-LT, the "don't care" regions of the image are assigned. Since the sampling in the image plane is very dense, the noise regions ensure that there are enough degrees of freedom to constrain the signal region in a sufficient number of points. The revised constraint in the image plane is then: the intensity in the signal region is assigned to be either identical to the target or a feedback term calculated from the LT algorithm, while the phase in the signal region is identical to the reconstruction. Both: intensity and phase of the noise field remain unchanged:

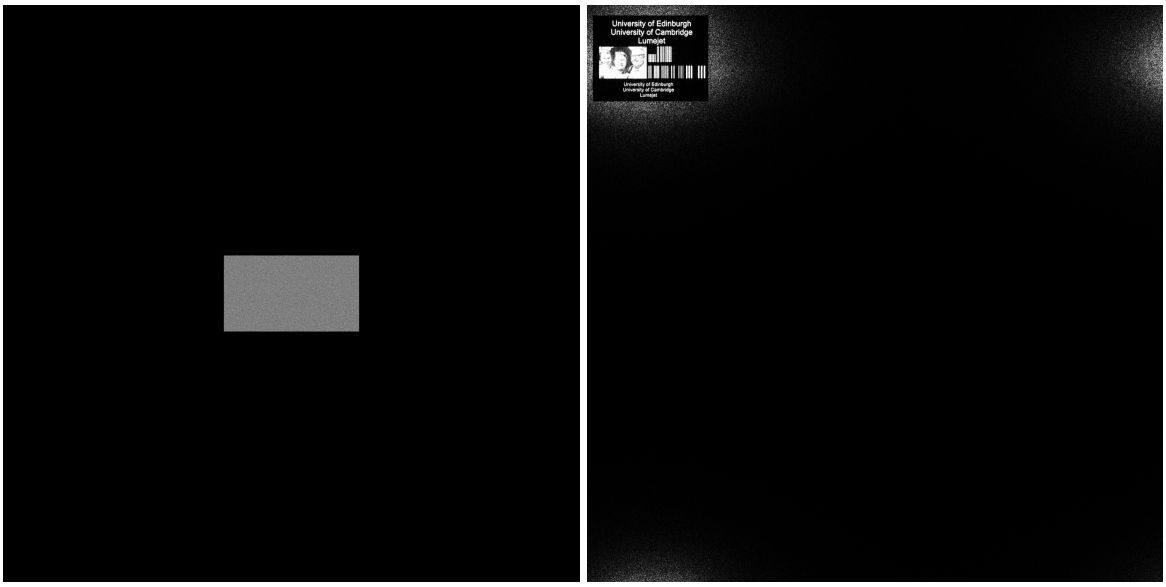
$$ConstrainImage(T_{rec}(u,v),I(u,v),u,v) = \begin{cases} \sqrt{I(u,v)}\frac{T_{rec}(u,v)}{|T_{rec}(u,v)|} & \text{if } \begin{cases} 0 \le x < M_{IMG} \\ 0 \le y < N_{IMG} \end{cases} \\ T_{rec}(u,v) & \text{otherwise} \end{cases}$$

An important decision to make is the resolution of the Fourier Transform operation. On one hand, a very high resolution will lead to big oversampling, and hence, in principle, higher quality image reconstruction. On the other hand, it seriously increases the memory as well as computational requirements of the generation algorithms. A few resolution variations were tested: 4096×4096px ($4K4K$), 8192×8192px ($8K8K$), 16384×16384px ($16K16K$), and 9600×5400px ($\sim 9K5K$). In general, it was found that $8K8K$ works significantly better than $16K16K$ while requiring much less computation. The problem found with $8K8K$ resolution was that the diffraction limit was different in two directions, caused by the non-square aperture of the SLM ($1920x1080$px). Therefore, the resolution, which was exactly 5 times greater than that of the SLM in both directions was tested ($9K5K$).

The result of the Super-resolution IFTA can be seen in Fig. 6.17. The hologram produced using the above algorithm, embedded into the $8K8K$ frame can be seen in Fig. 6.17a. The respective replay field, which, in practice is a Fourier Transform of the presented oversampled hologram can be seen in Fig. 6.17b. It can be seen that it consists of a small signal window (seen in the left-hand corner) surrounded by the noise window.

6.7 Image tiling

Once it is possible to achieve very high resolution inside a small region of the image, one can think of combining multiple tiles of this type and stitching them together in order to cover a bigger area. An advantage of this method is the fact that the holographic mechanism, unlike



(a) A hologram embedded into $8K8K$ frame (b) Respective RPF of the presented hologram

Fig. 6.17 A result of a Super-resolution algorithm

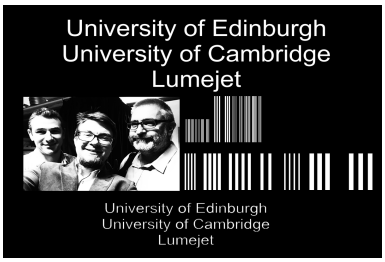
a mechanical one, does not require any calibration. Since no movement of the components is needed to perform it, the holographic stitching is always precise.

This idea is again, inspired by the work of Adrian Cable [40], who suggested a similar technique as a way to improve the image quality. Because the OSPR-LT technique introduces "noise regions" having an uncontrollable intensity, a shutter has to be used in order to selectively expose different tiles.

6.8 Results

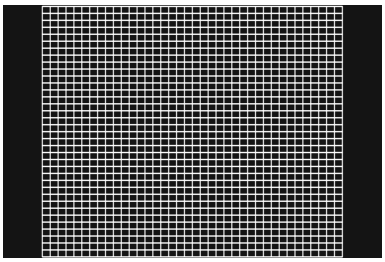
6.8.1 Target test images

Two groups of target images are tested. Binary targets (seen in Fig. 6.18), designed specifically to test the vertical and horizontal resolution of the image and the ability to eliminate noise in the signal region. The images from the second group are natural images from the author's personal collection (seen in Fig. 6.19). All the targets are of the size $1920px \times 1280px$ with a 10-pixel black border around them, except for Fig. 6.19e, which is used to test the image tiling and hence its size is 1920×1920.



(a) Resolution target containing different sets of parallel lines

(b) University crests



(c) An uniform grid

Fig. 6.18 Binary test targets

(a) Senate House, Cambridge

(b) Tower Bridge, London

(c) Butterfly and a bumble bee

(d) Selfie after a LumeJET on-site visit[1]

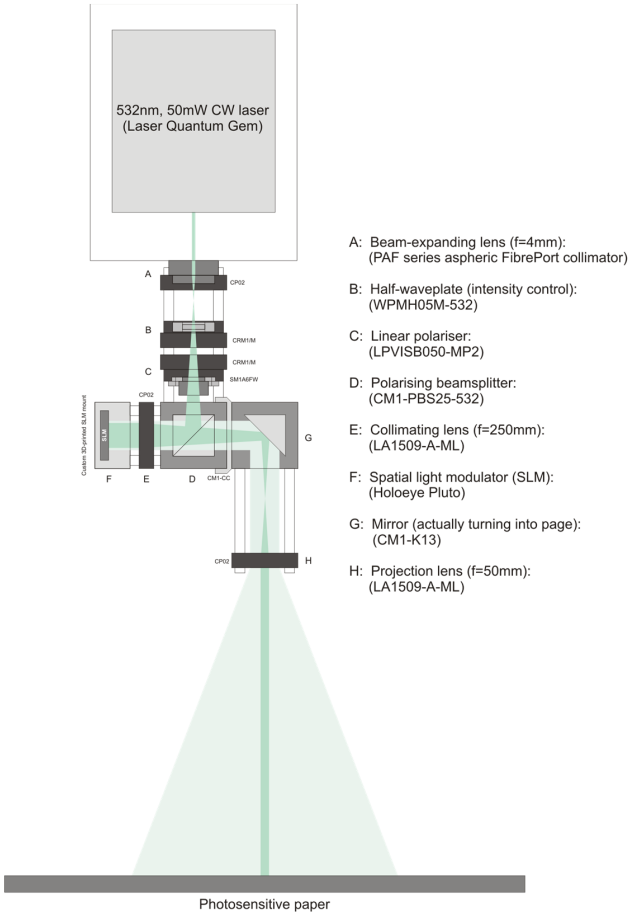
(e) Ice crystals shaped by the wind, used to test image tiling

Fig. 6.19 Set of natural images for testing colour reproduction

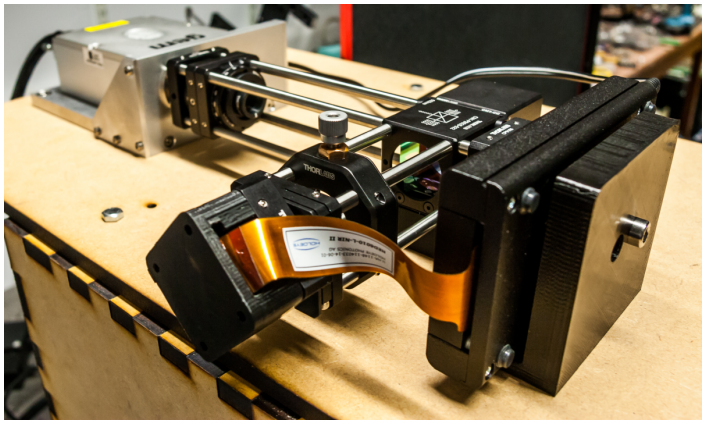
[1] from left to right: (ridiculously photogenic) Dr Phillip Hands, (hopefully soon Dr) Andrzej Kaczorowski, (tired and annoyed) Professor Timothy Wilkinson

6.8.2 Demonstrator 1 - wide-angle holographic printing

Optical design



(a) Optical design of Demonstrator 1



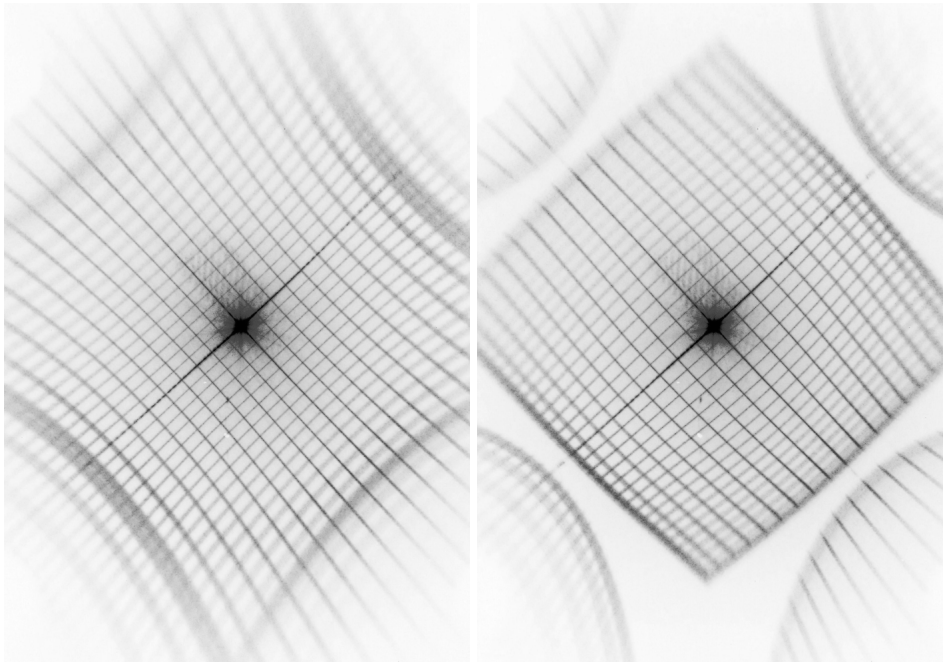
(b) Assembled Demonstrator 1, top of the projection chamber

Fig. 6.20 Demonstrator 1 projector

The optical design of the projector can be seen in Fig. 6.20a and the picture of the assembled setup in Fig. 6.20b. The projector was designed to have a wide field of view of 20cm x 20cm. The exact components of the system are: L1 - fibre port collimator, $f_1 = 4mm$, L2 = $f_2 = 225mm$, $f_3 = 25mm$

Distortion correction

The final correction can be seen in Fig. 6.21. Fig. 6.21a shows the original grid without the correction applied, while Fig. 6.21b shows the first distortion correction attempt. It can be seen that the distortion is not fully eliminated. The reason for this was a simple lack of time to perform a better correction. However, this being only the proof of principle, a precise distortion correction can easily be achieved in the future.

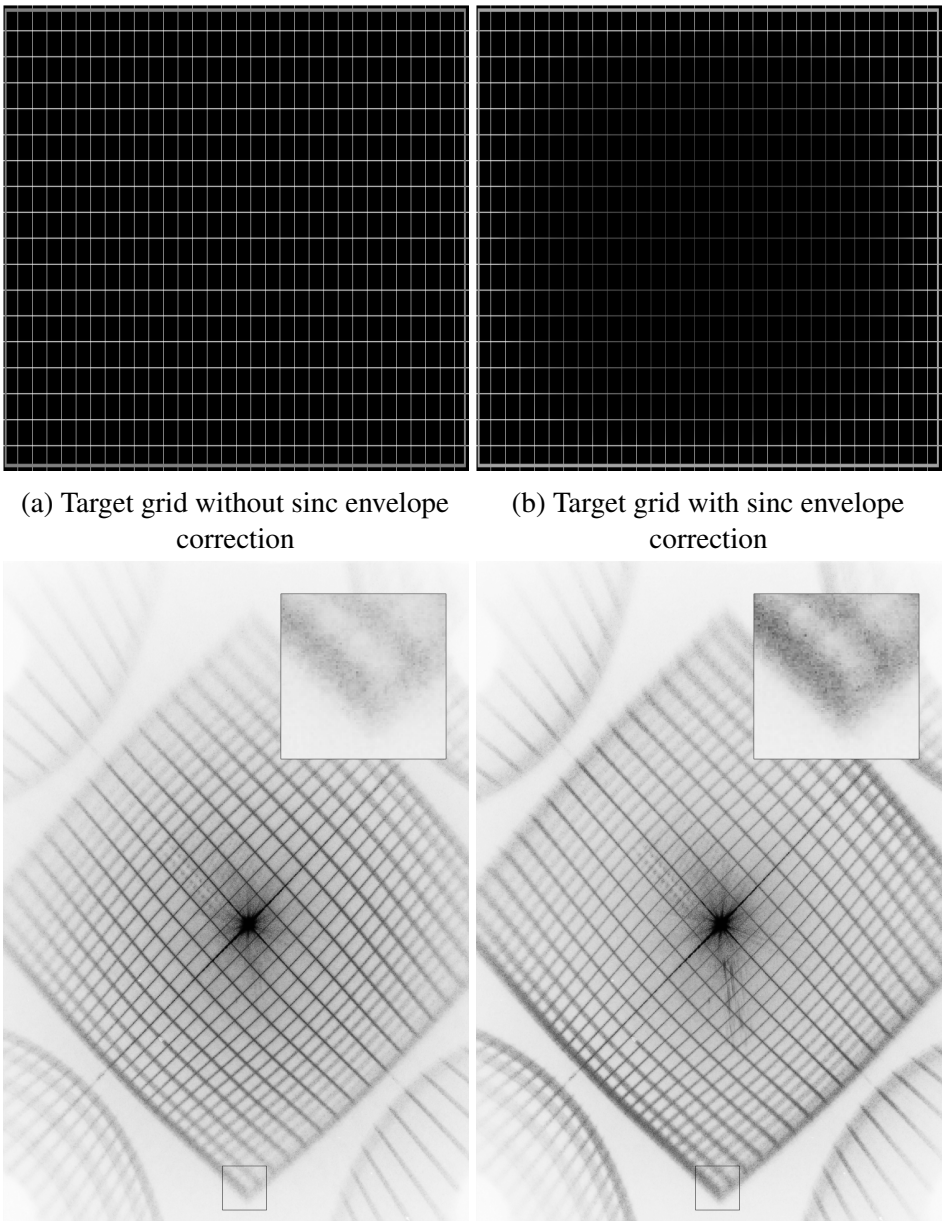


(a) No correction (b) Distortion correction

Fig. 6.21 Demonstration of distortion correction

Pixel shape correction

The SLM pixels are square-shaped, hence the replay field is modulated by the two-dimensional sinc function. This results in decreasing intensity towards the edges of the field (as previously discussed in Chapter 4). This error is accounted for by pre-computing the sinc envelope. The outer parts have their central intensity decreased to account for this error (as illustrated in Fig. 6.22).



(a) Target grid without sinc envelope correction

(b) Target grid with sinc envelope correction

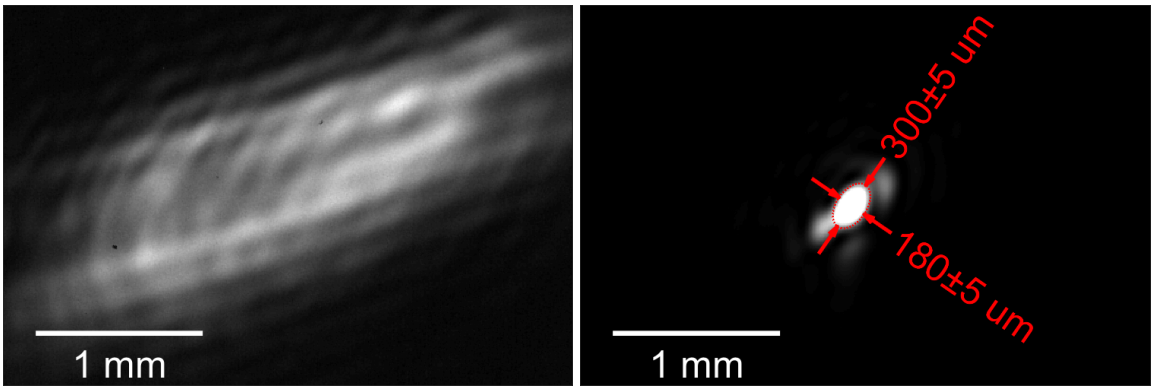
(c) RPF of the uncorrected target

(d) RPF of the corrected target

Fig. 6.22 Image intensity correction

Aberration correction

The aberration correction of Demonstrator 1 was performed employing webcam as a feedback device and the Hybrid Genetic-Heuristic Descent algorithm. An entire procedure took 5 hours and 11 minutes (58 iterations of the algorithm). The uncorrected and corrected spot sizes can be seen in Figs. 6.23a and 6.23b respectively.

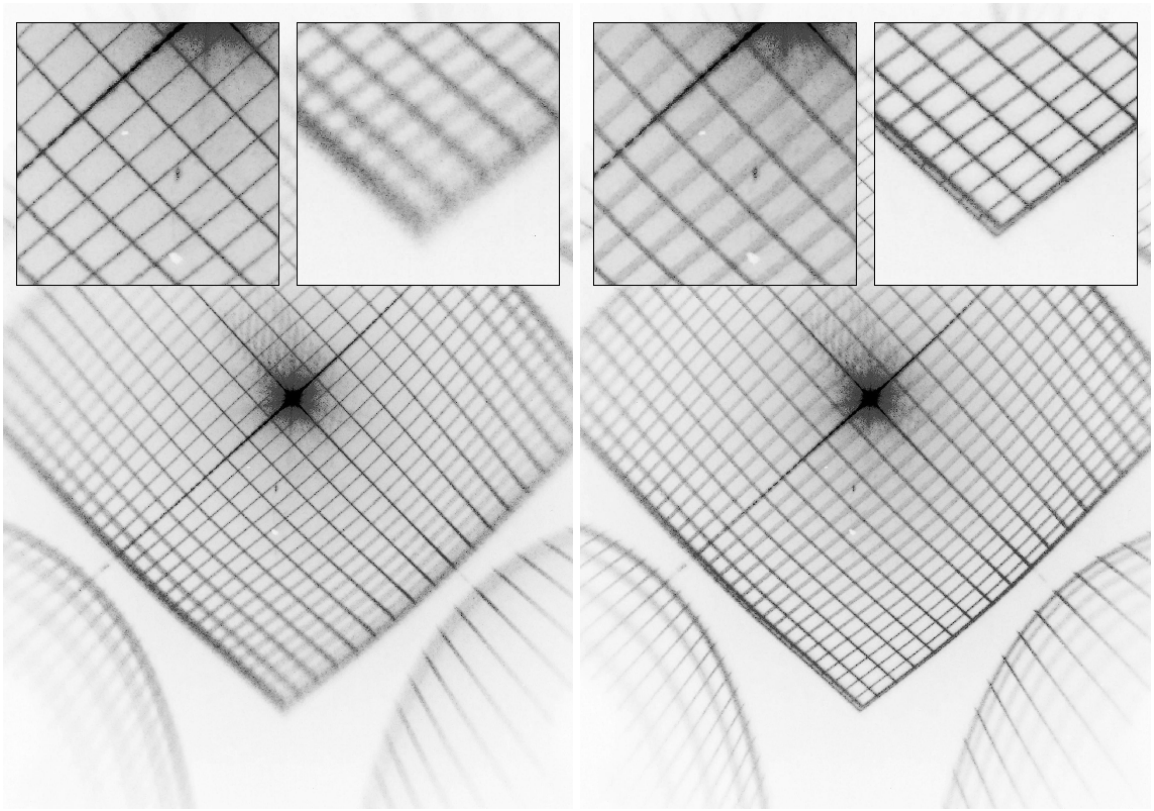


(a) An uncorrected spot of Demonstrator 1 (b) A corrected spot of Demonstrator 1

Fig. 6.23 An example of aberration correction in Demonstrator 1

Distortion with aberration correction

Distortion, aberration, and pixel shape corrections can easily be combined. Such an example is seen in Fig. 6.24.

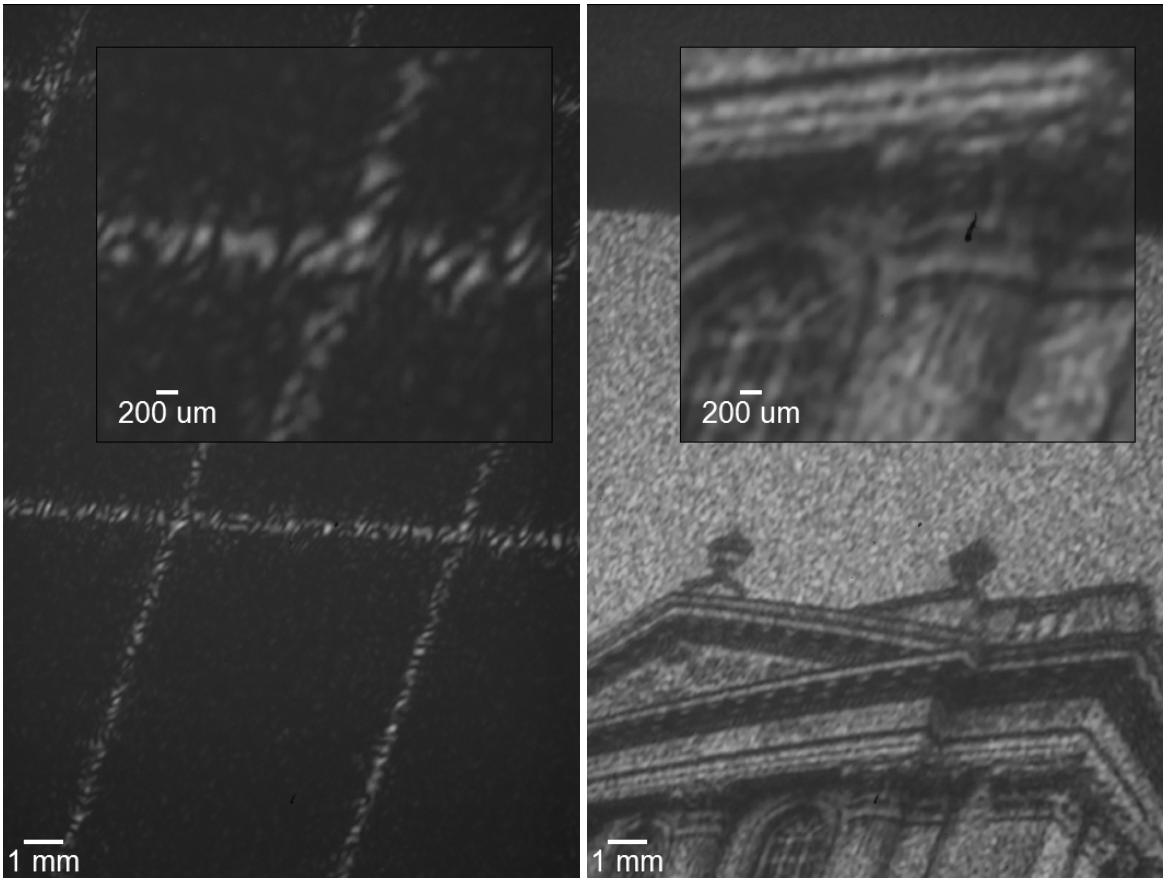


(a) No aberration correction (b) Edge of the field aberration correction

Fig. 6.24 Distortion and aberration corrections combined, Demonstrator 1

Diffraction limit breaking

The diffraction limit of this system was on the order of $250um$. That did not yet match the requirements set out by LumeJET. Therefore, additional techniques to circumvent this issue were investigated. In particular the super-resolution algorithm mentioned previously was employed. The result of its working can be seen in Fig. 6.25. Fig. 6.25a shows a diffraction-limited image of a grid of pixels, while Fig. 6.25b shows the replay field of a super-resolution hologram. Both images were taken in exactly the same conditions and have identical scales. It can be seen that while feature size of the grid is very similar to a previously-measured diffraction limit ($180 \times 300 \mu m$), the feature size of the super-resolution hologram is visually much smaller.



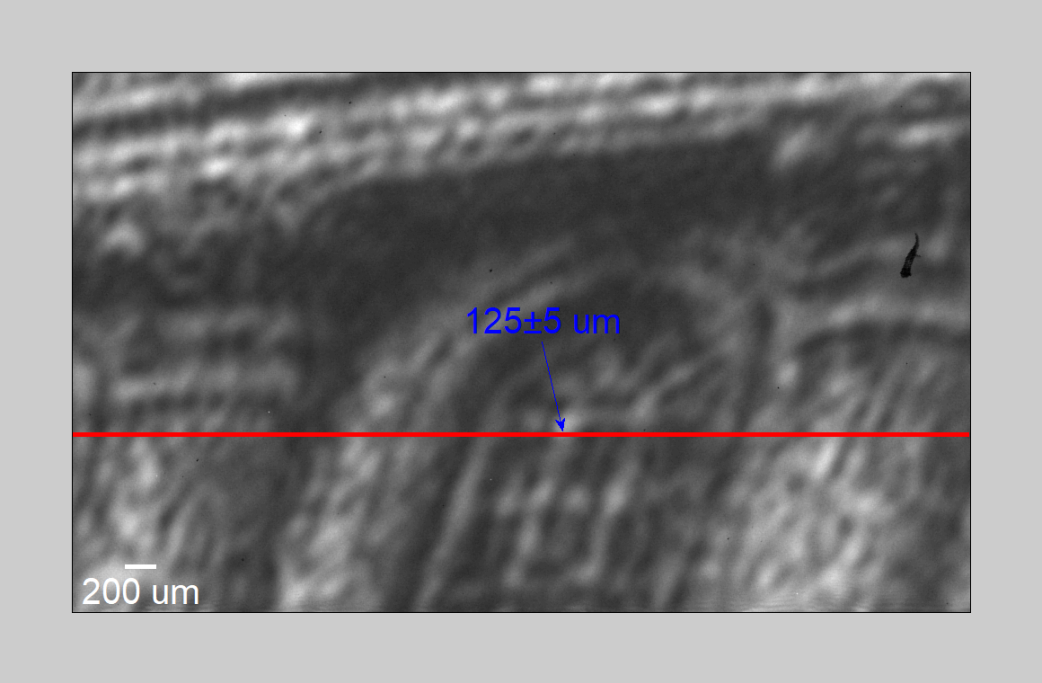
(a) Native HD resolution, Diffraction-limited grid

(b) $8K8K$ super-resolution, Target 1: Senate House, Cambridge

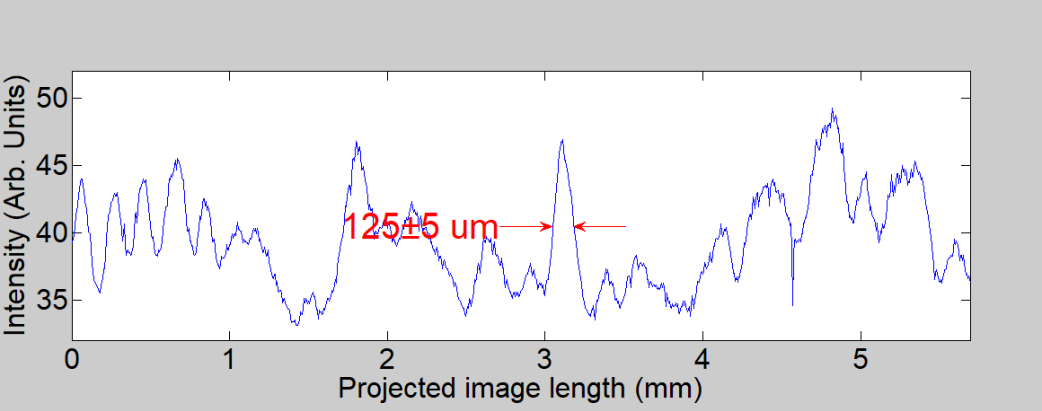
Fig. 6.25 Replay field of a Super-resolution algorithm

In order to quantify this, the cross section of the image seen in Fig. 6.26a across the red line has been taken and can be seen in Fig. 6.26b. After the inspection, one of the peaks in

the image, corresponding to a vertical window line has a measured feature size of $125 \pm 5 \mu m$. This is the first indication that the diffraction limit can be broken in holographic projection.



(a) Cropped part of the replay field



(b) Section through the replay field

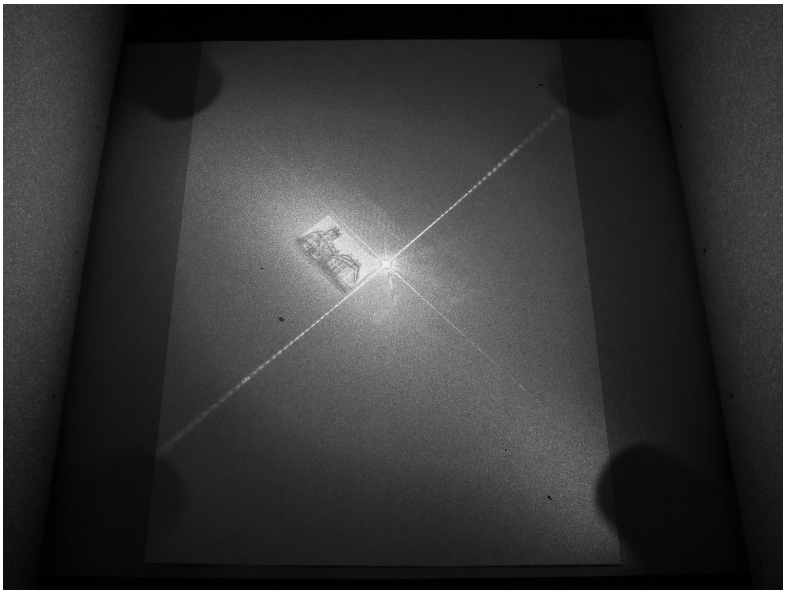
Fig. 6.26 Demonstrator 1: Resolution analysis

Image tiling

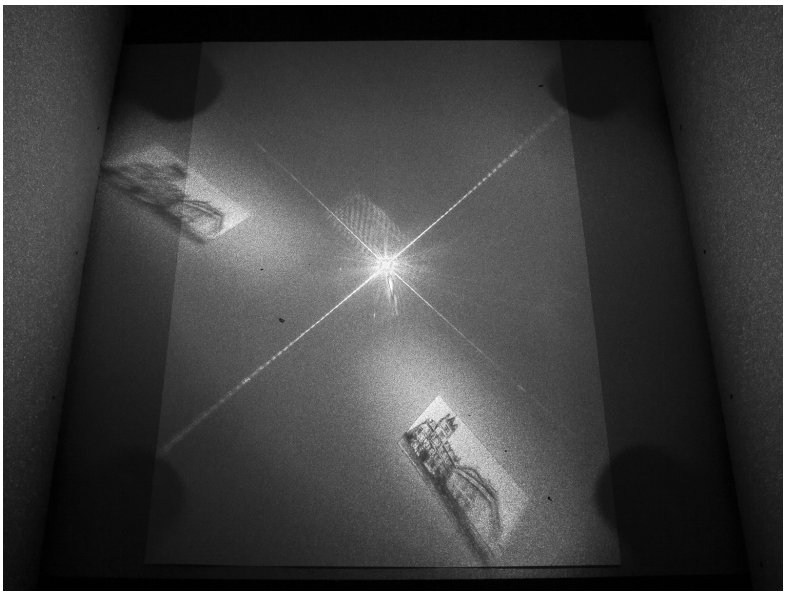
Once the holograms are computed using the super-resolution algorithm, one ends up with a relatively small and high-resolution signal region and a noise field outside. The position of the signal window can be arbitrarily changed by adding Tip and Tilt Zernike polynomials

into the wavefront. The result of this procedure can be seen in Fig. 6.27. It can be seen that a significant amount of distortion is visible, which has not been, at this stage, corrected.

One can imagine a situation where the noise regions are blocked by a selective replay field shutter mechanism and a number of such high-resolution tiles are exposed one next to the other. This mechanism would allow the drastic increase in the number of effective pixels in the replay field.



(a) An original RPF displayed in paper

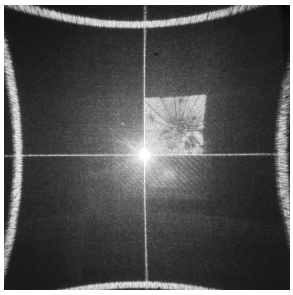


(b) Signal region positioned at the edge of the RPF

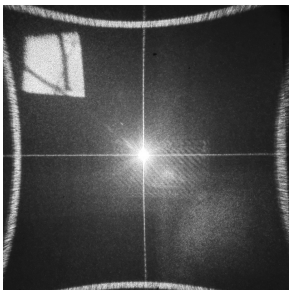
Fig. 6.27 Positioning the signal region
View inside the projection chamber, A4 page for scale

An example of this stitching method is seen in Fig. 6.28. Example tiles can be seen in Figs. 6.28b - 6.28d. Aberration correction is also demonstrated: Fig. 6.28c shows an outside tile, which suffers from aberrations, while Fig. 6.28d demonstrates the same tile with an aberration correction applied. The final, time-multiplexed image is shown in Fig. 6.28e. Some of the tiles can still be noticed, but this is caused by the imperfect timing of the frames, which results in slight differences in the brightness. This effect is eliminated in Demonstrator 3 by the insertion of a mechanical shutter controlled by an Arduino microcontroller.

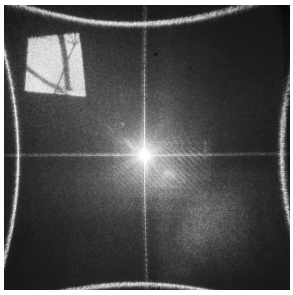
(a) A target image



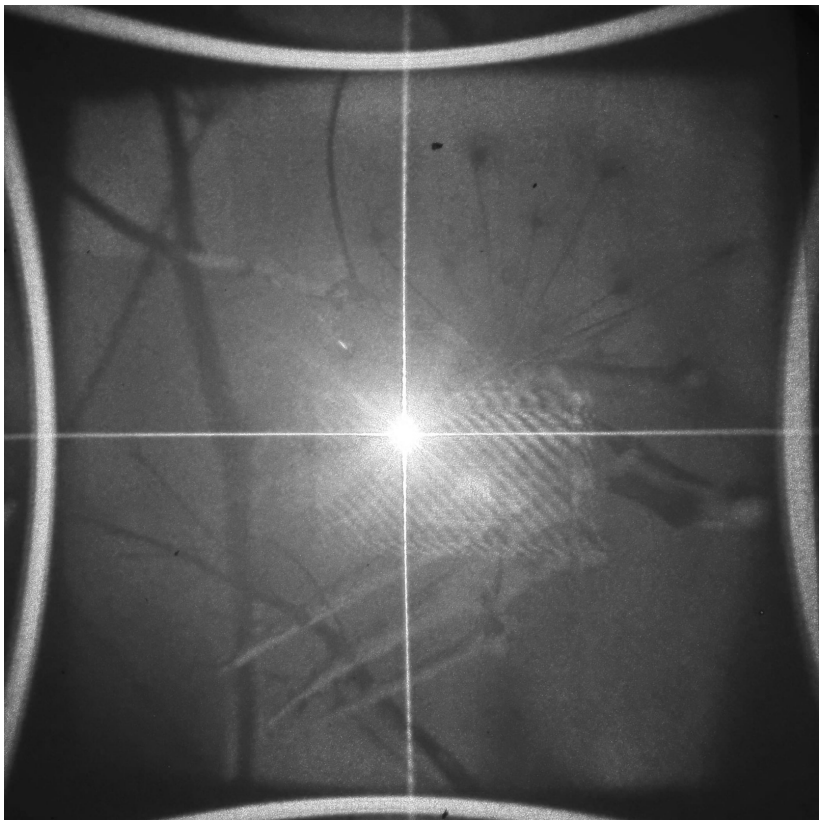
(b) An example tile



(c) Tile without aberration correction



(d) Tile with aberration correction

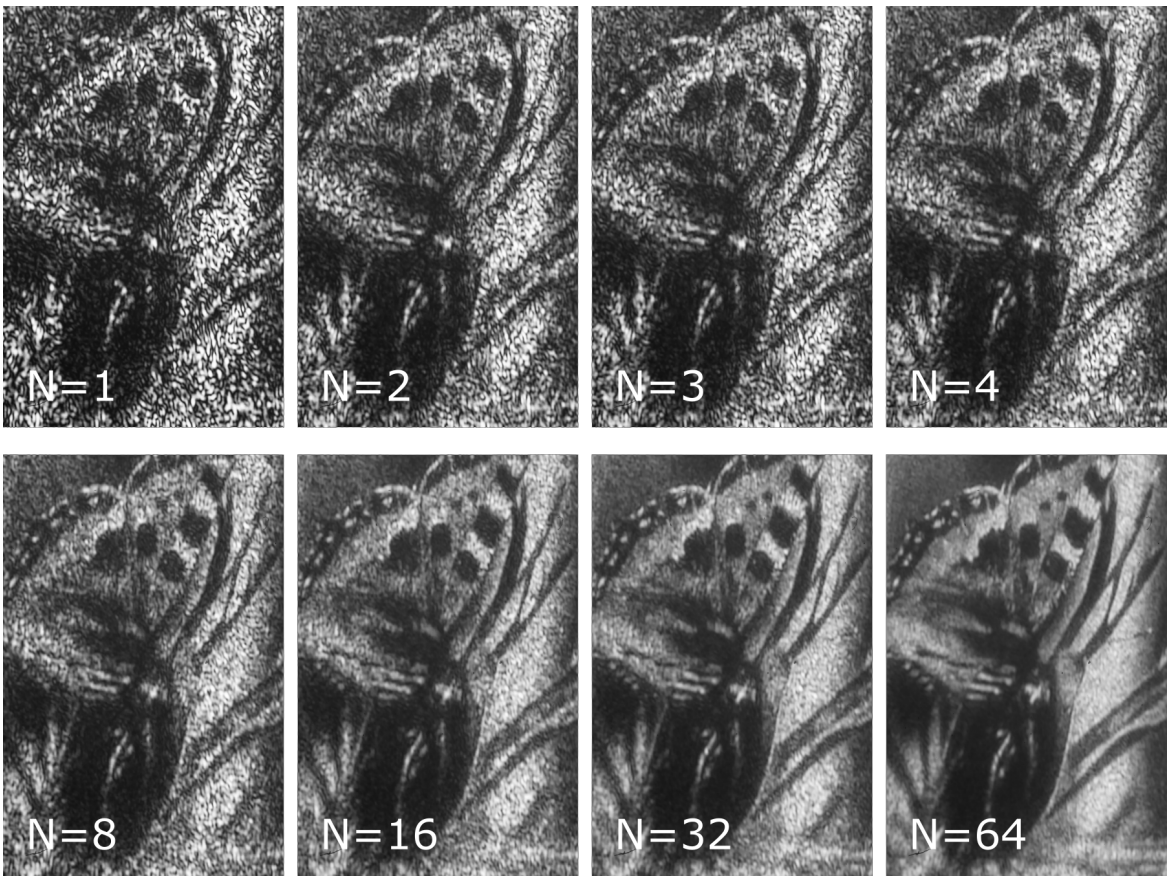


(e) Time-multiplexed images

Fig. 6.28 Tiling of the image

OSPR frame averaging

(a) Target image



(b) Averaging property of OSPR algorithm depending on the number of frames

Fig. 6.29 Averaging property of the OSPR algorithm

The effect of this procedure can be seen in Fig. 6.29. The target image is shown in Fig. 6.29a and the reconstructions depending on the number of frames N can be seen in Fig. 6.29b.

Images for $N > 16$ have slightly worse contrast, because of a small amount of background light, entering the projection chamber. Nonetheless, it can clearly be observed that at $N = 64$ the noise has been precisely suppressed.

6.8.3 Demonstrator 2

Demonstrator 1 did nearly satisfy the field of view requirement for the printing system. However, the minimum effective pixel size was still very similar to the previous LED projection system (by employing diffraction limit breaking). The investigation then turned into projecting much smaller structures with an appropriately reduced image size. Several improvements to the system's construction were implemented along a way based on previous shortcomings.

Optical design

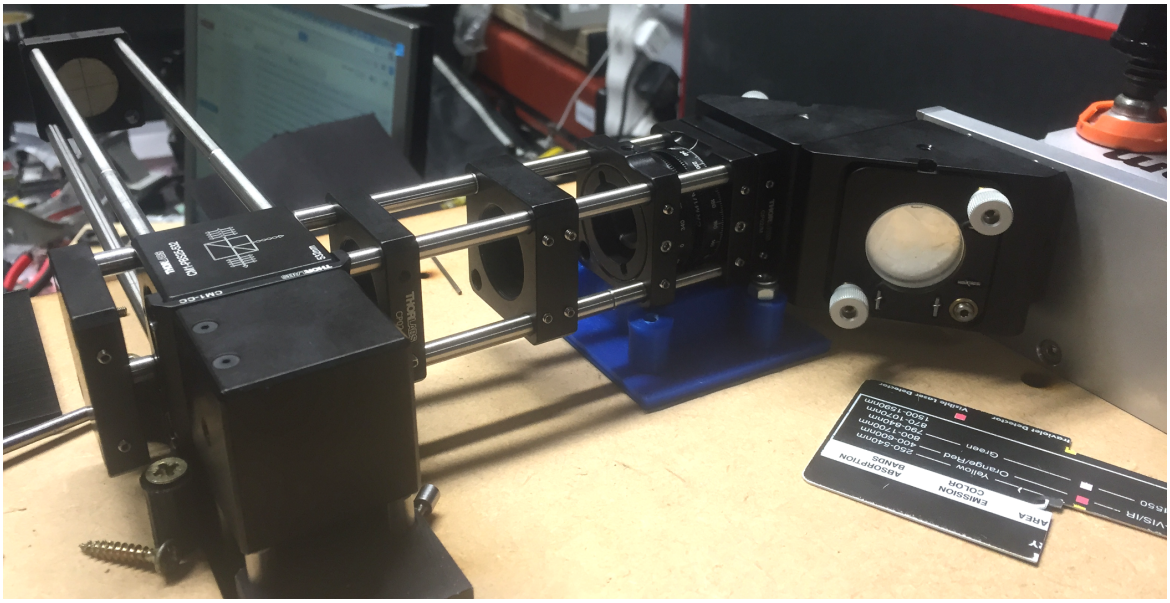
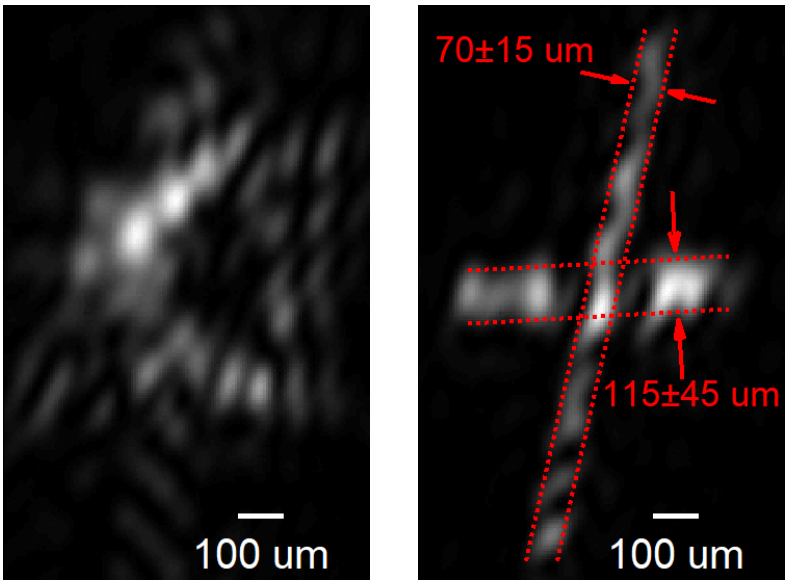


Fig. 6.30 Construction of Demonstrator 2

Unfortunately, not many pictures from the assembled projector remained. In Fig. 6.30, the initial assembly employing a set of collimating mirrors is shown. A waveplate between the beamsplitter and the SLM was also introduced at this stage to perform phase modulation, rather than amplitude modulation (and hence, increase the optical efficiency of the projector). The lenses used in construction of this projector were: $f_1 = 4mm$, $f_2 = 250mm$, $f_3 = 50mm$.

Aberration correction

The result of aberration correction in the second demonstrator is shown in Fig. 6.31. It can be seen that there is a substantial difference between the uncorrected and the corrected images. Fig. 6.31a shows a highly-aberrated structure that does not at all resemble any particular shape, while in Fig. 6.31b an intended shape of a cross can be clearly visible. By fitting a set of parallel lines to the arms of the cross, the approximate diffraction limit can be deduced. Here it is $70 \pm 15\mu m \times 115 \pm 45\mu m$. The confidence bounds of the approximation are fairly large, due to a large pixel size of the camera ($5.7\mu m$) as well as poorly-defined boundaries of the cross arms.



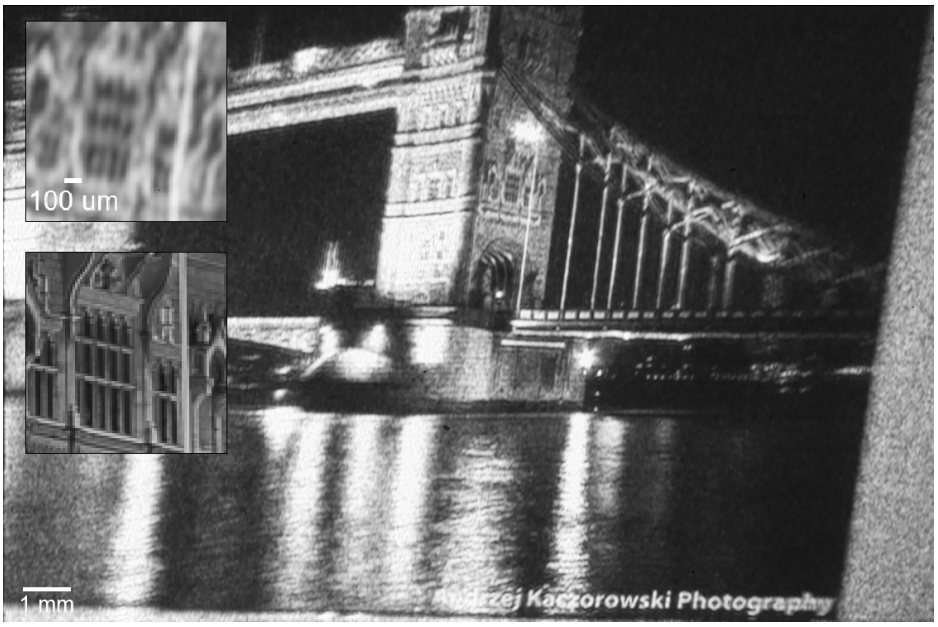
(a) Target cross - uncorrected (b) Target cross - corrected

Fig. 6.31 Aberration correction in Demonstrator 2

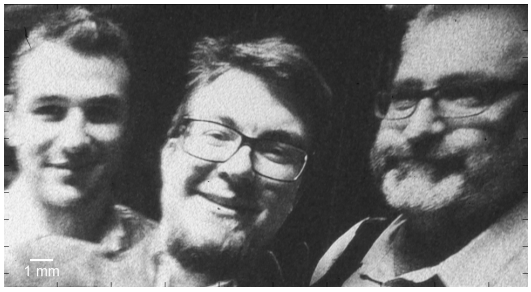
Diffraction limit breaking

The breaking of the diffraction limit can be witnessed in Fig. 6.32. For the matter of clarity, insets in Figs. 6.32a - 6.32b show the enlarged regions of interest together with the same region from the target image. In order to find an approximate feature size of this projector, the section consisting of three identical parallel lines was found [Fig. 6.33a]. A cross-section through the red line can be seen in Fig. 6.33b. That three-line structure had a measured length of $182 \pm 6\mu m$ and hence, the half-width of the smallest resolvable structure was calculated to be $30 \pm 1\mu m$. Comparing it with Fig. 6.31b, it can indeed be seen that this is less than a half of the diffraction limit. Also, unlike the cross, the projected image has well-defined smooth edges.

(a) Target 1: Cambridge

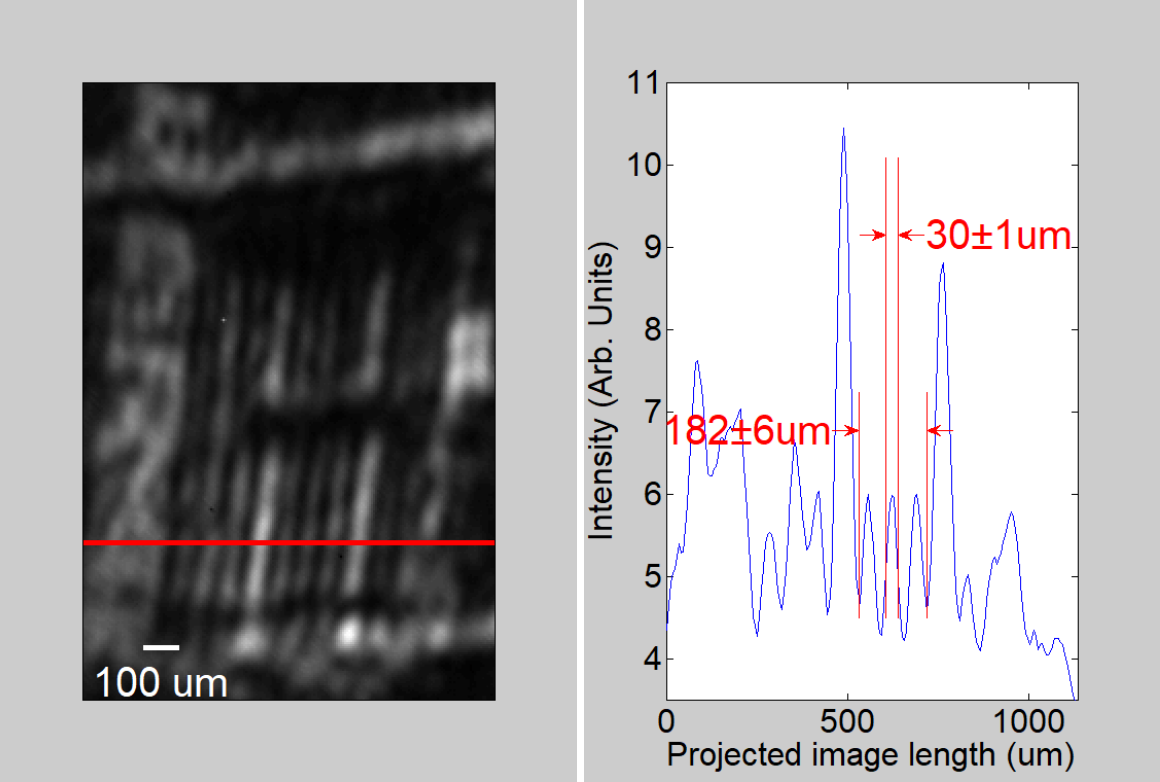


(b) Target 2: London



(c) Target 12: Selfie

Fig. 6.32 Test images captured through Demonstrator 2 projector



(a) Cropped part of the replay field (b) Section through the replay field

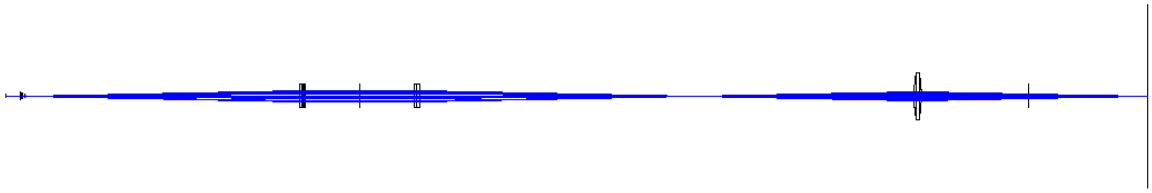
Fig. 6.33 Demonstrator 2: Resolution analysis

6.8.4 Demonstrator 3

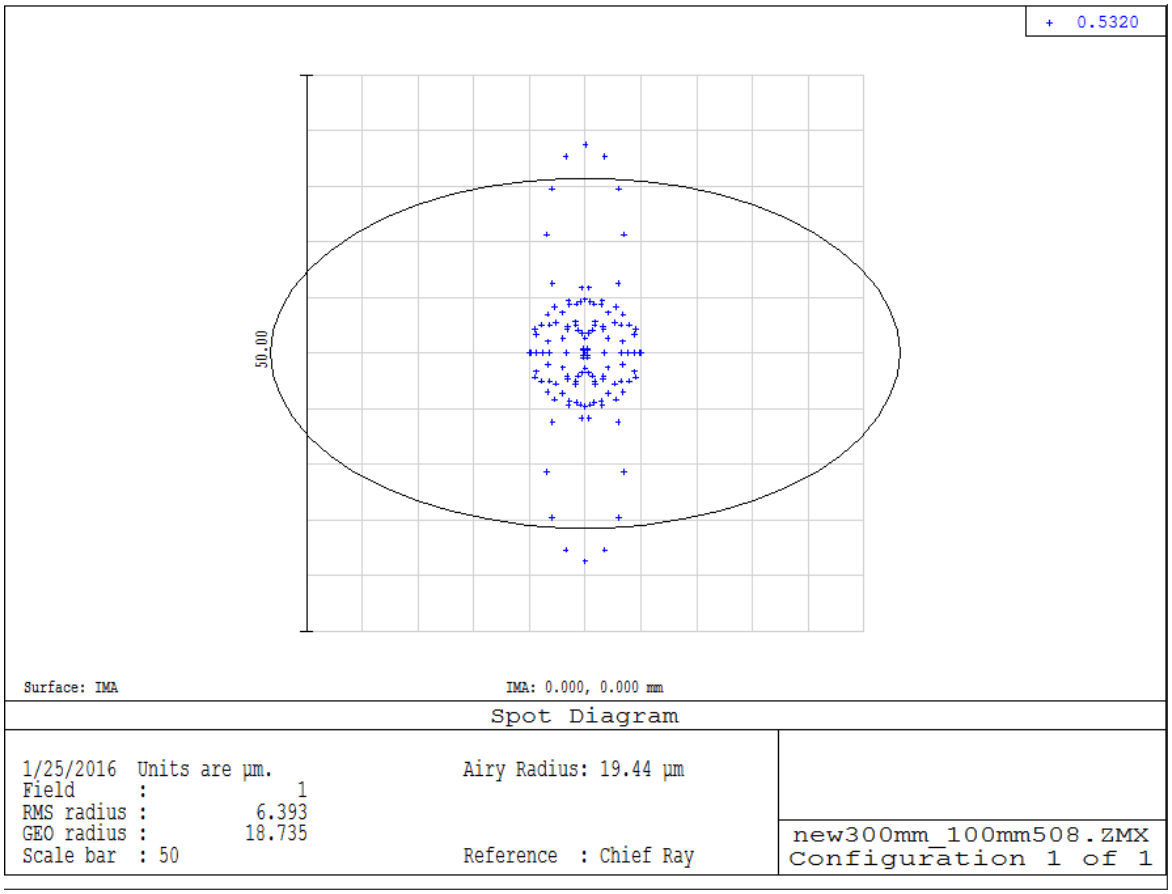
The reason behind the construction of the third demonstrator was to test whether the diffraction limit can be broken to the same extent as in the Demonstrator 2. The field of view has again been reduced to approximately $30mm \times 30mm$ hoping to display structures a few micrometers in size.

Optical design

At the time of Demonstrator 3, the shutter has been inserted into the optical system. The lenses used in the construction were $f_1 = 4mm$, $f_2 = 300mm$, and $f_3 = 100mm$. The final Zemax design together with a spot diagram can be seen in Fig. 6.34. It can be seen that the diffraction limit of this projector is approximately $32\mu m \times 55\mu m$.



(a) An optical system simulated in ZEMAX



(b) A spot diagram

Fig. 6.34 ZEMAX simulations of Demonstrator 3

Field of view

The field of view seen by the dSLR is presented in Fig. 6.35. The zero order spot is visible on the right-hand side as well as various reflections coming from optical components. The overall size of the image is the same as the size of the APS-C CMOS sensor of the camera ($20.2mm \times 13.5mm$). The super-resolution window of size approximately $8mm \times 6mm$ is visible inside.

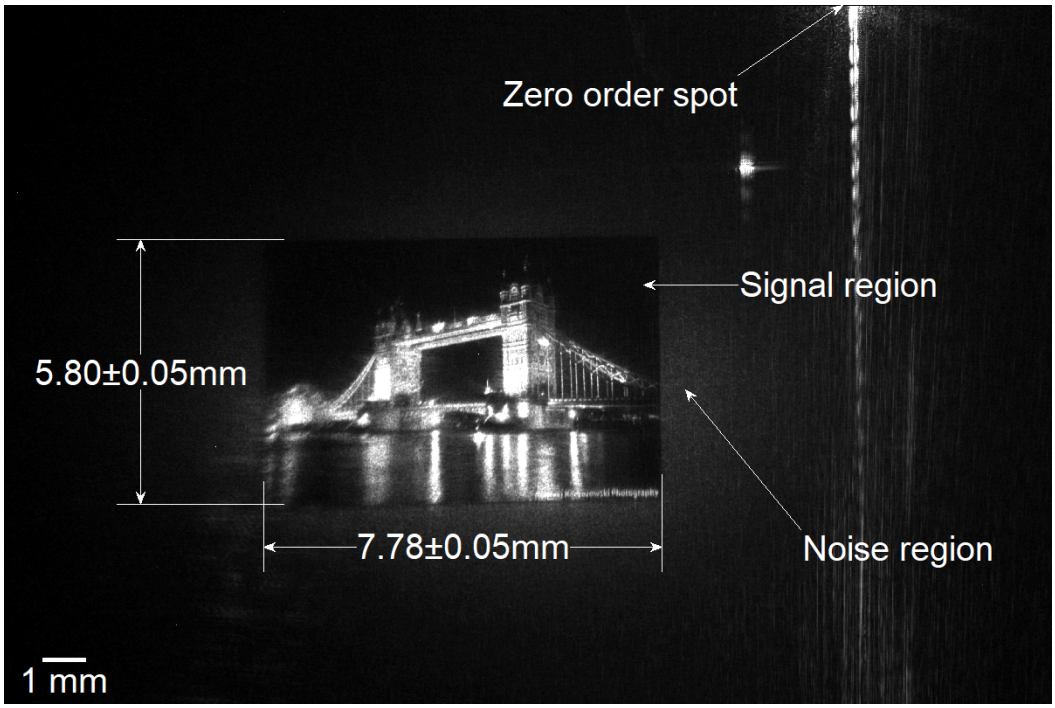
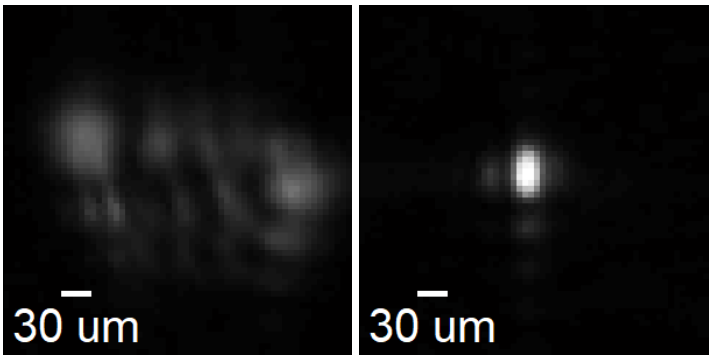


Fig. 6.35 Image structure and size of Demonstrator 3

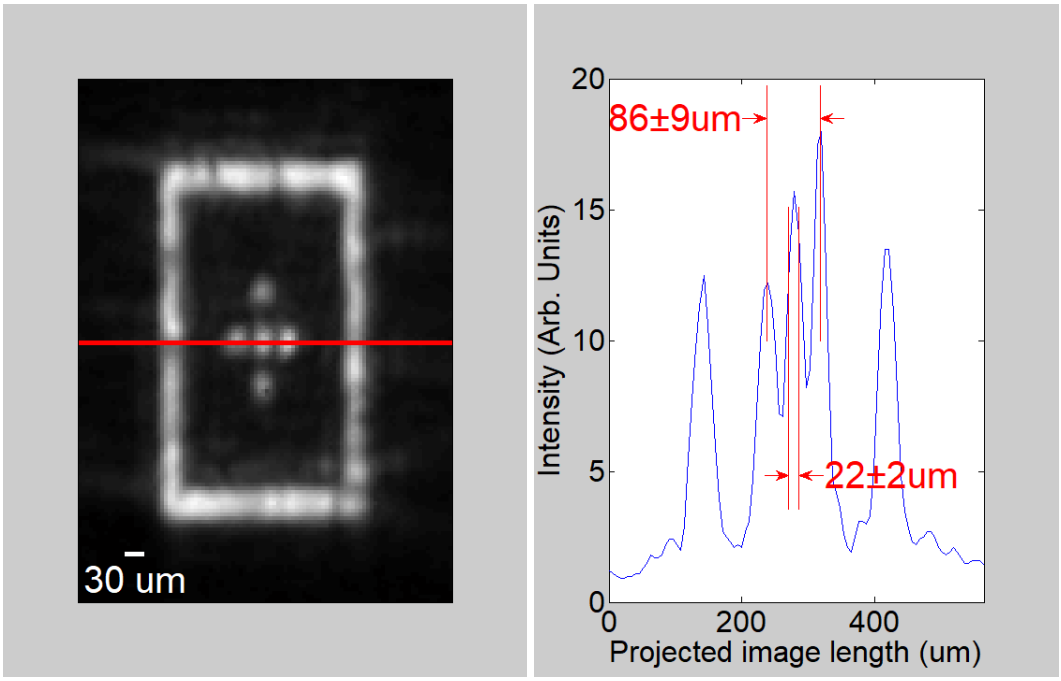
Aberration correction

Aberration correction in the Demonstrator 3 was a very challenging task for a number of reasons. First, the diffraction limit of the system was approximately 30 micrometers in size, which corresponded to just 5 dSLR pixels. Since the dSLR used a Bayer mask, an additional blur was introduced to the acquired image, which further complicated the operation. A considerable amount of time was spent optimizing and improving the correction. The final result consisted of 10 different phase masks, which have been manually inspected for image quality. One of the corrections can be seen in Fig. 6.36. A single spot seen in Figs. 6.36a - 6.36b is directly taken from the final run of the optimization and was acquired in LiveView mode, while the image in Fig. 6.36c was taken after the optimization has been finished. It is not possible to study the diffraction limit based on a single spot. As previously mentioned, the uncertainty of such estimate would be far too large to give any conclusive result. Instead, another method is employed based on the 5-point, diffraction-limited structure [Fig. 6.36c]. The structure is composed of 5 spots, separated by an empty pixel inbetween each 2 spots. Therefore, by measuring the distance between the centres of two outermost spots, one can measure the pixel spacing with greater accuracy. As can be seen in Fig. 6.36d that this length is equal to $86 \pm 9 \mu m$, and hence, the single pixel spacing is $22 \pm 2 \mu m$. This is even smaller

than the figure predicted by ZEMAX. It is postulated that the pixel spacing in holographic projection and the spot size are not exactly equivalent and this is where the error comes from.



(a) Uncorrected spot (b) Corrected spot



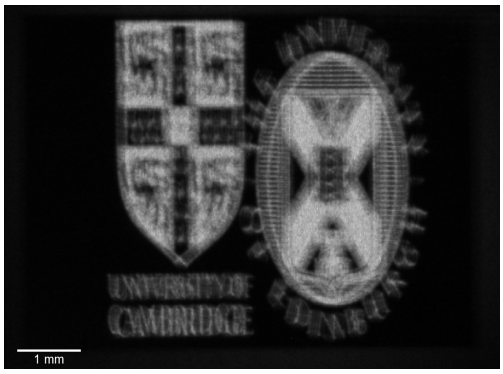
(c) Diffraction-limited image (d) Cross-section through the image

Fig. 6.36 Diffraction limit of Demonstrator 3

Diffraction limit breaking

The combined aberration correction results applied to diffraction limit-breaking holograms can be seen in Fig. 6.37. Figs. 6.37a-6.37b show the University Crests before and after aberration correction respectively, while Fig. 6.37c shows the specially-designed resolution test target with aberration correction already applied. It can be seen that the images, inspected

on a large scale are indeed both high-quality and resolution. The detailed resolution analysis of patterns from Fig. 6.37c is presented subsequently.



(a) Target: University crests, No aberration correction

(b) Target: University crests, Aberration correction

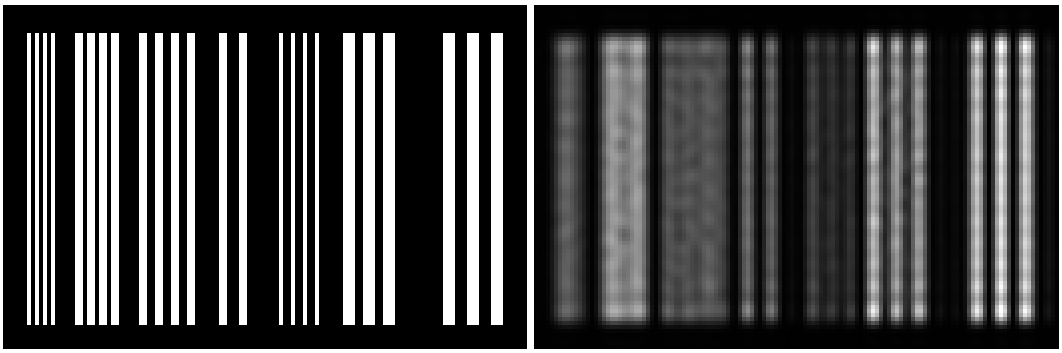
(c) Resolution test binary target

Fig. 6.37 Test images taken through the Demonstrator 3 projector

Resolution analysis

In order to assess the quality of the projected image, an effective projected pixel size has to be measured. Two methods are presented here. One is based on the image captured by the dSLR. However, the pixels of the dSLR used were $5.7um$ in size and hence it should be kept in mind that this approach is severely limited. Therefore, another method is proposed. It is possible to simulate the replay field of a given time-averaged hologram and find which sets of lines

are distinguishable inside of the hologram. Then, assuming the perfect aberration correction, the lines that can be distinguished within the hologram should also be distinguished in the projected replay field.



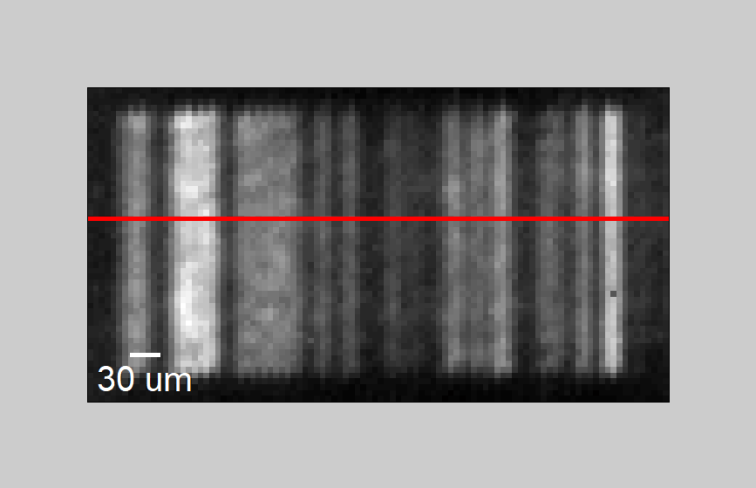
(a) Desired replay field (b) Simulated replay field

Fig. 6.38 Studying the resolution of Demonstrator 3

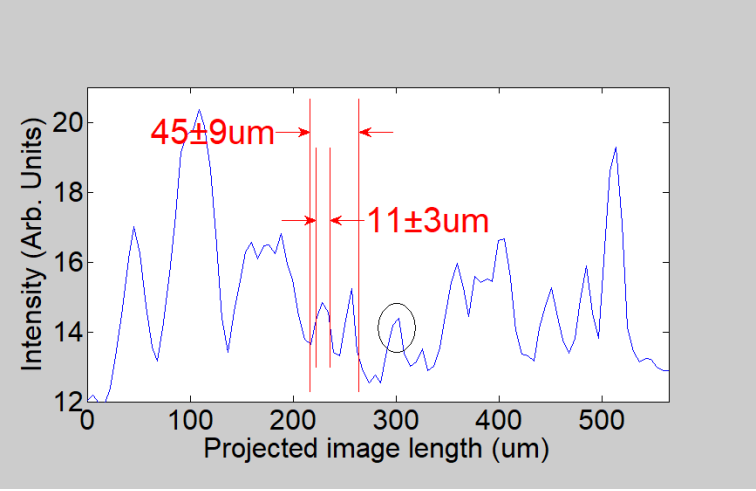
The length of the signal region was measured to be 1364 ± 8 dSLR pixels, which corresponds to $7.78 \pm 0.05mm$. The signal window within the FFT is 1940 hologram pixels in length, hence the size of the single FFT pixel is equal to $4 \pm 0.03\mu m$. Figs. 6.38a-6.38b show the input and simulated replay fields respectively. The simulation indeed indicates that certain pairs of lines are projected in such proximity to each other, that they cannot be resolved. This effect is observed for the first two groups. The third group is currently still impossible to resolve. However, the dips in the intensity indicate that it is just at the boundary of this hologram's resolving capabilities. Given that two parallel lines can be distinguished when their centres are 4 hologram pixels apart (as seen in Fig. 6.38), the half-width of the smallest displayable structures is equal to $8 \pm 1\mu m$.

Now, let us compare that prediction to the actual replay field measured through the camera. Fig. 6.39 summarises and compares the results. The first thing that should be noted is that both the simulated [Fig. 6.39a] and the actual [Fig. 6.39b] replay fields follow the same pattern, i. e. the lines being the brightest in the simulation are also the brightest in the actual measurement. There is a substantial amount of blurring and a bit of noise in the measured RPF due to a large pixel size compared to the actual structures as well as possibly still imperfect aberration correction. The slice through a replay field has been taken and is shown in Fig. 6.39c. It can be seen that the two lines from the 4th group when measured together are $45 \pm 9\mu m$ across, and hence the half-width of that structure is equal to $11 \pm 3\mu m$. We can also see a very faint line, which corresponds to a 5th structure in the simulated RPF. One can assume that given better equipment, also this set of lines could be distinguished.

(a) Simulated RPF



(b) RPF captured with a dSLR

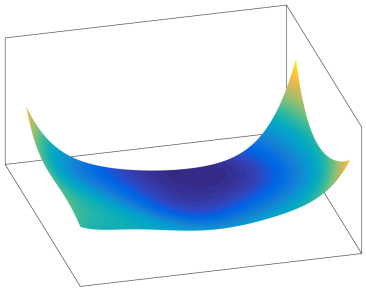


(c) A slice through the captured RPF

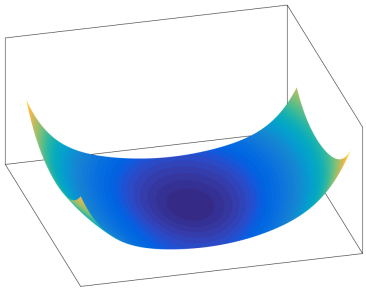
Fig. 6.39 Demonstrator 3: simulated and the actual replay field

Spatial variation of aberrations

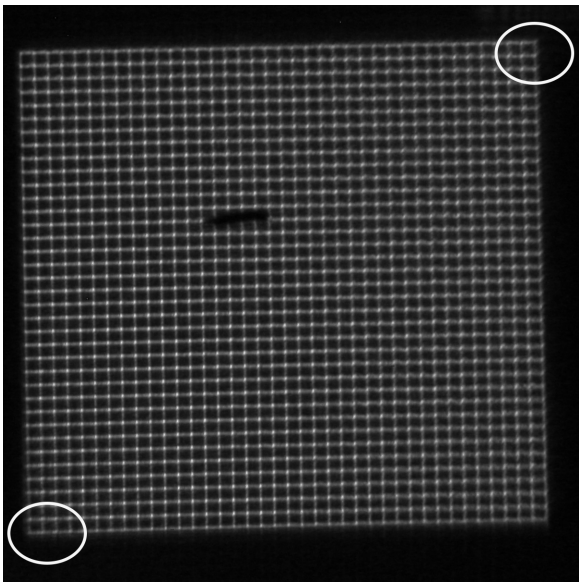
An interesting observation has been made while projecting an image of a high-resolution grid with different phase corrections. Although all the corrections have been performed on the same replay field position, different corrections prove to correct well only a particular replay field region. This is the best example of the spatially-varying aberration phenomenon. The two masks used for corrections are presented in Figs. 6.40a - 6.40b and the respective replay fields in Figs. 6.40c - 6.40d.



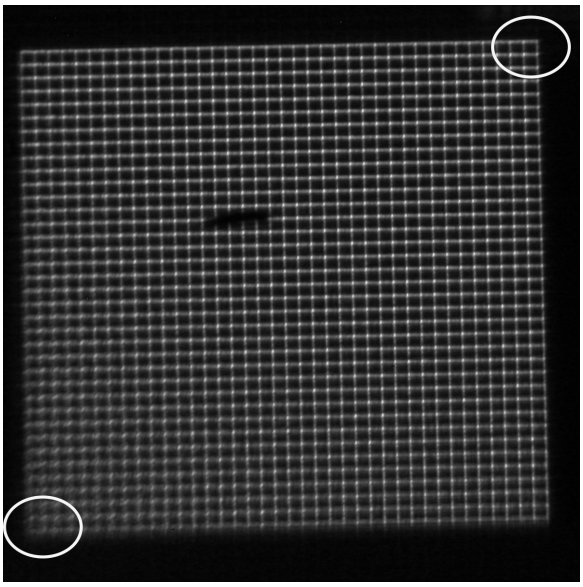
(a) Phase mask 2



(b) Phase mask 5



(c) PPF of the grid with a 2nd correction



(d) RPF of the grid with a 5th correction

Fig. 6.40 Spatial variation of aberrations

It can be seen that the two corrections are very similar to each other, but the correction from Fig. 6.40d is substantially inferior in the bottom-left corner, while the mask in Fig. 6.40c corrects the image well in this region, but is worse in the top-left corner.

6.9 Conclusions

The proof-of-principle cost effective system for maskless holographic lithography was presented and tested. It can be noted that as the field of view is reduced, the feature size of patterned structures can be shrunk accordingly. Three demonstrator projectors were tested:

- A wide angle printing prototype, with a field of view of $20cm \times 20cm$ and a feature size of approximately $120\mu m$
- A high-resolution photo printing prototype, able to project a field of view of $10cm \times 10cm$, having the feature size of approximately $30\mu m$
- A maskless lithography system with a field of view of $3cm \times 3cm$ capable of projecting structures $8\mu m$ in size

Various advanced techniques were outlined, namely the advanced adaptive distortion and aberration correction, diffraction image breaking and image tiling.

6.9.1 Replay field size and artefacts

The current system is able to display tiles of resolution 1920×1280 of which at least 960×640 can be treated as distinguishable. These tiles can be displayed at any position among the 9600×5400 addressable replay field pixels.

Certain artefacts in the replay field has not fully been eliminated. The zero order is significant and divides the replay field into four quadrants. These areas cannot currently be used for patterning.

There remained a small number of reflections from the optical components. It should be possible to eliminate these completely, once high quality optical components employing anti-reflection coating are purchased.

6.9.2 Digital correction

The correction algorithms presented proved to work very well, even in the most difficult case of Demonstrator 3. Once even smaller structures are being patterned, it will be necessary to upgrade the imaging device.

Distortion correction in Demonstrator 1 did not work very well. It is certain that it was a human mistake rather than the algorithmic error, because the same method proved to work very well applied to the other projector.

Tiling approach also proved successful, showing no significant errors.

6.9.3 Relation to the work of others

Even in the current state, the system proved to work better than any of the approaches reviewed previously in terms of the number of distinguishable pixels. If the tiling mechanism is employed (as outlined in the future work section) the total number of pixels can be increased to $4K \times 2.5K$. The diffraction limit has been broken around 3 times, which is indeed an impressive result. The only disadvantage of the current system is that the noise in the replay field is still significant, even in the case of time-averaging of 64 frames. Again, the methods to improve it are discussed subsequently.

6.10 Acknowledgements

The assembly of the demonstrator projector was funded by the University of Edinburgh Proof of Principle Fund, awarded and administrated by Edinburgh Research and Innovation (ERI). The mechanical design and manufacturing of the projector chamber was carried out by Constantine Tatalaev from the University of Edinburgh FabLab+. The Spatial Light Modulator used in this project was a kind loan from Professor Ian Underwood, University of Edinburgh. The funding for a two-week Edinburgh visit was provided partially by the Engineering and Physical Science Research Council via The Centre for Doctoral Training and partially by Jesus College.

Chapter 7

Conclusions and future work

7.1 Conclusions

A broad topic explored throughout this thesis is the adaptive holographic correction suited for display applications. Most of the errors found in displays can be characterized and corrected using methods devised here.

Aberration correction is performed by blind, sensorless optimization based only one the projected image. The image is characterized using an automated fitness function mechanism. Two algorithms suited for retrieving the phase error of the projector are presented. The discussed algorithms utilize the Zernike Polynomial expansion of the phase mask. Using a combination of Genetic and Steepest Descent algorithms, the correction parameters at a particular position of the replay field can be retrieved. As the algorithms are iterative, the corrections improve the longer the algorithms are ran. It was usually found that 5 hours is more than sufficient to find a very high-quality correction.

As a rather interesting add-on to the algorithm, was the realization that the optical aberrations in the centre of the field come primarily from the non-flatness profile of the Spatial Light Modulator used. Therefore, characterizing aberrations using the presented algorithm at the centre of the field implicitly characterizes the non-flatness of the SLM. That hypothesis was tested by comparing the output of the algorithm to the previously measured non-flatness using interferrometric methods. The outputs indeed match quite closely, confirming the validity of previous considerations.

A method of characterizing and correcting distortion is also described. For the time being, only cylindrically-symmetric distortion is dealt with, as this is the most popular type of distortion dealt with. However, the presented formalism allows incorporating other types of distortions in the future.

Given all of the errors characterized, a novel algorithm suited to correct these is described. It is a combination of two previously known algorithms. Pixel to Wrapped Phase Summation previously developed by Freeman was capable of correcting all the errors, but was very cumbersome in operation. Due to the basic nature of the algorithm, the calculations are extensively lengthy. On the other hand, a One-Step Phase Retrieval algorithm was able to produce a high-quality image in real time, but was not suited for correcting all the image imperfections. To address this issue, a variant of OSPR incorporating distortion and spatially-varying aberration correction was constructed. It appeared that diving the replay field into a small number of regions, the majority of aberrations can be eliminated. The algorithm, programmed using highly-parallel GPU programming allowed to generate holograms at a frame rate of 12 fps given all the corrections.

Throughout this thesis, the emphasis has been put on cost-effective solutions. An entire feedback loop consists of reasonably inexpensive components, such as a 10$ webcam or a standard, general-purpose dSLR. The remaining elements of the setup were either assembled from unused parts laying around the lab, or constructed with minimal costs (such as the microcontroller board). The rail mechanism was the only one, which has been custom-made for the purpose of the project.

The subsequent chapter is the result of a collaboration with Dr Phillip Hands from Edinburgh University and Trevor Elworthy from LumeJET photo printing company and is devoted to digital printing and maskless holographic lithography. The objective of the project was to test whether holographic projection methods developed in this thesis are capable of displaying high quality and high-resolution image. A number of novel techniques is presented, namely image tiling, diffraction limit breaking and replay field averaging. A number of demonstrator projectors, comprising off-the-shelf optical components demonstrating different field of view and image resolution. The final demonstrator achieved a spot size of $7.4\mu m \times 13\mu m$ while being able to pattern a wide field of $30mm \times 30mm$. A number of further enhancements is proposed.

The topic of 3-dimensional holography is studied subsequently. As it appears, the mathematics of calculating a 3D holograms can be shown to be equivalent to 2D hologram computation in the presence of spatially-varying aberrations. Using this observation, generation of 3D holograms can be substantially speeded up using a previously developed GPU implementation. To demonstrate the working of this algorithm, a prototype of holographic teleconference transmission system is demonstrated. A 3D model of the environment is first acquired using an XBox Kinect sensor, the hologram is calculated on the GPU and displayed in real-time on the spatial light modulator. This project was completed while the author was working part-time for Penteract28 Ltd, in collaboration with multiple researchers

(project students: Shengjun Ren, Vamsee Bhemireddy, Mikolaj Kosinski, Pawel Mackowiak, co-supervised by Dr Darran Milne).

7.2 Future work - Improvements of current methods

A number of improvements to the existing methods is already proposed. The improvements are algorithmic, as well as computational. Majority of this work, being a proof-of-principle carried on by a student, has not reached a limit of well-designed, optimized product.

The aberration-correction algorithm, although in every single case proved to provide a high-quality correction, still takes a significant amount of time to converge (5 hours). It is very likely that, when more time is spent on both: the algorithmic optimization and more efficient coding, this time can be significantly shortened. A detailed list of suggested improvements can be found in the next section.

The entire correction is currently semi-automatic and requires a person to operate a set of automated scripts as well as to monitor the results. However, nothing in the design of the algorithms prevents the full automation, once better components are purchased.

The particular type of GPU used for 2D real-time hologram generation was a standard model (nVidia GTX 760). The relatively small amount of RAM as well as relatively modest computational resources meant that the implemented algorithm had to obey all these limitations. The algorithm implemented was a standard, 8-frame OSPR at a native resolution and a nearest-neighbour distortion correction. When the GPU is upgraded, it will be possible to implement other variants of the same algorithm, for instance an Adaptive OSPR at a doubled resolution up to 24 OSPR frames, which will further enhance the quality of the projected image.

7.2.1 Adaptive Optical Mechanism

The system presented above is just an example of a feedback loop mechanism. The design of it was influenced by various hardware limitations. While running the algorithm, the need for improvement of few aspects of it became evident. Below, these postulated improvements are presented.

Limiting the solution space

The space in which the correction can be found can be virtually infinite. However, in reality, that vast space can easily be limited in order to speed the convergence time. Currently, the

maximum and minimum of each of the coefficients is set manually to $-3 \ldots 3$. These bounds proven to be large enough to find the correction in each case.

However, within the set bounds, there are still regions that will certainly not lead to the correction. Removing those regions will speed up the convergence time.

Limiting all the coefficients independently

The naive optimization would be to sample each of the Zernike coefficients and assign the limits around the place where the minimum of this function resides. But, a preliminary study revealed that this procedure might lead to erroneous results.

Randomised heuristic descent optimization

The proposed improved method is a heuristic descent optimization in a number of random starting positions. The small number of iterations of the HD algorithm is likely to return a point which, after further optimization, might lead to a global optimum. At the same time, with a random starting point, different regions of space are likely to be explored. Then, the maxima and minima of each of the Zernike coefficients can be calculated.

Software improvements

Being a low-budget prototype, the feedback loop code is still on the early development stage. Whenever this system is implemented on a larger scale, there is a multitude of improvements, which can be implemented, such as:

- Full-automation

  Because of hardware limitations, some of the procedures are done manually (for instance the positioning of the camera, brightness adjustment, etc.). These can be straight-forwardly automated, provided better equipment.

- Task multiplexing

  The MatLAB control script is naturally a single-threaded environment, where tasks are executed one after the other. A better solution would be to utilize a multi-threaded architecture. To give an example, hologram generation can be easily overlapped with picture taking, leading to substantial speed-ups.

- Integrated, rather than modular solution

  A current architecture is separated into modules, each written in another programming language. The objective of this approach was the time-efficient implementation and

easy debugging. By integrating all of the modules in one program and utilizing low-level C++ architecture, further speed-ups can be achieved.

Algorithm optimization

The algorithm presented here contains a variety of parameters, such as thresholds and multipliers. These parameters were fine-tuned by trial and error method and adjusted throughout the multiple run-times of the algorithm. The values obtained proven to work in the majority of cases. Nonetheless, it is not certain that they are optimal for all of the situations. To find the optimum values, a simulation of the feedback loop is necessary. This simulation could be designed in the following manner:

- The pattern, coming from an aberrated replay field can be simulated from basic theory
- The camera response curve should be taken into account to simulate how such a shape would be registered through a particular CCD device
- All of the aforementioned coefficients should be optimized such that the distance from the optimal correction correlates well with the value of the actual fit function.

7.2.2 Holographic Lithography - Improved Optical Design

A number of improvements was postulated in the preceding chapters. Here, all of them are summarised and the design of the future lithography system is presented.

Laser wavelength

The laser used in this research was a green $532nm$ DPSS laser. The replay field of the holographically-generated images depend linearly on the wavelength. Decreasing the laser wavelength therefore shrinks the overall size of the replay field, and hence, the size of projected structures. Therefore, choosing the wavelength to be in the UV region would imply projecting smaller structures.

Spatial Light Modulator

The spatial light modulator used in this research was able to display holograms at a full HD resolution(1920×1080). In the recent years, a number of manufacturers began introducing SLMs of higher pixel count displays. A few possible options to consider are Nematic $4K2K$ displays with a pixel pitch of $3.40\mu m$ introduced by Jasper Display Corporation and

Holoeye photonics and 2K1.5K ferroelectric displays introduced by 4th Dimension Displays. Increasing the pixel count of the hologram is going to increase the sampling of the replay field, and hence will lead to better RPF resolution.

Selective replay field shutter

The tiling mechanism relies on blocking the noise region. This can be achieved with either a simple filter or a ferroelectric replay field shutter. A simple prototype was designed in the Demonstrator 3, a mount for the RPF filter was designed and 3D printed. A piece of plastic, blocking the unwanted light would then be inserted inside. This mechanism was not finally tested due to the project's time constraint.

Improved ZEMAX design

The current design optimizes only one part of the replay field for a desired diffraction limit. A substantial amount of spatially-varying aberrations can be observed in the projected images. For precision printing, the aberrations have to be suppressed throughout entire printing area. While for the cost-effective system it might not be possible to eliminate the aberrations fully at the design stage, one might attempt to minimize the spatial variation of aberrations. A thorough ZEMAX simulation would be a point to start. Rather than optimizing one field position, a number of positions should be considered, ensuring that the aberrations parameters optimized for are applicable to a larger area.

Non-square replay field

A current system has a non-square aperture of the SLM which results in a different diffraction limit in X- and Y- directions. For some applications, it is not a substantial issue. However, for photo printing in particular, the desired pixels should be square-shaped.

A potential solution to this problem would be to use a combination of a spherical and a cylindrical lens. Assuming the thin lens approximation and the focal lengths: f_3 and f_{cyl}, the focal length of this two-lens system would be f_3 along one of the directions and $\left({f_3}^{-1} + {f_{cyl}}^{-1}\right)^{-1}$ along the other. As the RPF scales linearly with the focal length, this will result in changing the aspect ratio. In order to achieve a required square diffraction limit of the system given a non-square aperture of the SLM, the replay field size should have an aspect ratio of $\frac{M}{N}$. Rearranging the above equations, one arrives at a condition for f_{cyl}:

$$f_{cyl} = \frac{M-N}{N} f_3$$

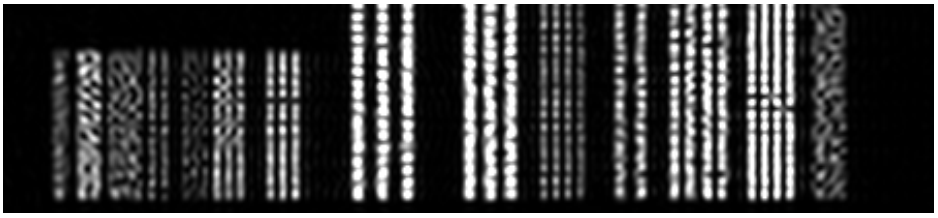
The introduction of the cylindrical lens will certainly introduce additional effects into the image, such as shifting the focal planes of the RPF and introducing astigmatism into the wavefront. These effects will have to be corrected digitally.

7.2.3 Holographic Lithography - Computational Techniques

Temporally Separated Pixels method

The Adaptive-OSPR approach naturally works with the same target image for all the projected frames. However, as it was shown while discussing the resolution of the target, some configurations of parallel lines simply cannot be displayed when positioned too close to each other (an effect indicated previously by Bay [89] and Freeman [56]).

(a) Desired replay field



(b) Simulated replay field of a single hologram frame

Fig. 7.1 Resolving power of a single hologram frame

It can be seen in Fig. 7.1 that two parallel lines 2 pixels in width cannot be distinguished when they are separated by less than 4 pixels. Looking at sets of 1 pixel in width lines, one can also think they might be resolved when spaced further away from each other, but this has not yet been attempted.

Another issue, which manifest itself in Fig. 7.1b is the fact that densely-spaced lines have a smaller intensity compared to other sets of lines.

Both of these problems can be resolved when closely-spaced parallel lines are moved to adjacent frames. There are two ways in which this procedure can be achieved. An obvious one involves manually or algorithmically separating close pixels (either by manual placement of lines or using a TSP grid, as outlined by Bay[89] and Freeman [56]. A more general approach would be to construct an improved hologram generation algorithm.

Since the desired quantity is the total averaged replay field, rather than a single frame, instead of using Gerchberg-Saxton approach, one can employ a more generic Direct Binary Search with Simulated Annealing optimization. Once constructed properly, the algorithm would ensure that the best configuration of amplitude and phase is used to give a desirable total time-averaged replay field.

Spatially-varying aberration correction

A future lithography system will certainly have to account for the spatial variation of aberrations. The method to correct these has already been presented using Adaptive-OSPR approach at a native and double resolutions. The same optimization should be applied to the windowed super-resolution algorithm. The time constraint of this project did not allow for studying this phenomenon in greater detail.

Simulating achievable grey-scale resolution

At the time being, high-resolution images are displayed and have the noise suppressed using frame averaging. However, it is not yet known what is the exact grayscale resolution of projected patterns. This phenomenon should be studied in more details in order to find if the projected images meet the criteria.

7.3 Future work - Projects carried within Penteract28 Ltd. (currently VividQ Ltd.)

The field of holography is so vast, that it is impossible to explore it deeply enough. During over 4 years of the holographic journey, the author had a chance to sample a number of ideas, that, because of time constraints, could not be investigated thoroughly enough to go into the main body of the thesis. This and the following sections are collections of such ideas, that when investigated further might lead to new exciting research and eventually to advancements in the field.

The projects discussed in this chapter are the original ideas of the author. They have been developed by project students: Ziheng Xiang, Vamsee Bheemireddy, Shengjun Ren, Mikolaj Kosinski and Pawel Mackowiak. This research was supported financially by Penteract28 and co-supervised by Dr Darran Milne. Great care has been taken to indicate the contributions of people other than the author. It should be emphasised, however, that the material presented here is still in its early research stage and some claims might require further validation.

7.3.1 3D hologram viewed as a spatially-varying optical aberration

In general, 3D holography is a field quite separate from 2D holography. There is, however, an intriguing parallelism between 2D holograms in the presence of spatially-varying optical aberrations and 3D digital holograms. The aberration-correcting holograms are generated by introducing an additional phase component into the wavefront. Following a similar thought experiment as A. Cable [40], that the phase component can be visualised of as a virtual aberrating Fresnel lens, aberrations of which ideally cancel out the optical system's aberrations at a particular position.

A 3D hologram is often visualised as a collection of virtual Fresnel zone plates superimposed on each other. The same formulae and in fact, the same code and its optimizations can be applied to generate these seemingly different types of holograms. Below, a thorough derivation of such statement is presented.

Hologram generation using Fresnel Slices method

The method of Fresnel Slices, or the layer-based method as it is sometimes called [40], constructs the 3-dimensional image of the complex objects by slicing them at a number of distances and focusing each slice at different depths. It can be visualized in Fig. 7.2. For the clarity of explanation, a small number of slices (16) is presented. In reality, this number is much bigger (64-256), leading to better object reconstruction.

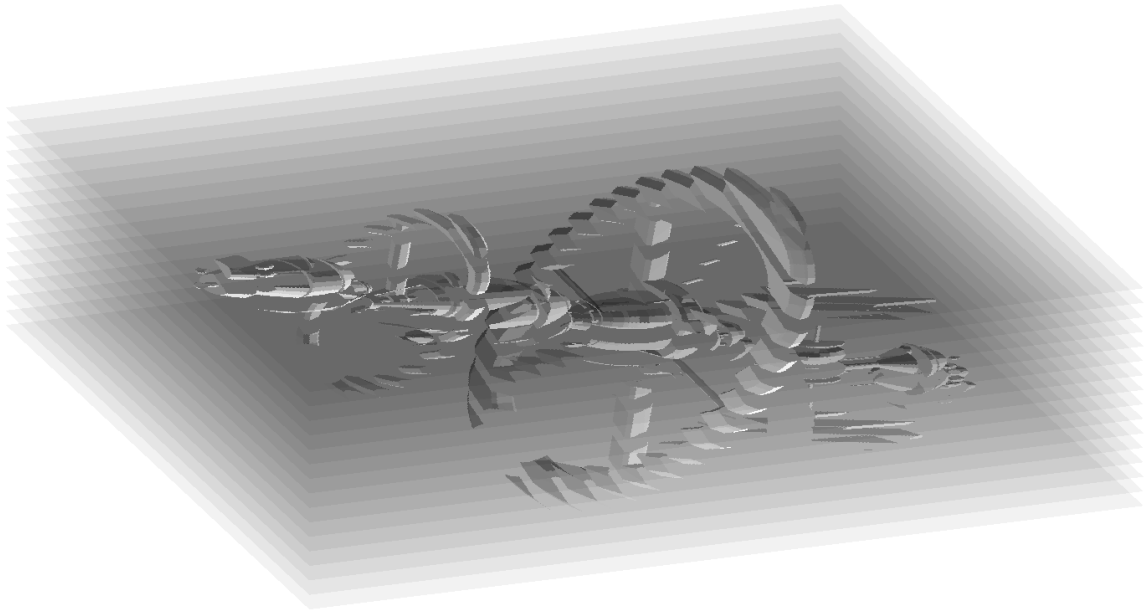


Fig. 7.2 Slicing of the object, visualised

First, it is shown that a PC-OSPR implementation can be used to generate holograms using the Fresnel slices method [1]. Rearranging Eq. 2.2, we find that the contribution to a hologram, coming from a single layer of the object $\psi(u,v,z)$ at a fixed depth $z = z_q$ is:

$$H_q(x,y) = e^{-i\frac{\pi}{\lambda z_q}\left(x^2+y^2\right)} \mathscr{F}^{-1}\left\{ \psi(u,v,z_q)\, i\lambda z_q\, e^{-ikz_q}\, e^{-i\frac{\pi}{\lambda z_q}\left(u^2+v^2\right)} \right\}$$

To simplify this formula, the following observations can be made:

- For display applications, the phase of the object field is far less important than the amplitude. It is a common practice to use random phase in order to equalize the amplitude spectrum of the hologram.
- Factor $i\lambda z_q\, e^{-ikz_q}$ can be rewritten as $\lambda z_q\, e^{-i(kz_q+\frac{\pi}{2})}$.
  The exponent is merely a constant phase change, which will only alter the output phase of all the pixels by a constant amount, therefore can be ignored.
- The quadratic phase factor outside of the FT is in the structure very similar to the third Zernike polynomial:

  $$e^{-i\frac{\pi}{\lambda z_q}\left(x^2+y^2\right)} \propto e^{-i\frac{\alpha}{z_q} Z_3(x,y)}$$

  where α is a scaling constant, depending on the experimental setup (size and number of SLM pixels and the wavelength).
- The object field $\psi(u,v,z)$ is a complex variable. A phase of this field, usually assigned uniformly distributed random variable, modulated by a quadratic factor $\exp\{\frac{\pi}{\lambda z_q}\left(u^2+v^2\right)\}$ is yet another random variable. Hence, for all practical purposes this factor can safely be ignored.

Once all of these approximations are applied, one arrives at a simplified formula:

$$H_q(x,y) = \lambda z_q\, e^{-i\frac{\alpha}{z_q} Z_3(x,y)}\, \mathscr{F}^{-1}\left\{ \psi(u,v,z_q) \right\}$$

This is the complex field, due to a single layer of the 3D object. In order to calculate the field coming from all the layers, corresponding holograms are summed:

$$H(x,y) = \lambda \sum_{q=0}^{layerCount} z_q\, e^{-i\frac{\alpha}{z_q} Z_3(x,y)}\, \mathscr{F}^{-1}\left\{ \psi(u,v,z_q) \right\}$$

[1] Vamsee Bheemireddy helped to refine the calculation

One can then assign $\varphi_q(x,y) = -\frac{\alpha}{z_q} Z_3(u,v)$ which transforms the above equation to:

$$H(x,y) = \lambda \sum_{q=0}^{layerCount} z_q \, e^{i\varphi_q(x,y)} \, \mathscr{F}^{-1}\left\{\psi(u,v,z_q)\right\} \tag{7.1}$$

By this point, it should be clear that Eq. 7.1 and Eq. 4.2 are, up to a scaling constant, equivalent. The aberration-correcting phase masks are now quadratic phase factors (third Zernike polynomial with appropriate scalings) and the PC-OSPR regions became the Fresnel slices.

The CUDA implementation of PC-OSPR can be seamlessly ported to generate 3D holograms. The execution time is, however, increased significantly, because instead of 6 aberration-correcting regions, the 3D object now consists of many more layers (128 or 256).

A full algorithm can be summarized below:

- Generate/acquire the point cloud
- Create a set of layers, by grouping the points that have the same Z-values
- Perform the FT on each layer and apply appropriate phase correction
- Sum corresponding holograms
- Repeat the procedure with a different input random phase

When 24 random phase holograms are calculated, the computation time is only 4 seconds on the TITAN X GPU. In order to reduce that time further, one can use smart quantization techniques. One of these techniques, termed OSPR with pseudo-random phase was constructed, but cannot be included here, because of the confidentiality agreement.

7.3.2 Point cloud generation

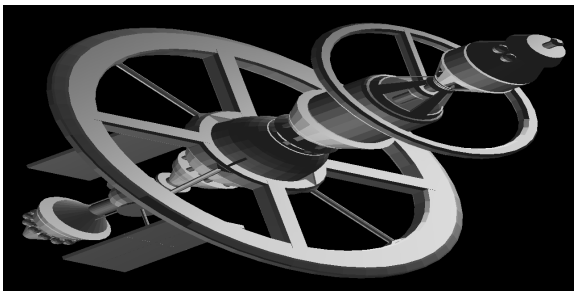
The above method requires a point cloud as the input. How to generate such point cloud depends on the application. Some applications will be discussed in the following section. For the purpose of algorithm performance evaluation, a computer-generated point cloud is used.

OpenGL-based point cloud renderer

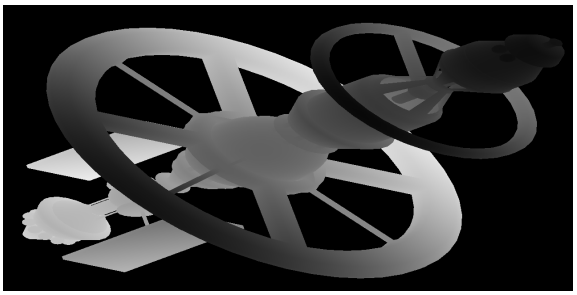
The point cloud generation module was implemented in C++ using OpenGL graphics engine [78] [2]. The module displays the 3D model given in the *.off file format [99]. The user can move, scale and rotate this object in order to place it in the 3-dimensional environment using the keyboard input. The model is then rendered with some predefined lightening and gets the occlusion calculated. The model is rendered to a framebuffer [100], where the colour as well

[2] Parts of the code were adapted from a previous implementation made by Rick Chen [98]

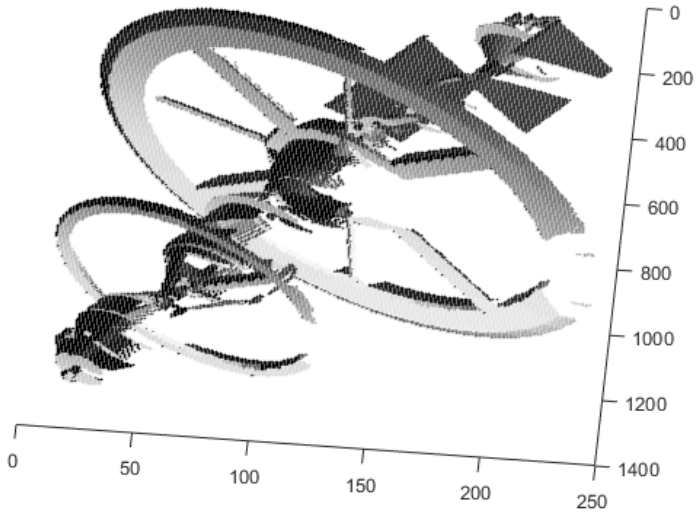
as depth information is captured (seen in Figs. 7.3a and 7.3b respectively). The point cloud is constructed by getting the X and Y coordinates from the position on the grid and the Z coordinate from the depth texture. The output 3D point cloud aligned so that the calculated occlusion is visible can be seen in Fig. 7.3c.



(a) Object Color



(b) Object depth



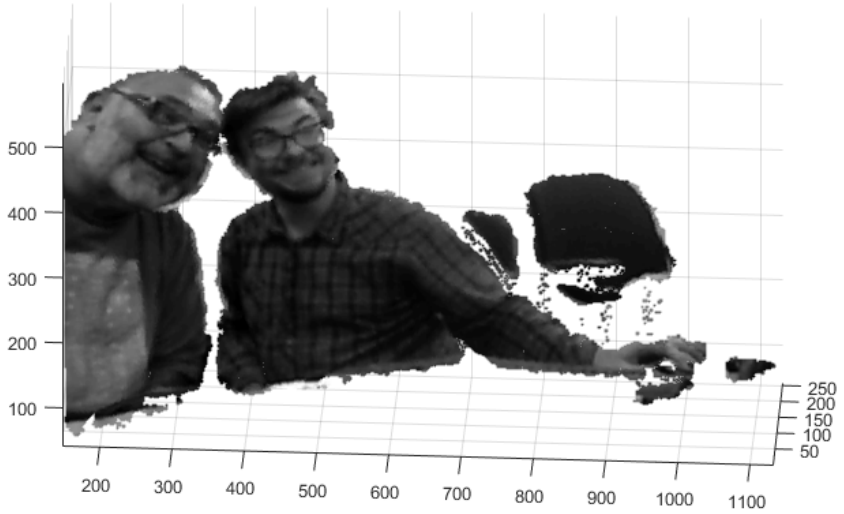
(c) 3D point cloud

Fig. 7.3 Production of the point cloud

XBox Kinect point cloud acquisition

Another way of constructing a point cloud is by using a 3D sensor to scan the environment. The primary proof-of-principle work was done using XBox Kinect v1[3]. It should be emphasized that, because Xbox Kinect is an old piece of hardware, the quality of the acquired

[3]Work on the Xbox Kinect was carried out by Mikolaj Kosinski and Shengjun Ren



(a) A sample PointCloud acquired from XBox Kinect

(b) Reconstruction of a hologram, focus set on the foreground

(c) Reconstruction of a hologram, focus set on the background

Fig. 7.4 Hologram generation from XBox Kinect

3D image is relatively low, outputting the image 640x480 pixels in size. Whenever a more advanced technology is used, the quality can be improved accordingly.

The XBox Kinect utilizes two cameras to construct a map of 3D environment. One camera acquires the colour and the other, the depth information. Depth is obtained by scanning the environment with an infrared laser and then, filtering only that wavelength with a sensor. Because the two cameras are offset with respect to each other, two images will have slightly different parallax. The mapping between the colour image and the depth image is done automatically using DirectX in a sample program provided by Microsoft.

7.3.3 Holography-over-IP

Once a 3-dimensional image is acquired, it can be converted to a point-cloud and passed as an input to our hologram generation algorithm. This way, one can then construct a live 3D holographic video stream. The elements of such system are presented in the schematic in Fig. 7.5. After the acquisition, the image is rendered in a computer program, where the necessary processing takes place. The ready point cloud is then passed into the high-performance GPU hologram generation module. The data is sent over a standard IP link to the client computer, where it's decoded and displayed on a holographic display.

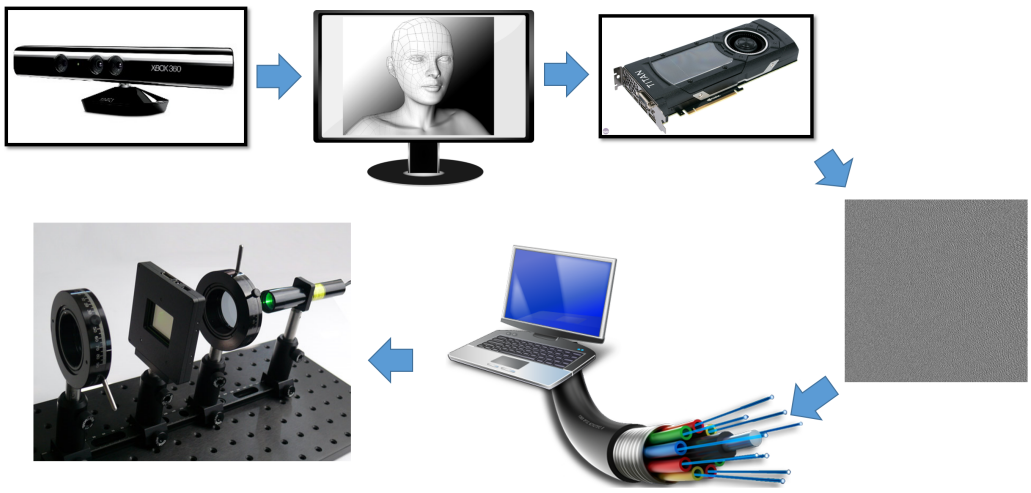
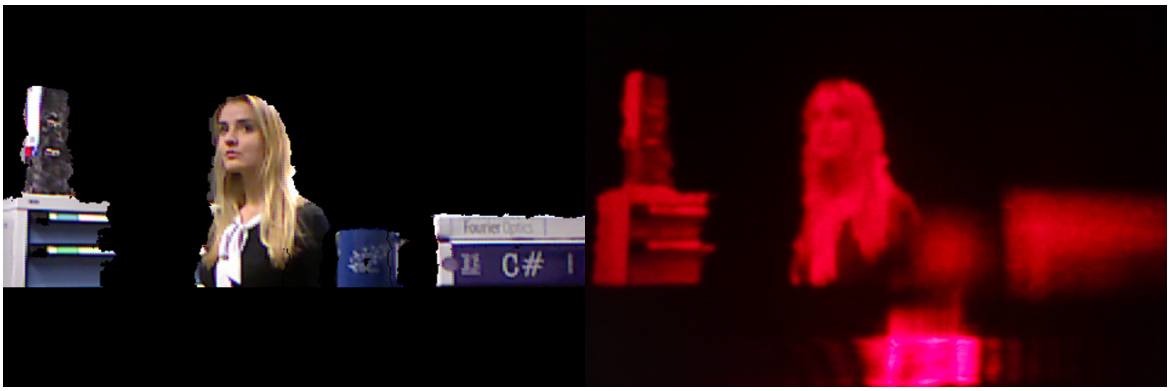
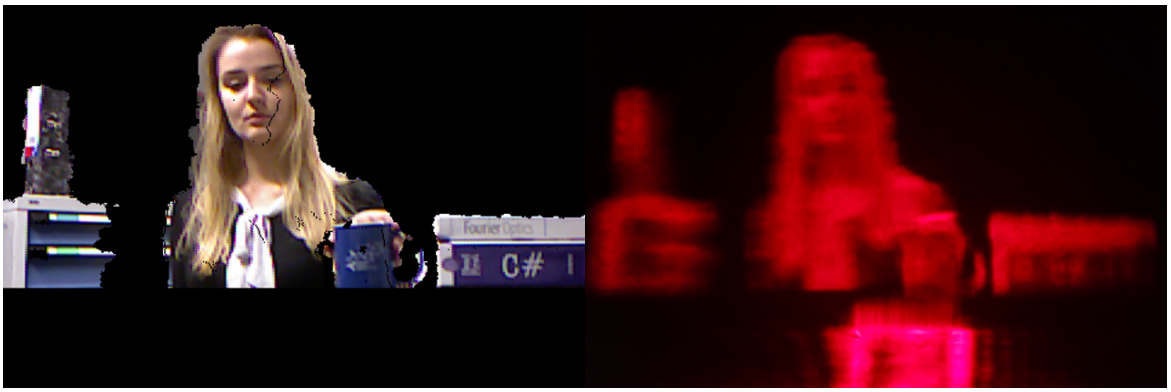


Fig. 7.5 The concept of Holography-over-IP

The prototype was ran on nVidia GeForce GTX TITAN X. In order to reach real-time speeds, the number of layers has been reduced to 64. The previously mentioned OSPR with pseudo-random phase method provided a continuous hologram generation at a frame-rate of 8fps. Two frames from this transmission are presented in Fig. 7.6. The input from the Kinect, which has a filtered out background is put side by side with the output from the camera. The images are slightly blurry, because of a rolling shutter phenomenon of the Canon EOS 6D. The lady's movements were being followed by the focus of the camera. In Fig. 7.6a the camera focuses on the far plane (the shelves are in focus), while in Fig. 7.6b, the books and the cup are in focus.



(a) Movie frame 306



(b) Movie frame 404

Fig. 7.6 Proof of principle Holo-over-IP real-time transmission[4]

7.4 Future work - Unfinished and postulated research projects

This section contains the list of projects that weren't completed during the course of this research. However, they show enough potential to be carried forward.

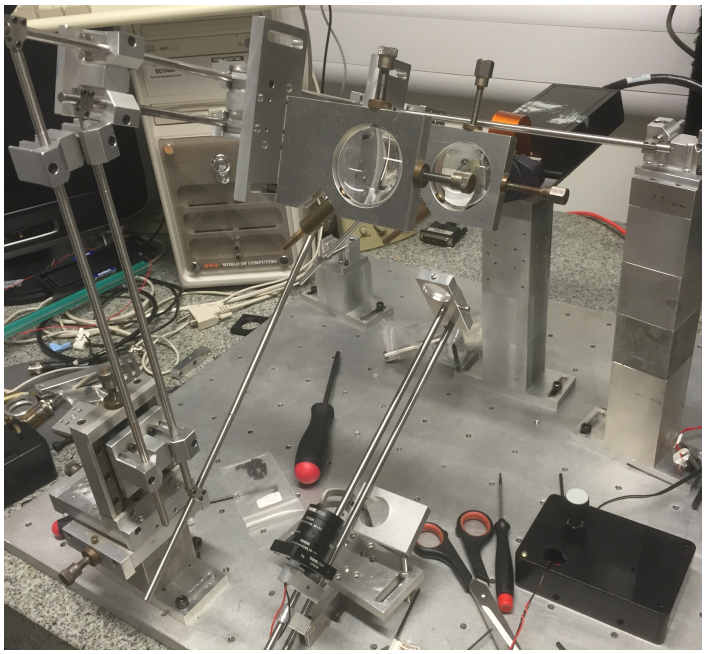
7.4.1 3D Aberration Correction

The aberration correction of 3D directly-viewed holograms was supposed to be a primary focus of this thesis. It was decided that it is more important to perfect the 2D aberration correction before attempting to solve the 3D case. Indeed, the idea of 3D aberration correction shows potential and should be studied in greater detail.

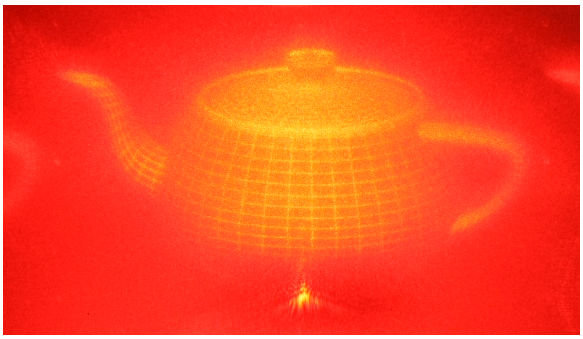
[4] Aleksandra Pe Pedraszewska kindly agreed to impersonate a busy businesswoman while multiple people assisted with the movie recording: Tom Durrant, Roman Pechhacker

The problem, however, is much more fundamental. A directly-viewed 3D object, when seen through a telescope, changes when viewed from different angles. Suddenly, instead of four degrees of freedom (two spatial coordinates and two hologram coordinates), we have seven of them (two spatial, two hologram, two angular and the object depth). All of these degrees of freedom affect the viewed replay field in a yet unknown fashion.

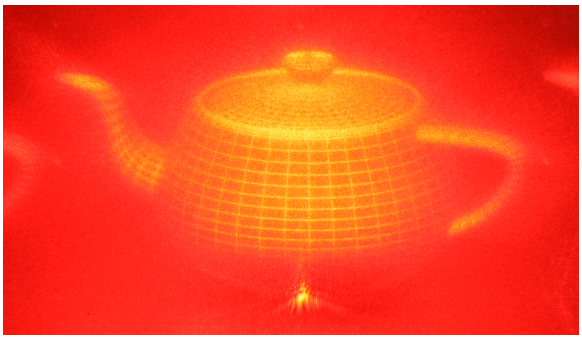
The proof of principle setup can be seen in Fig. 7.7a. Given a particular viewing angle and depth, the current setup can be successfully used to correct the aberrations, as seen in Figs. 7.7b - 7.7c.



(a) 3D viewing setup employing a telescope



(b) Uncorrected image



(c) Corrected image

Fig. 7.7 Spatial variation of aberrations

There are two separate ways in which this problem can be approached. A purely experimental perspective would correct the image in a number of 3D regions, similar to PC-OSPR. Another will involve a thorough study of the optical system using ZEMAX.

7.4.2 Ultra-realistic hologram generation using a ray-tracing engine

A large number of researchers in the field of 3D holography tend to focus on proof-of-principle generation of simple test objects, like teapots, Stamford bunnies, dragons, pyramids. The construction of these objects is indeed some indicator of the image quality, but not of real-life utility of such algorithms.

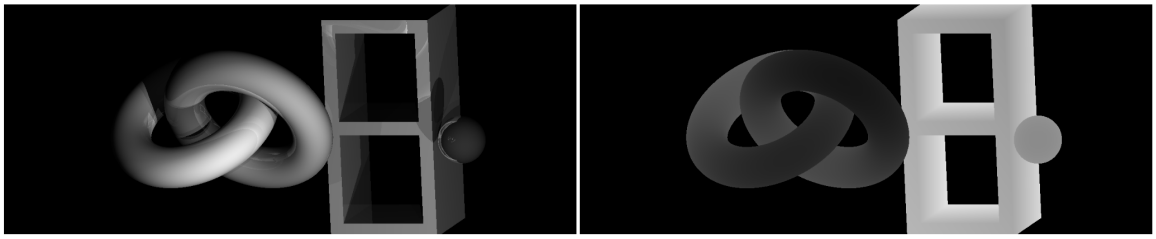
On the other hand, the field of computer graphics has developed an ability to generate ultra-realistic scenes, including most of the effects encountered in real life (lightening, shading, reflections, etc.). Majority of holography researchers, even while attempting to incorporate these advanced effects, tend to reverse-engineer computer graphics to suit the framework of hologram generation, which results in high computational loads.

There exist a number of issues with the generation of 3D holograms. These include the difficulty in calculating the correct occlusion and the computational complexity of the hologram generation algorithms. The first problem was partially handled by Rick Chen [98]. However, a closer examination of his approach and the holograms generated reveals that his method does not provide enough 3-dimensional cues. His holograms are therefore merely 2D far-field images with introduced viewer-dependent parallax. Therefore, an attempt has been made to port Chen's method to Fresnel hologram calculation.

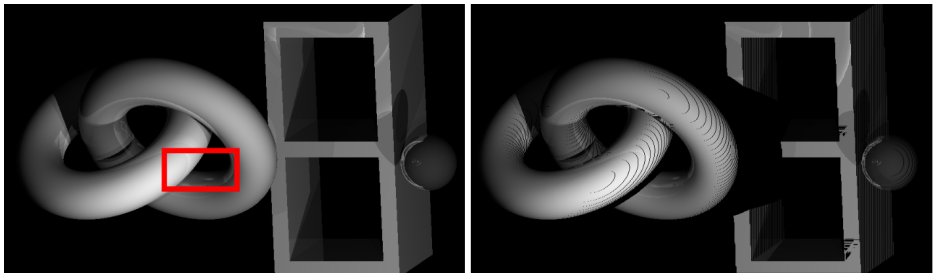
The proof-of-principle work has been carried out using a basic C++ ray-tracer, with an intention to use one of the state-of-the-art commercial solutions.

A simple set of 3 objects was generated, as seen in Fig. 7.8a. The simulation of viewing from different angles can be seen in Fig. 7.8b. When enlarged, it can be shown that the reflections of the objects, change with the viewpoint (as seen in Fig. 7.8c)

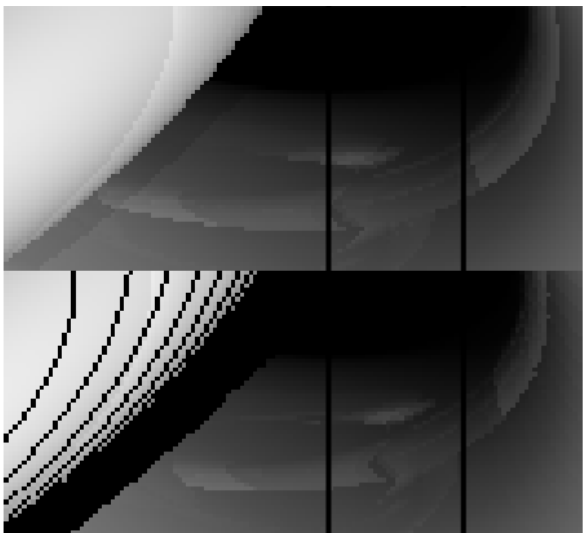
The ultimate goal of this project would be to construct a framework for the future generation of computer games, where current 3D games can be ported to the holographic domain with minimal code modifications.



(a) Output of a ray-tracer: (left) Intensity (right) depth



(b) Simulation of viewer-dependent effects: (left) viewer looking head-on (right) viewer looking from the side



(c) Viewer-dependent reflections

Fig. 7.8 Viewer-dependent visual effects

References

[1] D. Gabor. Holography, 1948-1971. *Science*, 177(4046):299–313, 1972.

[2] D. Gabor. Microscopy by reconstructed wavefronts. 1948.

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

[4] D. Gabor. Microscopy by reconstructed wave-fronts. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 197(1051):454–487, 1949.

[5] F. Zernike. How i discovered phase contrast. *Science*, 121(3141):345–349, 1955.

[6] M. Wolfke. Über die möglichkeit der optischen abbildung von molekulargittern. *Phys. Zeits.*, 21:495–497, September 1920.

[7] Historia Wydziału Fizyki PW - Rozdział 4 Działalność Profesora Mieczysława Wolfke, February 2015.

[8] A. K. Wroblewski. Polish physicists and the progress in physics (1870-1920). *Technical Transacions, Fundamental Sciences*, 2014.

[9] M. P. Sadowski. Hologramy i holografia, May 2005.

[10] E.N. Leith. 1 - introduction. In H.J. CAULFIELD, editor, *Handbook of Optical Holography*, pages 1 – 12. Academic Press, San Diego, 1979.

[11] Y. N. Denisyuk. Photographic Reconstruction of the Optical Properties of an Object in Its Own Scattered Radiation Field. *Soviet Physics Doklady*, 7:543, December 1962.

[12] Emmett N. Leith and Juris Upatnieks. Reconstructed wavefronts and communication theory∗. *J. Opt. Soc. Am.*, 52(10):1123–1130, Oct 1962.

[13] SA BENTON. HOLOGRAM RECONSTRUCTIONS WITH EXTENDED INCOHERENT SOURCES. *JOURNAL OF THE OPTICAL SOCIETY OF AMERICA*, 59(11):1545–&, 1969.

[14] H. Lopata. Fraud resistant credit card system, February 3 1987. US Patent 4,641,017.

[15] Holocenter.org. http://holocenter.org/what-is-holography/?gclid=CIaz0siT-dACFcIV0wodEtkPjA. Accessed: 2016-09-21.

[16] Joseph W. Goodman. *Introduction to Fourier Optics, Third Edition*. Roberts & Company Publishers, Greenwood Village, 2005.

[17] K. Izuka. *Elements of Photonics, Volume I: In Free Space and Special Media*. John Wiley & Sons, Inc., New York, 2002.

[18] P. Hariharan. *Basics of Holography*. Cambridge University Press, 2002. Cambridge Books Online.

[19] T. D. Wilkinson. 4b11 photonic systems course. Lecture Notes, Cambridge University, Engineering Department, 2011.

[20] D. R. McAdams. *Dynamic Maskless Holographic Litography and Applications*. PhD thesis, University of Pittsburgh, Swanson School of Engineering, 2012.

[21] James B Wendt. Computer generated holography. Master's thesis, Department of Physics, Pomona College, Claremont, 2009.

[22] Ronaldo D. Mansano Jose C. Pizolato Jr. Daniel B. Mazulquim Giuseppe A. Cirino, Patrick Verdonck and Luiz G. Neto. Digital holography: Computer-generated holograms and diffractive optics in scalar diffraction domain. In Dr. Freddy Monroy, editor, *Holography - Different Fields of Application*. InTech, 2011.

[23] A. Lohmann. Optische einseitenbandübertragung angewandt auf das gabor-mikroskop. *Optica Acta: International Journal of Optics*, 3(2):97–99, 1956.

[24] B. R. Brown and A. W. Lohmann. Complex spatial filtering with binary masks. *Appl. Opt.*, 5(6):967–969, Jun 1966.

[25] A. W. Lohmann and D. P. Paris. Binary fraunhofer holograms, generated by computer. *Appl. Opt.*, 6(10):1739–1748, Oct 1967.

[26] L. B. Lesem, P. M. Hirsch, and J. A. Jordan. The kinoform: A new wavefront reconstruction device. *IBM Journal of Research and Development*, 13(2):150–155, March 1969.

[27] C. Slinger, C. Cameron, and M. Stanley. Computer-generated holography as a generic display technology. *Computer*, 38(8):46–53, Aug 2005.

[28] V.M. Bove Jr, Q.Y.J. Smithwick, J. Barabas, and D.E. Smalley. Is 3-D TV preparing the way for holographic TV? 2005.

[29] Edward Buckley. *Computer-Generated Phase-Only Holograms for Real-Time Image Display in Advanced Holography - Metrology and Imaging*. InTech, London, 2011.

[30] Mark E. Lucente. Interactive computation of holograms using a look-up table. *Journal of Electronic Imaging*, 2(1):28–34, 1993.

[31] Yuan-Zhi Liu, Jian-Wen Dong, Yi-Ying Pu, He-Xiang He, Bing-Chu Chen, He-Zhou Wang, Huadong Zheng, and Yingjie Yu. Fraunhofer computer-generated hologram for diffused 3d scene in fresnel region. *Opt. Lett.*, 36(11):2128–2130, Jun 2011.

[32] Hiroshi Yoshikawa. Fast computation of fresnel holograms employing difference. *Optical Review*, 8(5):331–335, 2001.

[33] Joel S. Kollin, Stephen A. Benton, and Mary L. Jepsen. Real-time display of 3-d computed holograms by scanning the image of an acousto-optic modulator, 1989.

[34] Adrian Cable. Pico projectors: Interactive experience. *Nature Photonics*, 4(11):750–751, 2010.

[35] Y. Montelongo. Computer-generated holography, progress report. Master's thesis, Cambridge University, Department of Engineering, 2009.

[36] 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(1):1431–1433, 2004.

[37] Jason Geng. Three-dimensional display technologies. *Adv. Opt. Photon.*, 5(4):456–535, Dec 2013.

[38] Nathan J. Jenness. *Three-dimensional Holographic Lithography and Manipulation Using a Spatial Light Modulation*. PhD thesis, Duke University, Department of Mechanical Engineering and Materials Science, 2009.

[39] J. L. de Bougrenet de la Tocnaye and L. Dupont. Complex amplitude modulation by use of liquid-crystal spatial light modulators. *Appl. Opt.*, 36(8):1730–1741, Mar 1997.

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

[41] J. R. Fienup. Phase retrieval algorithms: a comparison. *Appl. Opt.*, 21(15):2758–2769, Aug 1982.

[42] R. W. Gerchberg and W. O. Saxton. A practical algorithm for the determination of the phase from image and diffraction plane pictures. *Optik (Jena)*, 35:237+, 1972.

[43] Olivier Ripoll, Ville Kettunen, and Hans Peter Herzig. Review of iterative fourier-transform algorithms for beam shaping applications. *Optical Engineering*, 43(11):2549–2556, 2004.

[44] E. Buckley. *Computer Generated Holograms for Real-Time Image Display and Sensor Applications*. PhD thesis, Cambridge University, Department of Engineering, 2006.

[45] James C Wyant and Katherine Creath. Basic wavefront aberration theory for optical metrology. *Applied optics and optical engineering*, 11(29):2, 1992.

[46] H. Gross. *Handbook of Optical Systems*. Wiley, New York, 2005.

[47] Lens correction model. PanoTools Wiki, 2000. [online] http://wiki.panotools.org/Lens_correction_model.

[48] Alex Chtchetinine. Radial distortion in low-cost lenses: numerical study. *Optical Engineering*, 47(2):023001–023001–7, 2008.

[49] von F. Zernike. Beugungstheorie des schneidenver-fahrens und seiner verbesserten form, der phasenkontrastmethode. *Physica*, 1(7):689 – 704, 1934.

[50] PhD Jum Schwiegerling. Ocular wavefront error representation (ansi standard).

[51] Robert J. Noll. Zernike polynomials and atmospheric turbulence∗. *J. Opt. Soc. Am.*, 66(3):207–211, Mar 1976.

[52] Ophthalmics - methods for reporting optical aberrations of eyes, ansi z80.28-2004, 2004.

[53] PhD Charlie Campbell. Ansi standards for reporting wavefront error. 6th International Congress on Wavefront Sensing and Optimized Refractive Correction, Feb 2005. Short Course in Ophthalmic Wavefront Sensing.

[54] *ZEMAX, Optical Design Program, User's Manual*. Jul 8, 2011.

[55] D. Malacara. *Optical Shop Testing*. Wiley Series in Pure and Applied Optics. Wiley, 2007.

[56] J. P. Freeman. *Visor Projected Helmet Mounted Display for Fast Jet Aviators using a Fourier Video Projector*. PhD thesis, Cambridge University, Department of Engineering, 2009.

[57] Jonathan P. Freeman, Timothy D. Wilkinson, and Paul Wisely. Visor projected hmd for fast jets using a holographic video projector, 2010.

[58] R. Cicala. There is no such thing as a perfect lens. PetaPixel blog, 2013. [online] http://petapixel.com/2013/09/14/perfect-lens/.

[59] Achmed Bouazzam, Torsten Erbe, Stephan Fahr, and Jan Werschnik. Lens-mount stability trade-off: a survey exemplified for duv wafer inspection objectives, 2015.

[60] Paul Hickson. *Fundamentals of Atmospheric and Adaptive Optics*. The University of British Columbia, 2008.

[61] C. A. Primmerman D. P. Greenwood. Adaptive optics research at lincoln laboratory. *The Lincoln Laboratory Journal*, 5(1), 1992.

[62] Justin D. Mansell. *Introduction to Adaptive Optics*. Active Optical Systems, LLC, 2011.

[63] Example wavefronts and images, 2003. [online] http://www.lyot.org/background/adaptive_optics.html.

[64] Ping Yang, Yuan Liu, Mingwu Ao, Shijie Hu, and Bing Xu. A wavefront sensor-less adaptive optical system for a solid-state laser. *Optics and Lasers in Engineering*, 46(7):517 – 521, 2008.

[65] A. Kaczorowski. Digital aberration correction for wide-angle holographic projectors. Master's thesis, Cambridge University, Department of Engineering, 2012.

[66] B.E.A. Saleh and M.C. Teich. *Fundamentals of photonics*. Wiley series in pure and applied optics. Wiley, 1991.

[67] Andrzej Kaczorowski, George S. Gordon, Ananta Palani, Stanisław Czerniawski, and Timothy D. Wilkinson. Optimization-based adaptive optical correction for holographic projectors. *J. Display Technol.*, 11(7):596–603, Jul 2015.

[68] J. A. Snyman. *Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms*, volume 97. Springer US, New York, 2005.

[69] J. Carpenter and T. D. Wilkinson. Aberration correction in spatial light modulator based mode multiplexers. In *Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference (OFC/NFOEC), 2013*, pages 1–3, March 2013.

[70] Agoston E Eiben and Marc Schoenauer. Evolutionary computing. *Information Processing Letters*, 82(1):1–6, 2002.

[71] S. D. Carpenter, P. M. Weber, J. Peter, G. Szabo, T. Szakacs, and A. Lorincz. Self-learning optical system based on a genetic-algorithm driven spatial light modulator. In P. di Lazzaro, editor, *Second GR-I International Conference on New Laser Technologies and Applications*, volume 3423 of , pages 130–134, July 1998.

[72] Brad L Miller and David E Goldberg. Genetic algorithms, tournament selection, and the effects of noise. *Complex systems*, 9(3):193–212, 1995.

[73] Digital image developer program, eos digital software development kit, 2010. [online] https://www.didp.canon-europa.com/.

[74] Chandra S. Vikram. Rayleigh versus marechal spherical aberration tolerance in in-line fraunhofer holography. *Optical Engineering*, 33(11):3715–3717, 1994.

[75] A. Kaczorowski. Digital aberration correction for wide-angle holographic projectors, progress report, 2013.

[76] Yunuen Montelongo, Ananta Palani, and Tim Wilkinson. Simulations of time multiplexed fraunhofer holograms produced by binary phase slms for video projection. In *Latin America Optics and Photonics Conference*, page LM2A.9. Optical Society of America, 2012.

[77] Kandrot Jason Sanders, Edward. *CUDA by Example: An Introduction to General-Purpose GPU Programming*. Addison-Wesley Professional; 1st Ed., 2010.

[78] The open graphics library (OpenGL) - the industry's foundation for high performance graphics. https://www.opengl.org/about/. Accessed: 2016-09-29.

[79] The OpenGL utility toolkit (glut) library. https://www.opengl.org/resources/libraries/glut/. Accessed: 2016-09-21.

[80] GLFW library. http://www.glfw.org/. Accessed: 2016-09-21.

[81] 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.

[82] Teamviewer remote control software, 2010. [online] http://www.teamviewer.com/en/company//.

[83] Canon EDSDK tutorial in C#, 2013. Accessed: 2013-12-18.

[84] Xieliu Yang, Suping Fang, and Yulin Yang. Accurate template-based correction technology for lens distortion. *Optical Engineering*, 51(10):103602–1–103602–8, 2012.

[85] Dave Coffin. DCraw: Decoding raw digital photos in Linux, 2013.

[86] DCraw for Windows, 2013.

[87] T. Elworthy, D. Bilson, N. Homan. Optical printers, 2010. Patent No.: US 2010/0014064 A1.

[88] Discontinued devices, 2013. Accessed: 2016-10-25.

[89] C. Bay. *Dynamic holographic masks for adaptive optical lithography*. PhD thesis, University of Cambridge, Department of Engineering, 2011.

[90] G. Shabtay. Three-dimensional beam forming and Ewald's surfaces. *Optics Communications*, 226:33–37, October 2003.

[91] IDEX: Optics and Photonics Marketplace. Gaussian Beam Optics, 2015. [online] https://marketplace.idexop.com/store/SupportDocuments/All_About_Gaussian_Beam_OpticsWEB.pdf.

[92] Laser Quantum. gem: High specification OEM CW lasers, 2015. [online] https://www.didp.canon-europa.com/.

[93] Laser Quantum. gem: 532nm high spec OEM laser, Technical data sheet, 2015. [online] https://www.didp.canon-europa.com/.

[94] Constantine A. Balanis. *Antenna Theory: Analysis and Design*. Wiley-Interscience, 2005.

[95] Thorlabs complete zemax catalog. [online] https://www.thorlabs.com/software_pages/ViewSoftwarePage.cfm?Code=Zemax.

[96] Edmund optics zemax catalog. [online] http://www.edmundoptics.co.uk/products/zemax-catalog/index.cfm.

[97] P. W. J. Hands. Digital holographic photonic printing. Final Report, Proof of Principle Project, October 2014 to June 2015, 2015.

[98] R. H-Y. Chen. *Computer-Generated Holograms for 3-D Holographic Display*. PhD thesis, Cambridge University, Department of Engineering, 2010.

[99] Object file format (.off). http://shape.cs.princeton.edu/benchmark/documentation/off_format.html. Accessed: 2016-09-29.

[100] Opengl framebuffers, 2015. Accessed: 2015-02-10.

Appendix A

Feedback loop: A highly-parallel, error-resistant implementation

A number of further speed improvements were made, based on the following observations:

- Every task executes with some overhead, that is independent of the task's duration. For instance, the overhead to prepare the GPU for hologram production is independent of the number of holograms produced [32, 75].
- The picture-acquisition module is known to break down for a multitude of reasons (occasional webcam failure, unresponsive DLL module, etc.). Running it for longer periods of time (several minutes), makes these events more likely to happen.
- It was noticed that, as large quantities of holograms were produced (even by multiple kernels), the file IO slows down rapidly. Closer investigation of the problem revealed the bottleneck in the low-level Windows API. This could not be avoided even when a fast SSD drive was used.
- The number of holograms produced is limited by the GPU memory to about 340 for GeForce GTX 760 used here.

A.1 Hologram generation kernel

Since the hologram generation is the most computationally expensive and easy to parallelize, this task was ported from initial MatLAB implementation to highly-parallel native CUDA C implementation [75]. For simplicity, one thread on the GPU has been assigned to one hologram pixel. Each particular thread keeps the information specific to its location, such as values of Zernike polynomials and the continuous phase surface. As the input coefficients have to be read from global memory by every thread, shared memory usage was introduced

for further speedup. Each thread in a block reads a portion of Zernike coefficients and puts it into the CUDA on-chip shared memory. During the execution, threads only refer to the shared memory, hence optimizing the read-time. The kernel supports few modes of hologram generation:

- Generating permutations given two parents
- Producing purely-random candidates
- Performing heuristic descent optimization

Every operation supports multiple inputs that are processed one after another to minimize the previously discussed overhead.

In order to keep track of the particular hologram's Zernike Coefficients, they are included in the file name separated by semicolons, for instance:

```
hol0012_Pt0.000;0.000;0.095;0.329;-0.017;0.854;-2.311;-1.089; (...)
0.000;0.525;0.688;-0.119;0.041;-1.010;-0.367.BMP
```

All hologram files start by default with a "hol" prefix. A number of useful parameters is also encoded in each filename:

- Iteration number: 0012
- Prefix "Pt" indicating that it is a single point (holograms of different shapes differ by prefixes, for instance Cr indicates a cross, Sq - a square, etc.)
- A set of Zernike coefficients: $a_1 = a_2 = 0.000, a_3 = 0.095, a_4 = 0.329, \ldots$

Following this naming convention made it easy to keep track of millions of generated holograms.

A.2 Picture acquisition module

After the holograms are calculated, feedback from the camera needs to be acquired. For high throughput as well as simplicity of interfacing the peripherals (webcam and the camera), C# was selected. The language proved much more user-friendly than C++ while the Visual Studio form environment made writing Windows applications as simple as a click of a mouse. The communication with MatLAB is achieved by command-line arguments passing. The input and output catalogues are passed this way as well as the number of the screen, and picture delay (in milliseconds).

Provided the input and output catalogues, the program loads all of the .BMP files from the input catalogue to the list. Each picture from the list is first displayed on the secondary monitor, has its picture taken using either a webcam, or a digital Single Lens Reflex (dSLR) in a LiveView mode [73]. The picture is saved with a "pic_" prefix, so that the information

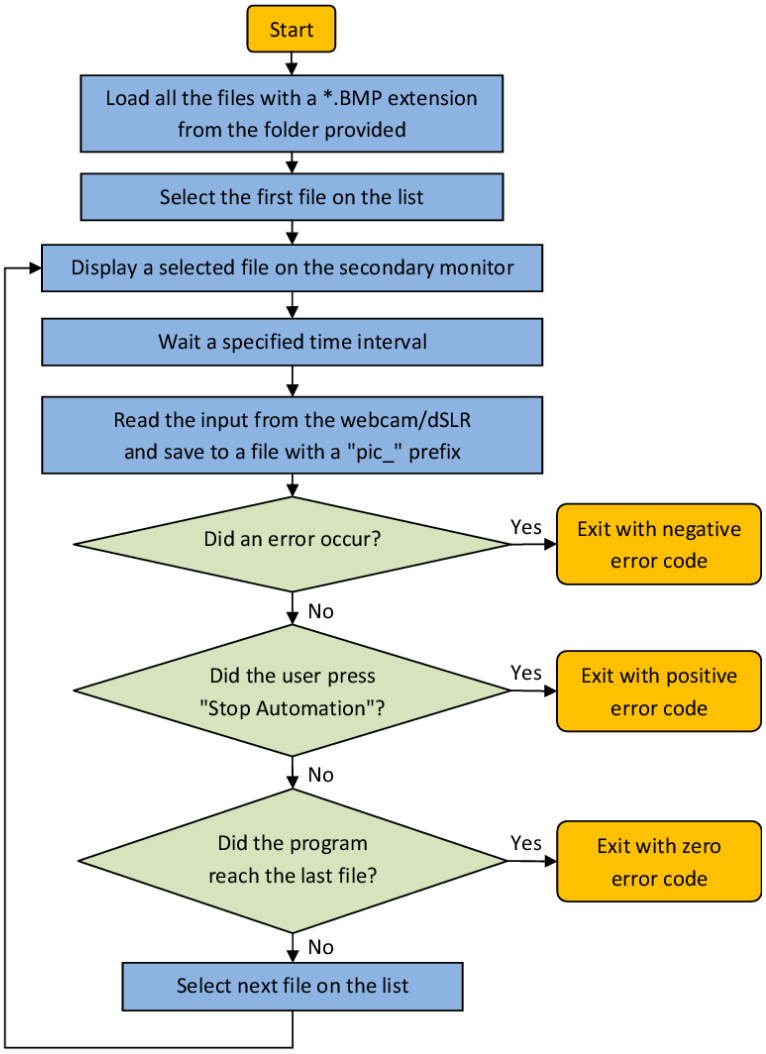


Fig. A.1 A flowchart of the picture acquisition module's operation

about the correction coefficients from the original hologram is preserved. Additionally, a high-level emergency timer is introduced. If the program does not exit within a specified time interval, it is likely to be a malfunction. If this is the case, the application exits with an appropriate error code.

The picture taking module communicates with MatLAB via error exit codes. If the program completes without errors, the returned value is 0. If the user clicks a "Stop Automation" button, the error code passed is positive, in which case the feedback loop will instantly be terminated. Any other error is represented by negative error codes, indicating that the execution of the program should be repeated. An appropriate error number corresponds to a specific error that occurred. For instance a value of -50 indicates that the execution was terminated by a global emergency timer.

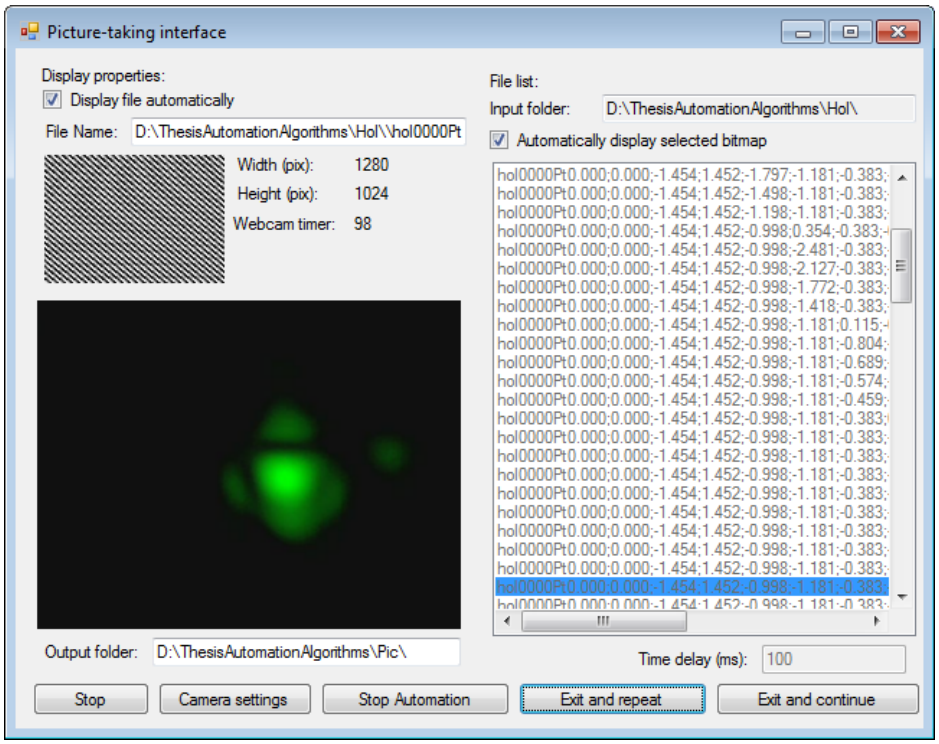


Fig. A.2 Screenshot of a picture acquisition module in operation

A.3 Main feedback loop control script

All of the modules described here need to be coordinated to achieve precise and fast correction. In the first iteration of the system, that main element was the C# program, calling MatLAB in the background. But because of its occasional break-downs, the feedback loop became unresponsive and was unable to continue. It was therefore decided that MatLAB, because of its stability, will become the main element in which the control script would be written. Even if a C# application breaks down and exits, it will pass an appropriate error code and its operation can automatically be repeated without loosing hours of experimental time.

To ensure maximum throughput together with error-resistance, the following rules were applied:

- In a single hologram generation batch, at most 1200 holograms are generated at once to avoid an IO bottleneck
- Long kernel executions are followed by picture acquisition. The number of pictures should not be greater than 2000, otherwise the breakdown of C# module becomes increasingly likely

- After the picture analysis, all of the results together with the state of the program are written to a log file in such a way that the mechanism can be restarted from any given point. In the rare case of a severe error, the operation can be fully resumed.