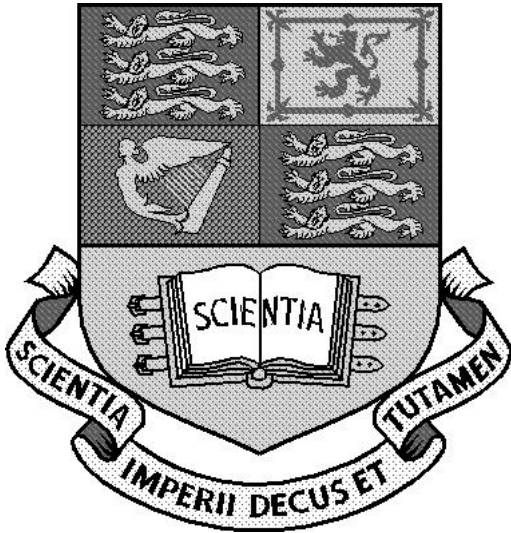


IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY
AND MEDICINE

UNIVERSITY OF LONDON



**Direct-search method for the computer design of
holograms.**

Matthew Clark

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Applied Optics Group,
Department of Physics

May 1997

Abstract

A direct-search method for the computer design of phase holograms is presented. This method optimises the hologram phase distribution in order to produce a specified image intensity distribution and tailors the hologram phase distribution to the illuminating wavefront and the hologram aperture.

It is a general method and does not rely on a specific optical geometry. The optimisation of the hologram phase distribution is carried out by examining the effect of small changes made to the hologram phase distribution on the resulting image. These changes are then kept or discarded depending on any improvement of the image. The effect of the changes on the resulting image are calculated using an approximation to the Fresnel-Kirchhoff integral.

A survey of the existing work and methods together with a discussion of hologram fabrication techniques relevant to computer generated holography is presented.

The direct-search method relies on an underlying mathematical model to describe the hologram and calculate the image complex amplitude distribution. This model is critical to the success of the direct-search method. The requirements of this model are investigated and presented.

A method capable of analysing the images produced by such holograms is presented and discussed.

Computational problems and restrictions associated with this technique are discussed.

This direct-search method has many operational parameters which affect the results of optimisations and the computational efficiency of the method itself. The direct-search method is a computationally intensive technique and therefore its computational efficiency is important. Parameters affecting the computational efficiency of the algorithm are discussed and investigated. Parameters affecting quality and reliability of the results this method produces are presented and discussed.

Novel computer generated holograms designed by this technique are presented and analysed.

Acknowledgements

I thank Robin Smith, my supervisor, for giving me the freedom to do this research and for having faith in the right places. I am also grateful to Carl Paterson, a good friend and colleague who's been really useful. Special mention to Irene Turner for her supreme competence. Thanks to the Applied Optics group.

Love and respect to Mez and Paul for putting up with me, to Thalia McElwee who showed me the light at the end of the tunnel and to Julia and Jackie who brought a glamour and a touch of class into my life.

Thanks to Jan, Pete and Bob for their long standing support and plentiful advice throughout and to Justin for (amongst other things) push yer luck.

Extra Special mention for Laura, she knows why.

Contents

List of Figures	5
1 Introduction	8
1.1 Diffractive optical elements (DOEs) and CGHs	8
1.2 Computer generated holography	9
1.3 Applications of CGHs	9
1.3.1 Laser machining	9
1.3.2 Optical interconnects	10
1.3.3 Laser ultrasound generation	10
1.4 Introduction to the direct-search method	11
2 Survey of existing work and methods	13
2.1 Optically generated holograms	13
2.2 Analytical methods	14
2.2.1 Direct complex amplitude calculation	14
2.2.2 Geometrical transform holograms	15
2.3 Optimisation methods	16
2.3.1 Projection-on-constraints	16
2.3.2 Genetic algorithms	18
2.3.3 Simulated-annealing	20
2.3.4 Direct-search algorithms	22
3 Hologram fabrication	24
3.0.5 Holographic recording materials	24
3.1 Overview of possible hologram fabrication techniques	24
3.1.1 Direct write by laser plotter	24
3.1.2 Direct write via CRT	25
3.1.3 Etched metallic substrates	25
3.1.4 Spatial light modulators	25
3.1.5 Direct machining of substrate surface	26
3.2 Electron-beam micro-lithography techniques	26
3.2.1 Direct e-beam write	27
3.3 Fabrication of binary holograms	28
3.3.1 Fabricating the masks	28
3.4 Fabrication of holograms used in this thesis	29
3.4.1 Coating the substrates with photoresist	29
3.4.2 Controlling the etch depth	30
3.4.3 Contact copying into photoresist	31

4	The direct-search method	34
4.1	A tailored search technique for the computer design of holograms . . .	34
4.2	Direct-search and simulated-annealing	34
4.2.1	Simulated-annealing	34
4.2.2	Direct-search	35
4.2.3	Computer generated hologram solution space and its impact on simulated-annealing and direct-search methods	35
4.3	The nature of the computer generated hologram solution space	36
4.4	An appropriate model for use with the direct-search method for the design of computer generated holograms	37
4.4.1	Polarisation, laser modes, scalar theory and the direct-search method	38
4.4.2	Other models	38
4.5	The underlying model used with the direct-search method	38
4.6	Sampling and space bandwidth product	41
4.7	Cost, starting, stopping and choosing functions	43
4.7.1	Cost functions	43
4.7.2	Starting functions	45
4.7.3	Stopping functions	46
4.7.4	Choosing functions	46
5	Methods	47
5.1	Direct-search approximations	47
5.1.1	The obliquity factor	47
5.1.2	Summation over pixels vs. integration	48
5.2	Assessment of the hologram design	48
5.2.1	Optical reconstruction	49
5.2.2	Propagation based on the angular spectrum of plane waves . .	50
5.2.3	Implementation of angular spectrum propagation method us- ing fast Fourier transform	51
5.3	Comparison of simulated reconstructions and optical reconstructions .	51
5.4	Analysis of simulations	52
5.4.1	The use of a mask function to divide the image	53
5.5	Computational methods	56
5.5.1	Round off errors during direct-search optimisation	56
6	The direct-search algorithm	58
6.1	The effect of basic model parameters	58
6.1.1	The effect of sampling.	58
6.1.2	The effect of the number of phase levels	66
6.2	Algorithm design: The effect of the starting, stopping and choosing functions	68
6.2.1	The effect of the starting function	68
6.2.2	The effect of the stopping function	73
6.2.3	Choosing functions	77
6.3	The effect of cost function design	91
6.3.1	Target-based cost functions	91
6.3.2	State-variables based cost functions	95
6.3.3	Phase dislocations and cost factors	97

6.4	Designing three-dimensional intensity distributions	100
6.4.1	Three-dimensional sampling	101
6.4.2	Focal length multiplexing	101
6.4.3	True three-dimensional designs	105
6.5	Designing arrays of points	107
6.6	Tailoring the design to the input illumination	110
6.6.1	Tailoring the design to the hologram aperture	110
6.6.2	Uniform and Gaussian input illumination	115
6.7	Comparison of fixed-target cost function and state-variable cost function based direct-search methods	116
7	Conclusions and future work	119
7.1	Conclusions	119
7.1.1	Existing work on computer generated hologram design	119
7.1.2	Hologram fabrication and importance to hologram design	120
7.1.3	Direct-search method and the underlying model	120
7.1.4	Computational and analytical methods	120
7.1.5	The direct-search method in practice	120
7.1.6	Summary of optimal direct-search method	122
7.1.7	Conclusion	123
7.2	Future work	123
7.2.1	Wavelength multiplexing	123
7.2.2	Improving the speed of computation	123
7.2.3	Non-scalar theory model for use with the direct-search method	123
7.2.4	Investigation of the effect of hologram fabrication defects	124
7.2.5	Gray scaling	124
7.2.6	Development of a model for predicting the number of optimisation cycles	124
7.2.7	Development of a specialised 1-D algorithm for pseudo 1-D devices	124
A	Derivation of the Kirchhoff integral.	125
B	Effect of square hologram pixels	127
C	Computational restrictions	130
C.1	Vectorisation and calculation of inverse tangents	131
D	Not so sensible cost functions	132
D.1	Very simple cost functions	132
D.2	Other target based cost functions	132
	Bibliography	125

List of Figures

1.1	Computer generated hologram concept.	12
2.1	Flowchart of a typical projection-on-constraints algorithm.	18
2.2	Flowchart of typical genetic algorithm.	19
2.3	Flowchart of a typical simulated-annealing algorithm.	20
2.4	Flowchart of a typical direct-search algorithm.	22
3.1	Diagram showing how an 8 level CGH is fabricated using 3 binary masks.	27
3.2	Diagram showing basic stages in mask manufacture.	29
3.3	A typical contact copying process.	31
3.4	Different optical paths through developed photoresist.	32
3.5	Interferograms of hologram made with interference microscope.	33
4.1	Flow chart of the direct-search method.	35
4.2	Typical CGH working layout	37
4.3	Schematic diagram showing the pixellated hologram phase distribution.	40
4.4	Sampling parameters.	41
5.1	θ_a and θ_b used in the calculation of the obliquity factor.	48
5.2	Optical reconstruction apparatus.	49
5.3	Simulated and optical reconstructions for a 2 spot focus element.	52
5.4	Simulated and optical reconstructions for a 4x4 array focus element.	53
5.5	Simulated and optical reconstructions multi-line arc focus element.	54
5.6	The mask function used to divide up the simulated reconstructions.	55
6.1	Diagram showing required image shapes.	59
6.2	Foreground intensity vs. image sampling rate.	60
6.3	Noise vs. image sampling rate.	60
6.4	Mean intensity, noise and signal-to-noise ratio vs. image sampling rate.	61
6.5	Mean efficiency vs. image sampling rate.	61
6.6	CPU time vs. image sampling rate.	62
6.7	Reconstruction of designs made using different image sampling rates.	63
6.8	Average intensity and noise vs. hologram sampling rate.	64
6.9	Signal-to-noise ratio vs. hologram sampling rate.	64
6.10	Efficiency vs. hologram sampling rate.	65
6.11	CPU time vs. total number of hologram pixels.	65
6.12	Stopping parameter used at different numbers of phase levels.	66
6.13	Intensity and noise for different numbers of phase levels.	67
6.14	Signal-to-noise ratio for different numbers of phase levels.	67
6.15	Efficiency for different numbers of phase levels.	68

6.16	CPU time for different numbers of phase levels.	69
6.17	Intensity and noise during optimisation for two different starting functions.	70
6.18	Performance of holograms optimised using the direct-search algorithm optimised using different starting conditions.	71
6.19	Chart showing the CPU time required to perform direct-search optimisations using different starting conditions.	72
6.20	Ten holograms and reconstructions designed using different starting conditions.	72
6.21	Measured final efficiency and estimated efficiency, E_{finite} , vs. stopping parameter.	75
6.22	Intensity and noise vs. the stopping parameter.	75
6.23	Signal-to-noise ratio vs. stopping parameter.	76
6.24	Efficiency vs. stopping parameter.	76
6.25	CPU time vs. stopping parameter.	77
6.26	Intensity and noise for different pixel selection procedures.	79
6.27	CPU time for different pixel selection procedures.	79
6.28	Intensity and noise during optimisation for different pixel selection procedures.	80
6.29	Intensity and noise for different fringe following levels.	81
6.30	CPU time vs. fringe following level.	82
6.31	CPU time per hologram pixel for fringe following levels.	82
6.32	Intensity and noise vs. total number of hologram pixels.	83
6.33	CPU time vs. total number of hologram pixels.	84
6.34	Intensity and noise for five different phase selection procedures. . . .	86
6.35	CPU time required for five different phase selection procedures. . . .	87
6.36	The mean image sample intensity during optimisations for five different phase selection procedures.	88
6.37	Image sample noise during optimisations for five different phase selection procedures.	88
6.38	The probability of accepting a change during optimisations for five different phase selection procedures.	89
6.39	Graph showing the efficiency vs. target.	92
6.40	Intensity and noise vs. target.	93
6.41	Signal-to-noise ratio vs. target.	93
6.42	Signal-to-noise ratio vs. dynamic-target parameter.	95
6.43	Intensity and noise vs. b ($a = 2$).	96
6.44	Signal-to-noise ratio vs. b	97
6.45	Signal-to-noise ratio for three different design problems vs. b	98
6.46	Signal-to-noise ratio vs. d	100
6.47	Diagram showing reconstruction configuration.	101
6.48	Reconstructions of a 2-d design at various planes (left side) and of tailored 3-d designs at the same planes (right side).	102
6.49	Mean intensity vs. separation of reconstruction planes.	102
6.50	Noise vs. separation of reconstruction planes.	103
6.51	Signal-to-noise ratio vs. separation of reconstruction planes.	104
6.52	Efficiency vs. separation of reconstruction planes.	104
6.53	Optimisation time vs. separation of reconstruction planes.	105
6.54	3-d diagram showing the image samples	106

6.55	A 3d design reconstructed over its design surface.	107
6.56	A 2-d design reconstructed over the parabolic surface shown in figure 6.55.	108
6.57	A CGH and an array of lenslets	109
6.58	Image sample intensity for array designs.	110
6.59	Hologram and reconstruction for “2x2 array” design.	111
6.60	Hologram and reconstruction for “4x4 array” design.	111
6.61	Hologram and reconstruction for “8x8 array” design.	111
6.62	Reconstruction of “8x8 array” design with arrows indicating the com- plex amplitude at the image sample points.	112
6.63	Diagram showing designs tailored to an aperture reconstructed using different apertures.	113
6.64	Diagram showing signal-to-noise ratio for designs tailored to specific apertures reconstructed with different apertures.	114
6.65	Comparison of results for uniform and Gaussian illumination recon- structed using uniform and Gaussian illumination.	115
6.66	Plot of the intensity along the line of the alpha shapes shown in figure 6.65.	116
6.67	Simulations and designs for fixed-target and state-variables cost func- tions.	118
A.1	Diagram showing Kirchhoff integral vectors.	126
B.1	How the hologram phase is constructed	128
B.2	The effect of sampling and pixellation	129

Chapter 1

Introduction

In 1948 Dennis Gabor [1] proposed the hologram as a novel imaging technique for electron microscopy. He demonstrated that it was possible to reconstruct an image of an object from the recorded interference pattern of a wavefront scattered by an object and a coherent reference wave. The Gabor method recorded the hologram on axis. Consequently when these holograms were reconstructed unwanted orders were present overlapping the object wave. In 1963 the problem of overlapping orders was solved by the use of off-axis recording which allowed the object wave to be spatially separated from the other orders [2, 3].

Practical work on holography was severely hampered by the absence of high-power coherent light sources [4]. The invention of the laser [5] allowed the development of holographic techniques which are now applied over a broad range of subjects. These include display holography, microscopy, interferometry, imaging through aberrating or turbid media, optical information processing [6, 7, 8, 9, 10, 11] and “synthetic image construction”.

It is with this last application, synthetic image construction using computer generated holograms (CGHs), that this thesis is concerned. The general problem is the generation of a required image from a known incident wavefront using one or more optical components. Essentially one dimensional problems, for instance those with radial or linear symmetry, can be solved very easily; simple examples of this include a single point focus and a single straight line focus. More complicated problems requiring the generation of wavefronts not possible with simple refractive optics can be solved using diffractive / holographic optics.

1.1 Diffractive optical elements (DOEs) and CGHs

There is no definitive distinction between DOEs and CGHs but it is possible to draw a distinction between these two categories by considering what sort of information the device imparts on the incident wavefront and what the intended application is.

DOEs are optical devices that operate using diffraction, these include CGHs. They include diffractive devices that manipulate the information already present in the incident wavefront to form a useful image, examples are inverse filters [7, 9, 10, 11] and zone plates / diffractive lenses.

CGHs are a sub-set of DOEs that contain the information required to form the useful image and when illuminated with the correct wavefront (which is not considered to carry the image information). The image may also be formed with

the aid of additional optical elements, for instance a lens. CGHs are holograms that have been designed using some numerical technique.

1.2 Computer generated holography

Computer generated holography was demonstrated in 1966 by Brown and Lohmann with the detour phase hologram using an implicit carrier frequency. This technique used binary amplitude modulation and divided the hologram into cells each containing an aperture, the position of each aperture is determined by the required hologram phase distribution and the area of each cell determined by the required magnitude distribution of the hologram. Lee [12] gives a thorough review of early techniques for computer generated hologram design.

1.3 Applications of CGHs

This thesis is concerned with the development of a computational technique for the design of hologram phase distributions. These are used to produce useful distributions of light energy primarily from lasers.

There are many applications for such devices: laser machining [13, 14], optical interconnects [15, 16, 17, 18], laser beam shaping [19, 20, 21, 22] and laser ultrasonics [23].

These applications share a number of requirements. All require efficient transfer of energy from the incident wavefront to the useful resulting intensity distribution. Often it is important to eliminate the need for any additional optics (this can be especially important in high-power applications). Many of the applications involve “squeezing” the available light into a small area, for instance a line or pattern of dots in order to raise the intensity of these regions as much as possible. The tolerance and specification of noise in the image will vary between the applications. Some applications (for instance optical interconnects) require highly uniform intensity distributions.

1.3.1 Laser machining

With high power lasers [14] the image generated by a hologram could be used to cut, machine or surface treat materials [19]. There are advantages in cutting some materials using laser light [24, 25]. It is possible to do this by “scanning” a focused spot of laser light over the surface of the workpiece or moving the workpiece under a static spot in order, for example, to ablate or melt the surface away. This is comparatively slow, with typical scan speeds of a few ms^{-1} [14, 24, 25]. For machining mass produced objects the slow speed of scanning systems makes them prohibitively expensive to use. It is possible to replace a scanning system with an imaging device that produces the required pattern of light on the workpiece. This could be used as a “machine tool” or “stamp” which could perform the required cutting with high-power laser pulses.

High power lasers are expensive and any imaging system should use the light as efficiently as possible to perform the machining process. This rules out passing the laser light through a mask which is then imaged onto the workpiece because the mask would have to absorb most of the light. This is not only inefficient but the mask

itself would have to be able to withstand the high-power laser beam without damage. However this approach is possible with very small or highly sensitive workpieces, for instance, micro-machining.

Holographic phase elements are suited to laser machining applications. They are designed not to absorb any light. The holographic element can also take on the functions of other optical elements, such as lenses that may normally be required, greatly simplifying the high-power optics. It is also possible to use computer generated holograms to produce three dimensional images.

1.3.2 Optical interconnects

Optical interconnects are used to distribute light from one or more light sources to one or more light receptors. In these applications the light usually carries some signal or information with it and so optical interconnects can be thought of as analogous to wiring in electrical and electronic circuitry.

There are many applications for optical interconnects, but they share the basic requirements of transferring light from a signal source to a signal sink [19, 26]. Applications include optical fibre interconnections, launching light into optical fibres or bundles of fibres, matching sources of light onto detectors [27], optical computing including the distribution of computer clock signals [18] and coupling lasers.

Most of these applications require the light from a relatively simple source to be distributed into a pattern of discrete points. The generation of this sort of pattern is possible with purely refractive and reflective optical elements¹ or with arrays of lenslets but these approaches result in bulky optics and may be subject to problems that render them impractical for some applications²

Computer generated holograms have advantages for many applications. A single diffractive element can greatly simplify and reduce the size of interconnection optics. They are capable of producing smaller and hence higher intensity spots than arrays of lenslets³. An array of lenslets also requires uniform illumination to produce uniform intensity distribution of spots. Computer generated holograms can be tailored to produce a uniform array of spots without uniform illumination of the holographic element. It is also possible to specify the relative intensity of each individual spot.

1.3.3 Laser ultrasound generation

Ultrasound⁴ can be generated by directing a pulse of laser light onto the surface of the material [23]. The ultrasound is excited by the shear forces generated in the material by the localised expansion and contraction caused by the localised heating and cooling [23] resulting from the absorption of light energy.

¹Consisting of many beam splitters and lenses.

²For instance when distributing clock signals for computing applications it is important that the clock signal arrives in sync at all of the clock receptors. This can be difficult to arrange with refractive optics, especially in confined spaces.

³Arrays of lenslets consist of an array of small lenses packed onto one substrate, these are often made using lithographic techniques rather than by grinding. Arrays of lenslets have applications unsuitable for computer generated holography, for instance as wavefront sensors such as Shack-Hartman sensors used in adaptive optics applications.

⁴Surface acoustic waves or bulk acoustic waves.

Short ultrasound wavelengths⁵ can be generated with this technique depending on the laser pulse repetition rate and material [23]. The pattern of the light incident upon the surface of the specimen determines the initial ultrasound wavefront and therefore how it propagates/diffracts in the specimen[28, 29]. Refractive ultrasound optics are possible⁶ but involve altering the material or sample [23] or physically coupling to the an ultrasonic transducer. Clearly this is undesirable if the ultrasound is to be used to inspect or test the sample without contacting it.

Laser ultrasound generation is inefficient in terms of the energy transferred from the laser light into the material as ultrasound, most of the energy incident on the sample just results in heating. Consequently this application requires high laser powers and any loss of laser energy before it hits the surface of the material is highly undesirable.

Typical laser ultrasound applications require focused lines of light, either straight (for the generation of “plane waves”) or arcs (for the generation of “circular waves”, the equivalent of spherical waves in two dimensions) [29]. More complicated patterns can be used to enhance the generation of harmonics of the fundamental ultrasound frequency and thus generate ultrasound with wavelengths shorter than the fundamental (determined by the laser pulse repetition rate). These patterns require the basic distribution (a straight line or an arc) to be repeatedly displaced along the direction of propagation of the ultrasound by a whole number of wavelengths of the required ultrasound. This arrangement can also allow more laser power to be applied to the sample without damaging it as the energy is spread over a greater area[28].

In some materials (usually materials with crystalline structures) the ultrasound speed varies with the direction of propagation this may require more complicated patterns of illumination to generate the equivalent of “circular waves”⁷.

Such distributions as these can be very difficult to generate efficiently without the use of diffractive optics.

1.4 Introduction to the direct-search method

The direct-search method discussed in this thesis was developed specifically to design (phase) holograms that, without the aid of additional optics, would with good efficiency form a required image from a known incident laser beam.

These “stand-alone” holograms greatly simplify the whole optical system. This is particularly important in high-power, long-wavelength applications and where size and weight are important. However the direct-search method is not restricted to working without additional optics and, with minor modification it can work with more complicated optical systems.

The method was also required to work with “quantised” phase distributions. Quantisation arises from the fabrication techniques and in the worst (and most

⁵Ultrasound frequencies approaching Gigahertz can be generated this way, with corresponding wavelengths in the the μm range depending on material properties.

⁶In the case of SAW waves surface the ultrasound travels in a thin layer along the surface of the specimen, a depression or lump on the surface will increase the acoustic path length for a SAW wave travels across it. With surface acoustic and bulk acoustic waves changes in the material can change the speed of propagation through it and therefore refract the acoustic waves.

⁷These waves would not appear circular outside of the material but the initial wavefront should lie along a line of constant phase from the focus or “centre” of the wave.

usual) case results in only two levels of phase being available (binary phase) at fabrication. With this in mind the direct-search method was specifically designed to work with quantised phase distributions. The majority of the work presented in this thesis uses binary phase. This is often the easiest number of phase levels to fabricate and often the most difficult number of levels to produce high quality designs for. It is essential that a useful design algorithm is capable of working effectively with binary phase. A key feature of the direct-search method is that it recognises the quantised nature of the fabrication process and produces solutions containing only the quantised phase levels. This means that the algorithm is “aware” of the noise resulting from phase quantisation and is able to compensate for it to some degree.

The direct-search method is used to optimise a hologram phase distribution. This phase distribution is intended to modulate the incident wavefront (such as a laser beam) to form a useful image. The usual design problem involves changing a simple incident wavefront so that the resultant wavefront produces an approximation to a desired intensity distribution.

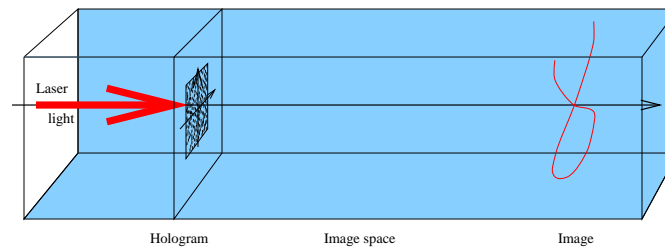


Figure 1.1: Computer generated hologram concept.

Figure 1.1 shows a simplified schematic diagram of the desired optical setup. In this diagram the laser beam travelling from the left passes through the hologram and enters an “image space”. The aim of the direct-search algorithm is to manipulate the quantised phase distribution of the hologram so that the required image intensity distribution (for instance the α shape indicated in figure 1.1) is generated as efficiently as possible from the light incident upon the hologram.

Many previous hologram design algorithms did not take the hologram fabrication process into account and many ignored the quantisation of the phase, the size of the aperture and the intensity distribution of the illumination. This can lead to a significant difference between the theoretical performance of the design (as far as the design algorithm is concerned) and the actual performance of the design when it is optically reconstructed. A major advantage of the direct-search method is that, provided the underlying mathematical model is accurate, the results returned by the design algorithm match the optical reconstructions well with no significant differences between the theoretical performance and the optical performance.

Chapter 2

Survey of existing work and methods

A brief survey of existing methods available for computer generated hologram design is presented. Many of the methods discussed below were developed partially to meet the limited computing resources available at the time. Consequently many of them rely heavily upon the use of the fast Fourier-transform (FFT) algorithm. This has, in practice, restricted the range of optical geometries and realistic modelling of the computer generated holograms designed.

2.1 Optically generated holograms

Before discussing methods of computer generated holography it is relevant to discuss optical recording of holograms and discuss reason why these may not be suitable for many of the applications mentioned.

Optically generated holograms are made by photographically recording the interference pattern produced when light scattered from an object (the object beam) is interfered with a reference beam. There are many different types of media that can be used to record the hologram. These include silver halide emulsions, dichromated gelatine (DCG), photoresist and specialist photo polymers.

The fringes that make up the interference pattern are typically sinusoidal, hence the exposure at the media is sinusoidal and consequently the developed amplitude or phase distributions are to some extent sinusoidal¹. In many cases the intensity of the fringes and the contrast of the fringes will vary across the hologram aperture. This may result in uneven development of the hologram. This restricts the use of optical recording to applications where uniform fringe brightness and contrast can be achieved².

The sinusoidal nature of the fringes limits the diffraction efficiency of such holograms to around 10% for binary amplitude and 40% for binary-phase [30, 31]³. It is possible to achieve higher efficiency using volume phase holograms in DCG but this has problems because of it is sensitive to a narrow range of wavelengths of light

¹The development of many materials is non-linear so the resulting developed pattern will generally not be sinusoidal.

²For instance the holographic production of gratings and some complicated optical elements like head-up displays.

³The figures quoted are the maximum theoretical diffraction efficiency defined as the proportion of energy diffracted into the +1 order.

and there is a tendency for the gelatine to shrink after exposure and thus change the fringe spacing.

As mentioned above optical recording requires an object and a reference beam to be present. Generating suitable object and recording beams optically can be difficult unless they are very simple.

It may also be difficult to find materials sensitive at the “playback” wavelength so that the optical recording may have to be made at a different wavelength and the object and reference beams altered to compensate for the aberrations resulting from the change in wavelength. The usual approach to achieving roughly uniform fringe brightness and contrast is to use a “diffuse” image, this tends to result in undesirable image speckle on reconstruction.

For high power applications the range of possible recording materials is extremely limited and those that are available tend not to be stable at very high powers.

When taken together the numerous problems encountered with optically generated holograms means that the technique is generally not useful for the production of phase holograms.

2.2 Analytical methods

2.2.1 Direct complex amplitude calculation

Direct complex amplitude calculation involves assuming what the ideal image complex amplitude distribution should be and then calculating what the wavefront leaving the hologram needs to be. This can be done by “back-propagating” the idealised image to the hologram. It can also be done by assuming the required wavefront for each part of the image and building the hologram phase distribution with a combination of simple zone plates[31].

Riley and Birkett [13] developed a technique to design binary-phase holograms for use with CO₂ lasers. Their aim was to produce a specified intensity distribution (in this case the letters “LUT”) for use as a laser machine tool. The required intensity distribution, “LUT”, was first sampled and represented as a pattern of points. In the subsequent calculations to produce a binary-phase distribution it was assumed that these points were coherently illuminated and that the light scattered from each point had a random phase added to it [13], (the object acts as a random diffuser). Riley and Birkett calculated the complex amplitude at the hologram resulting from the superposition of the scattered light from sample points at the image.

The hologram phase distribution ϕ_d , given by the direct complex amplitude calculation is given by equation 2.1.

$$\phi_d = \arg \left[Q \left(U \left(e^{i \left(\frac{2\pi r^2}{F\lambda} \right)} \text{FT} [A_i] \right) \right) \right] \quad (2.1)$$

where r is the radial distance across the hologram from the optical axis, F is the working distance from the hologram plane to the image plane, λ is the wavelength, A_i is the sampled complex amplitude (with random phase) of the image given by

$$A_i = a_i(x_i, y_i) \text{comb}_x \left[\frac{x_i}{x_s} \right] \text{comb}_y \left[\frac{y_i}{y_s} \right] e^{i\phi_{\text{rand}}} \quad (2.2)$$

where x_s and y_s are distance between image samples in the x- and y-directions,

ϕ_{rand} is a random phase and $a_i(x_i, y_i)$ is the desired image amplitude at (x_i, y_i) . comb_x and comb_y are comb functions as defined in Goodman [4]. The quantisation operator, Q , is given by

$$Q(ae^{i\phi}) = \begin{cases} ae^0 & 0 < \phi < \pi \\ ae^{i\pi} & \pi < \phi < 2\pi \end{cases} \quad (2.3)$$

and the uniform intensity operator, U , is given by

$$U(ae^{i\phi}) = e^{i\phi} \quad (2.4)$$

The hologram phase distribution was calculated by taking the two dimensional discrete Fourier-transform of the image sample points to give a bandwidth limited complex amplitude distribution, which was repeated (or tiled) to fill the entire hologram aperture. A quadratic phase term was added to the phase of the resultant complex amplitude [13]. The intensity distribution over the hologram aperture was assumed to be roughly uniform [32] and was set to a uniform value, i.e., only the phase information was kept. This calculated phase distribution was continuous and it was subsequently quantised according to equation 2.3. The resultant bandwidth limited binary-phase distribution was then used to fabricate the hologram.

The hologram was manufactured by etching the binary-phase pattern into a stainless-steel substrate which was then polished so that the desired phase change was achieved for incident light reflected by the substrate (hologram).

Failure to sample the image adequately meant that the reconstruction consisted of well separated points rather than a continuous intensity distribution. By setting the image sample phase but not optimising the hologram phase distribution this method failed to exploit the phase distribution in the image as a partial degree of freedom.

The diffraction efficiency of the hologram was poor and the crude quantisation process⁴ introduced noise which degraded the image considerably. This method also assumed that the hologram was uniformly illuminated (for laser machining purposes the hologram would be illuminated by a laser beam with an intensity distribution across this beam depending upon the laser design). Consequently this method is unsuitable for high efficiency, high quality applications.

2.2.2 Geometrical transform holograms

Geometric transform holograms use an analytic function which maps the available energy in the hologram intensity distribution onto the image intensity distribution. This mapping can then be used as a basis for calculating a hologram phase distribution. With the exception of an image consisting of a single point there is no unique mapping function and it is often the case that some arbitrary boundary condition must be set in order to find a particular mapping [35].

⁴Other methods of phase quantisation such as error diffusion [33, 34] can trade off low frequency representation of the continuous phase with high frequency noise and result in better image reconstruction, as the noise is relocated out of the image area.

It is also noted that a continuous phase distribution can be accurately represented with a binary-phase distribution provided a carrier frequency is used, in this case the continuous phase is used to modulate continuously the position of the carrier fringes. Provided the carrier frequency is high enough the first diffraction order will carry the desired continuous phase [35, 36].

The geometrical transform function can be thought of as defining a bundle of discrete rays indicating the path of the light as it travels from the hologram to the image. The required wavefront is normal to these rays. Unless the bundle of rays is simply organised there will be no surface that is *continuous* and *normal* to the rays. Under this circumstance it is necessary to approximate the ideal surface with one that is discontinuous and approximately normal to the rays. The discontinuity can be localised by splitting the hologram into sub-apertures or facets and writing a continuous distribution into each. This approach shifts all the discontinuities to the edge of each sub-aperture. The interfaces between the facets and the phase differences between the facets introduce noise into the image.

Geometric transform holograms have also been implemented using spiral phase dislocations [35]. The spiral dislocations are carefully placed in a continuous phase distribution so that the resultant geometrical transform approximates the ideal one. The phase distribution is discontinuous at the centre of each spiral dislocation and this leads to “holes” in the intensity distribution at the image where a zero in the intensity occurs for each spiral dislocation [37].

This method is difficult to implement for general use. The geometrical transforms can only be found easily in a small number of special cases. The subsequent steps such as the separation of the grating vector field into continuous and discontinuous parts and the positioning of the spiral dislocations are problematic and restrict the usefulness of geometrical transform holograms to special cases.

Since the particular transform used has to be selected before any of the other steps are carried out, the particular solution arrived at may not be optimal, for instance it may require more spiral dislocations than another solution and therefore introduce more noise into the image. The spatial coherence of the hologram is limited to one facet or the space between the dislocations depending on the technique used. This effectively limits the hologram NA with respect to the size of the smallest image feature in much the same way as it is limited in an array of lenslets to the NA of one lenslet.

As with any methods that result in a continuous hologram phase distribution the phase must be quantised to suite the hologram fabrication process. This can be done using any of the techniques mentioned. Paterson [35] used a carrier frequency to encode the phase and spatially separate the different orders upon reconstruction so that the required +1 order could be isolated.

The method does not allow much control over or exploit the partial degree in freedom of the image phase distribution although choosing a sensible geometric transform tends to give a sensible image phase distribution.

2.3 Optimisation methods

2.3.1 Projection-on-constraints

The method of projection-on-constraints [38, 39, 40, 41, 26, 42, 43, 44] works by projecting a wave from an initial hologram design (usually with uniform intensity and a random phase distribution) to an image plane where the complex amplitude distribution at the image is restrained (usually the calculated intensity distribution is set to that of the required intensity distribution while the calculated phase distribution is retained). The wavefront at the image is back-projected to the hologram and

the resultant complex amplitude distribution at the hologram is restrained (usually the hologram intensity distribution is set to the required intensity distribution at the hologram and the phase distribution may be quantised, see “soft quantisation” below). This process is repeated a number of times (usually 10-15 times but occasionally many more) after which the resulting phase distribution at the hologram is retained, quantised if necessary, and used to produce the hologram.

If the phase is quantised at the hologram during optimisation, the convergence proof discussed by Gerchberg and Saxton [38] does not hold.

When the number of quantisation levels is low (for instance, binary-phase), the quantisation process can introduce so much noise to the hologram phase distribution that the algorithm fails to converge satisfactorily [45, 38]. A possible way to avoid unsatisfactory convergence is to vary the number of levels of quantisation during the optimisation process. Starting with continuous phase and then quantising with many levels (say 64), the number of quantisation levels is then reduced with each cycle of the algorithm until the desired number of levels is reached. The aim is to allow the solution to “relax” and not to destroy it by quantising it so severely at each stage of hologram restraint that too much of the information is lost. A typical scheme for “soft” quantisation is shown in table 2.3.1.

cycle	number of phase levels
1	continuous phase
2	continuous phase
3	continuous phase
4	64
5	32
6	16
7	8
8	4
9	2

Table 2.1: Soft quantisation scheme.

The method of projection-on-constraints has one significant advantage compared with many alternative methods: it can make efficient use of the fast Fourier-transform (FFT) algorithm to project between the hologram and the image. This significantly reduces the amount of computation time required for the projection stages of the algorithm since the FFT routine takes $N \ln(N)$ operations, where N is the number of pixels or samples making up the hologram, compared with the corresponding number of operations (N^2) required without the FFT routine. A consequence of the use of FFT routines is that the entire image field is computed at each stage rather than just a particular area of interest. This makes the routine suitable for the design of holograms where the image is spread over a large area, or when a grey scale image is required.

The algorithm has some drawbacks; for instance, it is difficult to control the phase distribution of the image as this contains the very information that the algorithm uses to converge. In this algorithm it is assumed that the samples that make up the image are independent. Unless the samples are optically well separated, in which

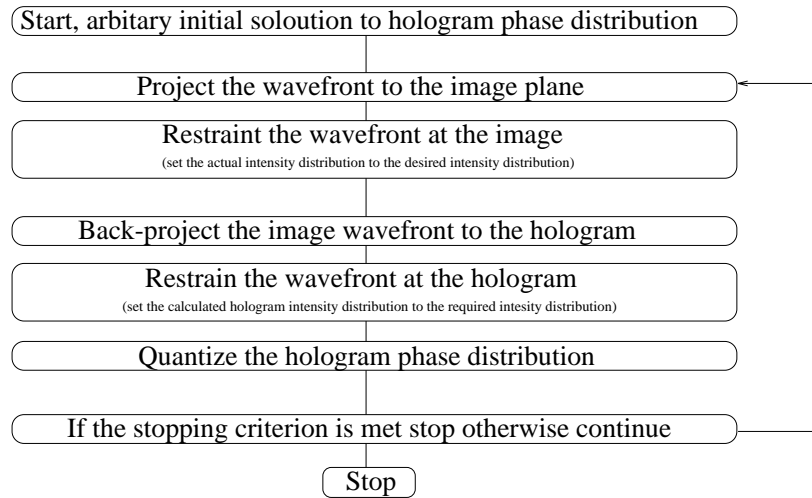


Figure 2.1: Flowchart of a typical projection-on-constraints algorithm.

case the image cannot be continuous, the hologram will exhibit additional speckle noise upon reconstruction [46].

Such holograms require a lens for reconstruction. This can cause considerable problems for some applications (for instance laser machining where the laser power is very high). A diffractive lens can be incorporated into the design after optimisation but this requires either additional quantisation or the use of a Fresnel propagation model rather than a far-field (Fraunhofer) propagation. The use of Fresnel propagation can itself introduce considerable complexity to the algorithm, especially if the hologram and image do not lie in parallel planes.

The projection-on-constraints algorithm can be considered computationally efficient provided the area of interest in the image is big (occupying a large proportion of the available image area), the number of phase quantisation levels is high (or some method of “soft quantisation” or “error diffusion” of the phase is used) and the hologram and image each lie within parallel planes. If either the hologram or the image is distributed over a volume rather than a flat plane normal to the optical axis then many additional transform calculations will be required in order to sample the image or hologram over this volume. This can dramatically increase the amount of computation required.

2.3.2 Genetic algorithms

Genetic Algorithms [47, 48] use an “evolution” approach to finding a solution for the hologram phase distribution. They attempt to express the information representing the hologram phase distribution as “genes” and then, by effectively “breeding” these genes and applying an evolution type “survival of the fittest” rule to the resultant genes, arrive at a solution. Figure 2.2 shows a simple flowchart of the process.

The starting point for the genetic algorithm is a “population” of many arbitrary solutions (typically 100 [47]). The phase distributions of the solutions are quantised and represent the genes of the solutions. The quality of the reconstructed image of each solution indicates how well the solution performs. The solutions are “bred” and “mutated” to generate new solutions. The “breeding” process involves

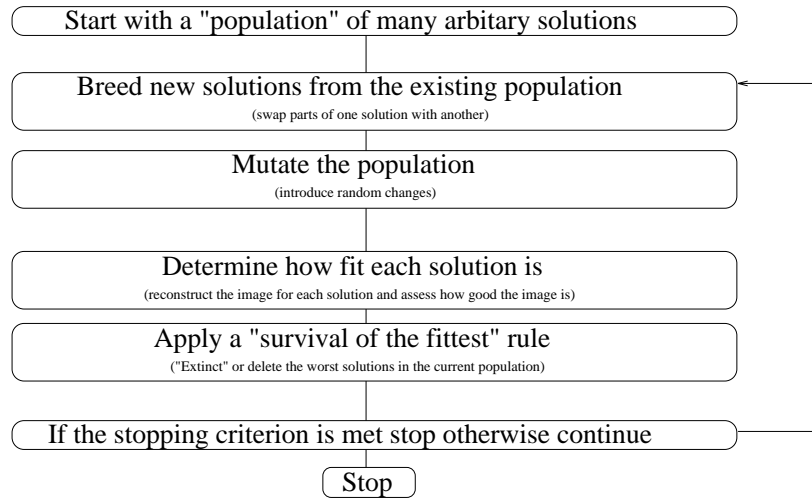


Figure 2.2: Flowchart of typical genetic algorithm.

exchanging parts of the hologram phase distribution from existing solutions to generate new ones. The mutation process involves introducing random changes to existing solutions⁵. The resultant population is a mixture of the existing solutions and new solutions generated from the breeding and mutation process. The images from this new population of solutions are then reconstructed and assessed. This is used to determine which of the solutions will “survive” and which will “die”. The surviving solutions represent a new population which is used to replace the existing population and the process of breeding, mutation and selection is repeated until some stopping criterion is met. This typically is when a member of the population (a solution) meets some predetermined performance criteria or the solutions stagnate [47].

The number of iterations required (or the number of generations of solutions) is typically larger than with the projection-on-constraints method (see Section 2.3.1, page 16) (for instance 5000 iterations for the genetic algorithm compared with 10-15 for projection-on-constraints). The amount of calculation required for each iteration is also high, with one propagation required for each member of the population. The total amount of computation is therefore very high, even when relying on the fast Fourier-transform algorithm.

An advantage of the genetic algorithm is that it can avoid poor local solutions by keeping along with the “best” solution many not-so-good solutions. Convergence to the globally optimal solution will occur eventually with an arbitrarily large initial population and mutation, but not necessarily for small populations without mutations.

A large proportion of the optimisation time appears to be spent making the various members of the population “coherent” with each other. Without this coherency a combination of excellent genes may well result in a very poor solution. This will tend to mean that this technique does not efficiently use the redundancy of the absolute phase of the image as a degree of freedom with which to optimise the image.

The computational inefficiency of this technique means that it is not well suited to the computer design of holograms. Reported results using this technique to

⁵Many genetic algorithms used for CGH design do not use mutations.

date are unimpressive despite the very lengthy optimisation times. This is largely due to the small starting populations, the absence of mutation and the limited number of generations or optimisation iterations (chosen in an attempt to reduce the optimisation time).

2.3.3 Simulated-annealing

Simulated-annealing [49, 50, 51, 52] is a stochastic method that can find solutions to many different types of problem. It is particularly useful for solving “combinational” type problems. These are typically characterised by many variables. It is common for each variable to have a restricted number of allowed values that can be taken. The range of allowed values may or may not be altered by the value of another variable. Typically the solution may not vary smoothly with the change of any variable. This means that there is no sense of “downhill” and it is difficult to tell if a change will or will not improve the solution without trying it out.

A classic example of a problem suited to simulated-annealing is the “layout” or “routing” problem, for instance, finding the quickest route for a drill machine to drill 500 holes or finding an optimal way to lay out a hundred thousand electrical components. When solving this type of problem in practice it does not normally matter that the globally optimal solution is found, rather that a “pretty good” solution is found.

A flowchart of a typical simulated-annealing algorithm is shown in figure 2.3.

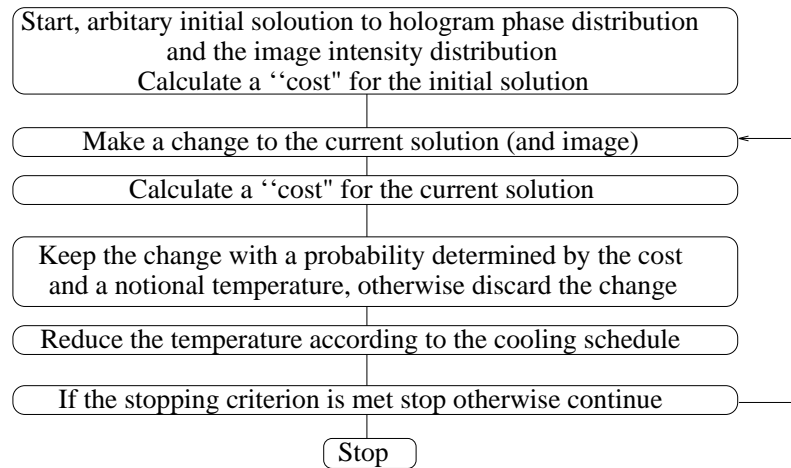


Figure 2.3: Flowchart of a typical simulated-annealing algorithm.

The simulated-annealing algorithm associates a “cost” with the solution. Essentially the cost is a measure of how well the solution performs, with the lower the cost, the better the solution. The annealing algorithm works by making a change to the current solution and therefore the cost of the current solution. The change is retained with a probability calculated from the cost and a notional temperature. When the temperature is high the chance of keeping a change even if it increases the cost (makes the solution worse) is close to unity, when the temperature is low the chance of accepting a change that increases the cost is low. A typical relation between the probability of accepting a change, P_{accept} , and the cost is given in equation 2.5.

$$P_{\text{accept}}(\Delta c) = \begin{cases} 1 & \Delta c \leq 0 \\ e^{-\{\frac{\Delta c}{T}\}} & \Delta c > 0 \end{cases} \quad (2.5)$$

where P_{accept} is the probability of accepting the change, $\Delta c = c_i - c_{i-1}$, (where c_i is the cost of the i th solution) and T is the notional temperature.

The value of the notional temperature, T , is initially high indicating that there is a high chance of accepting bad changes, the temperature decreases as the algorithm progresses until there is little chance of accepting a bad change.

When the temperature is low enough that the probability of accepting a bad change is effectively zero then the simulated-annealing algorithm becomes a basic direct-search algorithm where all good changes are kept and all bad changes are rejected.

It is noted that provided the initial temperature T is high enough (such that the initial probability of acceptance of a bad change is near unity) and the cooling schedule is slow enough⁶, then the probability of the final solution being the global best solution is unity [53].

Obviously an infinitely slow cooling schedule is out of the question in practice but the algorithm does “pretty well” provided there are enough iterations at each temperature for the current solution relax to a “quasi-equilibrium”⁷ at each stage before the temperature is reduced significantly again.

As the temperature drops, the time taken to reach “quasi-equilibrium” increases and consequently the time taken to reach the subsequent temperature drops increases exponentially [54]. To reduce the problem of waiting for the solution to reach “quasi-equilibrium” it is possible to wait only until the chance of having reached “quasi-equilibrium” is high and then reduce the temperature [54].

In practice deciding how to determine the cooling schedule and other factors concerning the simulated-annealing algorithm are problematic and often problem specific so that a set of rules which perform well on one particular problem may not perform well on a very similar problem. Methods that get around this problem by monitoring themselves and setting their own cooling schedules and other factors are known as “adaptive annealing” algorithms.

The problem of computer generated hologram design is effectively a “combinational” one where there are a very large number of variables and each one has a restricted range of values it can take; there is also no sense of “downhill”. There are considerable drawbacks with annealing-type algorithms with respect to their computational efficiency. Most annealing algorithms will spend most of their time calculating and accepting bad changes at random. In many circumstances this is necessary as along with the “pretty good” solutions and the global best solution there are numerous “pretty bad” solutions that are to be avoided. Typically solutions to many types of problems (other than CGH design) occupy a very small amount of the solution space and have no particular route from bad solutions to good solutions. Thus without the annealing process of accepting bad changes the algorithm could very easily get trapped in a poor local solution.

It is important to note that for computer designed holograms there are many good solutions, that these solutions are easily found and that they cover a large amount of the solution space. This makes the annealing process a time consuming

⁶In this case infinitely slow.

⁷This means that the number of changes accepted is approximately the same as the number of changes rejected.

luxury when designing holograms. The size of the computational problem encountered when designing holograms is so great that the usefulness of the annealing process is limited mainly to simple cases such as simple Fourier-transform interconnect holograms [55].

2.3.4 Direct-search algorithms

Direct-search algorithms [56, 57, 58, 59, 60, 61] work by making a change to the current solution, calculating the effect of the change and if the effect is desirable then the change is kept otherwise the change is discarded. This process is repeated until some stopping criteria are met. The way the changes are selected, the way the changes are made and how the effect of the changes are evaluated are crucial to this method.

It is obvious that, given a known set of direct-search parameters, the solution is determined absolutely by the starting conditions. This may seem to be a reason for using a simulated-annealing method, but as mentioned in Section 2.3.3 the availability of solutions when designing holograms is high. The amount of space in the solution space that each solution occupies means that there is a good chance of finding a good solution⁸.

A flowchart of a typical direct-search algorithm is shown in figure 2.4.

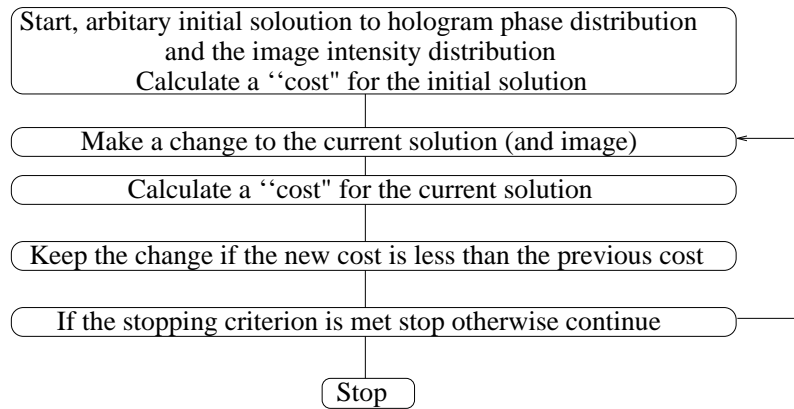


Figure 2.4: Flowchart of a typical direct-search algorithm.

It is fairly easy to demonstrate that each solution occupies a large amount of solution space [38] by considering the case of binary-phase Fourier-transform holograms. In a normal type of problem a particular solution would be represented by particular values of the variables. In the case of a Fourier-transform hologram this refers to a particular pattern of pixels with phase 0 and phase π . As the absolute phase of the solution is irrelevant any translation of the pattern of pixels is also irrelevant thus a particular pattern can be repositioned N times (where N is the total number of pixels making up the hologram) without affecting intensity distribution of the solution. This means that a particular solution has N different combinations of pixels representing it rather than 1.

⁸In the authors experience the algorithm has never failed to converge to a stable solution. The quality of the solution is determined by the design and the optimisation parameters and is, in practice, largely independent of the starting conditions, see Section 6.2.1.

With non-Fourier holograms the argument is not so clear, as moving from one combination to another affects the “quality” of the solution to a small degree. In practice finding any of the combinations of pixels that represents a good solution is good enough and the best of these combinations does not need to be found.

The direct-search algorithm is more sensitive to the optimisation conditions than simulated annealing. If the conditions are not well suited to the type of solution being sought then many more poor local solutions become available. Simulated-annealing algorithms may avoid these but the direct-search algorithm cannot. In the case of computer designed holography important considerations are the bandwidth and sampling of the hologram and image.

The design of the the controlling cost function is of great importance to the performance of the direct-search algorithm. It is essential that the cost function accurately discriminates between good and bad solutions and that it accepts as many good solutions as possible.

The direct-search method is fairly flexible. As the changes are made pixel by pixel there is no advantage in using FFT routines to calculate the effect of the changes so there is no computational advantage in using one particular optical set up over another. This allows the design of holograms that do not require any additional optics to reconstruct the image, such as a Fourier-transform lens. Additionally the image and hologram need not be located on planes parallel to the each other. It is perfectly possible to design the hologram and the image over curved surfaces or to design three-dimensional images in space.

Chapter 3

Hologram fabrication

This chapter describes some of the many hologram fabrication techniques available. It is intended to demonstrate that fabrication of the hologram should be considered from the very beginning of the design process. The fabrication technique restrains how the hologram can affect the wavefront. Failure to consider the effect of the fabrication at an early stage of the design process will result in unsatisfactory optical performance of the hologram because the design may not take into account the limitations of the fabrication technique and the design phase modulation will be distorted resulting in increased noise and decreased efficiency.

3.0.5 Holographic recording materials

The holograms in this thesis were fabricated using chrome-on-glass masks. The phase holograms were made using photoresist on glass substrates. Other methods of making phase holograms include the use of non-linear crystals, liquid crystals, photo-polymer coated glass substrates and bleached photographic film.

3.1 Overview of possible hologram fabrication techniques

Practical limitations such as the bandwidth, highest spatial frequency, number of phase levels and the accuracy of manufacture are of paramount importance to the successful optimisation of the hologram phase distribution. It makes no sense to develop a design method that is incapable of producing practical designs or to expend effort designing holograms that will not perform as well optically as they do in theory or simulation.

3.1.1 Direct write by laser plotter

Phase holograms can be written using a scanned laser spot to write directly into photoresist [62, 63]. This can then be developed to produce a phase hologram.

The exposure control can be binary [63] or continuous [62]. It is necessary to adjust the exposure levels to take the non-linear properties of the photoresist into account. Typical write speeds of 1000 dots per second [63] are reported for binary exposures which gives a print time of about 1 hour for each square centimetre of hologram.

This process can be used to make a high quality surface-relief master that may be used to press or cast copies. These copies can be of high quality and could be mass produced extremely quickly and cheaply [62].

3.1.2 Direct write via CRT

A microfilm plotter [16] has been used to write binary amplitude holograms directly onto film. It consisted of a cathode-ray tube to display an image of the required pattern. This was then imaged down onto high resolution film to produce the binary amplitude hologram. The quality and efficiency of the resulting holograms was extremely low and the process only provided a rapid way of making optical reconstruction¹ in order to test hologram designs rather than fabricate practical devices.

3.1.3 Etched metallic substrates

Riley and Birkett [13] fabricated holograms by etching a highly polished steel substrate. The binary phase hologram design was printed onto paper using a large-scale computer plotter². The design was then photo-reduced down to the correct size onto a photoresist-coated steel substrate³. After development the unprotected metal surface was etched away to give a binary surface relief pattern. The correct etch depth was achieved by etching to a greater depth than was required and then polishing the original surface until the correct depth was achieved. The quality of the hologram fabrication process appeared fairly good although the quality of the optical reconstructions were low because of the poor design strategy used. It may be possible to etch continuous surface profiles into metal. These can then be used as reflection holograms or as masters for pressing or casting copies.

3.1.4 Spatial light modulators

Spatial light modulators are devices that can change part of the incident wavefront under electronic control [64, 22].

They are commonly used as display devices in video projectors. They are generally pixel or bit-mapped devices and each element can usually be electrically addressed individually.

It is possible to use spatial light modulators [64, 65] to impart a phase shift into the incident wavefront. The required phase shift can be down-loaded from computer to the spatial light modulator and set at approximately video rate⁴. Both binary [65] and continuous [64]⁵ phase shifts have been achieved with fairly good (10%) phase accuracy. With binary spatial light modulators it is possible to get more phase levels by cascading different layers of spatial light modulators, each one imparting a

¹The reported efficiency was *so* low that it may have been a typographic error with a misplaced decimal point. The reported efficiency was 0.015% and was probably meant to be 0.015 or 1.5%.

²The plotting time was two hours.

³Riley and Birkett were working at a wavelength of 10.6 microns so the required feature sizes were comparatively large.

⁴For instance the process used by Scarbrough and Paige [65] required about 30 microseconds to down-load and set the data and then about 30 microseconds for the phase-shift to stabilise.

⁵Given the accuracy of the resultant phase shift, this device can be considered to have between eight and sixteen levels of phase shift.

different amount of phase shift. This, however, is technically difficult as the pixels in each stage have to be imaged upon the pixels of the other stages making the required apparatus required extremely bulky, fragile and hard to align compared with a prefabricated hologram. The resolution of currently available spatial-light modulators is low [66]. The devices themselves are expensive and delicate⁶ but their ability to set up the required phase shift quickly makes them useful for demonstration purposes.

It is possible that future devices may have higher resolutions and be robust enough for use in laser machining and holographic switching applications.

3.1.5 Direct machining of substrate surface

When working at a sufficiently long wavelength it is possible to machine the phase profile into a substrate [42]. This has been achieved to good effect in the microwave region where a continuous design phase was machined into a metallic substrate using a computer controlled milling machine. The phase distribution was unwrapped to remove 2π phase jumps which were too difficult to machine. This device also incorporated a “twist polariser”. This was a pattern of grooves with sub-wavelength spacing on the surface of the hologram that rotated the direction of polarisation of the incident light to the required polarisation

The hologram was used to convert a “doughnut” microwave cavity mode⁷ into a Gaussian-shaped intensity distribution. The Gaussian distribution required linear polarisation whilst the doughnut mode was polarised tangentially. The grooves of the twist polariser allowed the polarisation of the incident mode to be rotated to give the required polarisation⁸. This method is very expensive and currently only useful at very long wavelengths.

Micro-machining techniques are more generally applicable to hologram fabrication. These have more in common with lithographic techniques as they usually use some form of mask and machine / etch process.

3.2 Electron-beam micro-lithography techniques

Electron-beam micro-lithography can be used to fabricate computer generated holograms either by writing directly onto the hologram substrate or by writing a mask which can be copied onto the hologram substrate. At present electron-beam micro-lithography is a binary process. Multi-level holograms can be fabricated by the use of multiple exposures or multiple masks. The number of phase levels that can be fabricated using N writes or masks is 2^N .

Before each exposure the substrate must be coated with resist and if direct write by e-beam is to be used a metallic layer to conduct the e-beam current away. This is not a problem when the substrate is flat, but can be technically difficult when the

⁶Spatial light modulators tend to have temperature dependent performance.

⁷This is a linear combination of TEM₀₁ and TEM₁₀ modes, the intensity distribution is circularly symmetric and has zero intensity at the centre. A characteristic of this mode is that opposite sides of the distribution are out of phase with each other resulting in a spiral dislocation at the centre. The mode can have a variety of polarisations, including tangential and radial. This hybrid mode is often chosen for very high power applications as the mean cavity power density is very high.

⁸The device was designed to convert a doughnut mode with tangential polarisation to a Gaussian mode with linear polarisation.

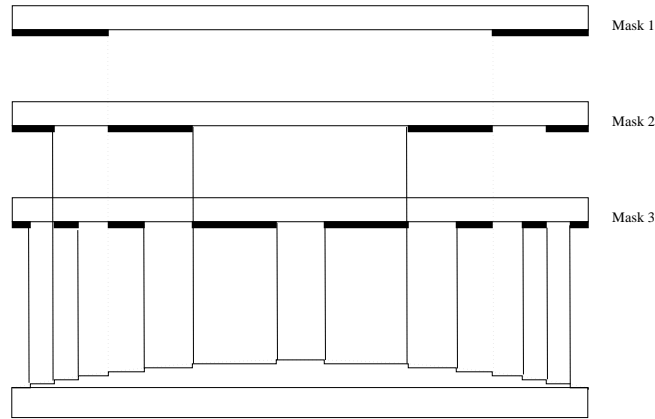


Figure 3.1: Diagram showing how an 8 level CGH is fabricated using 3 binary masks.

substrate has already been patterned. This does not affect binary designs that are only written once but can make the fabrication of multi-level designs difficult.

When making multi-level holograms with a binary writing process each resist exposure stage must be aligned with the previous ones. If the alignment is not ideal, additional features will be introduced into the hologram phase distribution. If these features are very small⁹ they will effectively result in a “rounding over” of the phase profile which will reduce the efficiency of the optical reconstruction probably by increasing the amount of light remaining in the zero order. Larger, but still small features will result in more light being diverted into higher diffractive orders. If they become too large, similar to the design features, they may prevent the hologram from working as it was intended.

3.2.1 Direct e-beam write

With a direct-write process the glass substrate under the chrome can be dry-etched to give a phase hologram¹⁰. This process can result in very high quality phase distributions, as the etched pattern in the glass follows the pattern written by the e-beam very closely. The dry etch process can also be controlled very accurately¹¹. If many copies of the same hologram are required the cost of writing the each copy of the design with the e-beam machine can become prohibitive.

Multi-level holograms

It is difficult to make multi-level holograms since the writing and development processes are binary and multi-level designs must use several write and develop processes which must be aligned to a high degree of accuracy. It is envisaged that the production of multi-level elements by direct e-beam write will become available in the

⁹Smaller than the reconstruction wavelength.

¹⁰The metal layer resists the dry-etch well so only the exposed glass areas are etched. The chrome is then removed to leave a clear glass surface with the required phase pattern etched into it.

¹¹The depth of a single etch can be controlled to within 5nm. This may allow control of the phase depth to within 1%.

near future¹².

Grey level resist process

It is also possible to write different levels of exposure into resist using a single e-beam write. The resist can be etched through to give different etch depths in the substrate. This process is difficult to control but offers the possibility of multi-level hologram fabrication with only one e-beam write. This would avoid recoating over a patterned surface and alignment between subsequent writes.

A substantial problem with direct e-beam write concerns the electrons back-scattered by the substrate, by choosing a substrate with very low back-scatter it may be possible to fabricate multi-level holograms without multiple e-beam writes. It may be possible to write into solid PMMA (perspex) which can act both as substrate and resist and has very low back-scatter. This process is not currently available but under development.

3.3 Fabrication of binary holograms

Binary holograms were produced by copying the high resolution designs, written on standard micro-lithography masks, into photoresist. These copies were relatively cheap and easy to produce. It would be possible to use such a binary hologram produced in photoresist as a master for producing copies by pressing and casting.

The holograms were made by first fabricating chrome-on-glass micro-lithography masks. The patterns on these masks were then copied into thick layers of photoresist coated onto glass optical flats. After development these were used as phase holograms.

3.3.1 Fabricating the masks

The e-beam machine used had a positional accuracy of about $0.1\mu\text{m}$ but the smallest feature size it could write was around $1\mu\text{m}$. The design was written by vector scanning a electron-beam spot over the mask.

Typically the masks were made of a glass substrate a few centimetres across and typically a couple of millimetres thick. The mask plates were coated with an optically dense layer of chrome and electron-resist.

Prior to e-beam writing, the hologram design may be processed to take into account the finite spot size of the electron-beam and the exposure properties of the resist.

After the e-beam write the resist was developed exposing the chrome layer where the e-beam pattern was written. The mask is then etched: where the resist remains the chrome is un-etched, where the resist has been removed the chrome is etched away. The mask is then cleaned to remove any remaining resist leaving the desired pattern in chrome on the mask. Figure 3.2 shows the mask plate during the fabrication stages.

E-beam machines can only write over a small area, typically one or two millimetres square. Outside of this area the aberrations in the e-beam focusing system result

¹²Multi-level DOEs have been reported but the processes are far from routine and have a limited chance of success especially with high resolution designs.

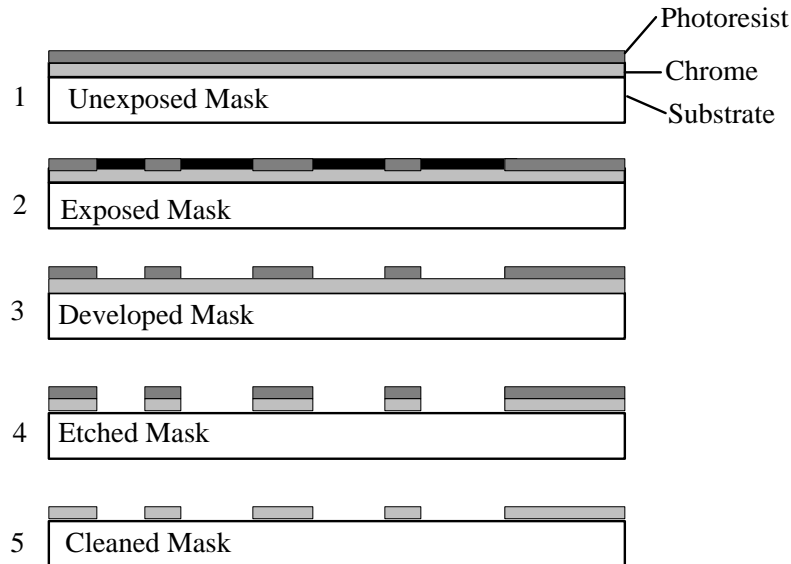


Figure 3.2: Diagram showing basic stages in mask manufacture.

in loss of resolution. Larger designs are made writing small patches and then moving the mask so that the next area can be written. The alignment of these stitched areas is typically within the positional accuracy of the e-beam machine. This is around $0.1\mu\text{m}$ and small enough not to be of concern when fabricating binary holograms¹³

The final design data is usually specified using a convenient format for the design program and needs to be converted into a format that the e-beam machine will take. The maximum size or bandwidth of the design is usually determined by the size of the data at this stage rather than the machine itself. The limits imposed by this process are often far larger than the limits encountered during the design process and are therefore not usually a problem. In practice it is possible to put several designs on one mask without encountering any significant problems with the total design data size.

3.4 Fabrication of holograms used in this thesis

The holograms for this thesis were fabricated by copying chrome-on-glass masks written using an e-beam machine at the Rutherford Appleton Laboratory. They were designed as binary phase holograms and the binary amplitude masks were copied into photoresist on glass substrates. The photoresist was then developed so that there was a difference in the optical thickness of the photoresist coating in the exposed and un-exposed areas of approximately $\lambda/2$.

3.4.1 Coating the substrates with photoresist

It is not possible to use the photoresist on a commercially available mask plate as a phase hologram as these substrates have a layer of chrome between the photoresist

¹³This is much more significant when making multi-level holograms using multiple writes as the “stitching” error places a limit on how well the subsequent exposures can be aligned.

and substrate and there is no way of removing the chrome without removing the photoresist. Additionally the photoresist layers used on commercial substrates are usually too thin to produce enough phase depth. It is necessary to copy the design made in the chrome layer into photoresist on another substrate. It is necessary therefore to prepare substrates by coating optical flats with layers of photoresist.

The substrates are spin coated with photoresist, the spin speed and the viscosity of the resist control the usable depth of the coating. The substrates need to be carefully cleaned and care must be taken when spinning to avoid contaminants such as dust from settling on the substrate or the resist. A dust particle can have a size much greater than the thickness of the resist and can also prevent the resist from adhering to the substrate.

The thickness of the coatings range from around $1/2\mu\text{m}$ to around $10\mu\text{m}$. It has been found that the important properties of a particular batch of photoresist such as the light sensitivity and viscosity vary slowly over time¹⁴ so that test coatings and exposures are necessary between batches of exposures. The faster the spin speed used during coating the thinner the coating, however it is necessary to spin quite fast (typically $> 1000\text{RPM}$) just to achieve an even coating. If the desired resist thickness is not possible at sensible speeds then a different viscosity resist is required.

Once the substrate has been spin coated it is baked at a moderate temperature (100°C) for about an hour. Baking helps to ensure that no solvent is left in the resist and help to relieve any stress locked into the resist as it dried during spin coating. This helps to improve the stability the resist properties and improve its uniformity.

3.4.2 Controlling the etch depth

Two methods were used to control the etch depth; either a thick coating was applied and the depth was controlled with the exposure and development conditions, or the depth of the coating was adjusted by adjusting the spin speed and developing completely through the resist.

Typical photoresist coating “recipes” are as follows.

Method one (thick resist)

Resist Shipley Microposit 1400-37, spin speed 3000RPM for 80 seconds, bake for one hour at 100°C , expose for about 10 minutes using a mercury source with an approximate intensity of $415\text{W}/\text{m}^2$, development in Microposit 303 developer diluted 1:8 in distilled water at 20°C for 30 seconds. This gives a thick resist coating several times thicker than the required etch depth.

Method two (thin resist)

Resist Shipley Microposit 1400-17, spin speed 3200RPM for 80 seconds, bake for one hour at 100°C , expose for > 15 minutes using a mercury source with an approximate intensity of $415\text{W}/\text{m}^2$, development in Microposit 303 developer diluted 1:8 in distilled water at 20°C for 30 seconds. This gave a resist depth corresponding to approximately a π phase change when the entire resist thickness was removed. Increasing the exposure time above 15 minutes did not affect the phase depth.

¹⁴A significant change may be noticed over a couple of months.

3.4.3 Contact copying into photoresist

In order to achieve accurate copies the chrome side of the mask must be in close proximity to the photoresist. Cleaning off the photoresist from around the edge of the substrate leaving just enough photoresist to receive the design helps to reduce any gap between the chrome layer on the mask and the photoresist. This improves the copying quality. If the gap between the photoresist and the chrome is too large, fringes caused by the light diffracting around the pattern in the chrome are observed and these are recorded into the photoresist. These can introduce a great deal of noise into the copy. The effect of this noise is more significant if the design is made up of small features

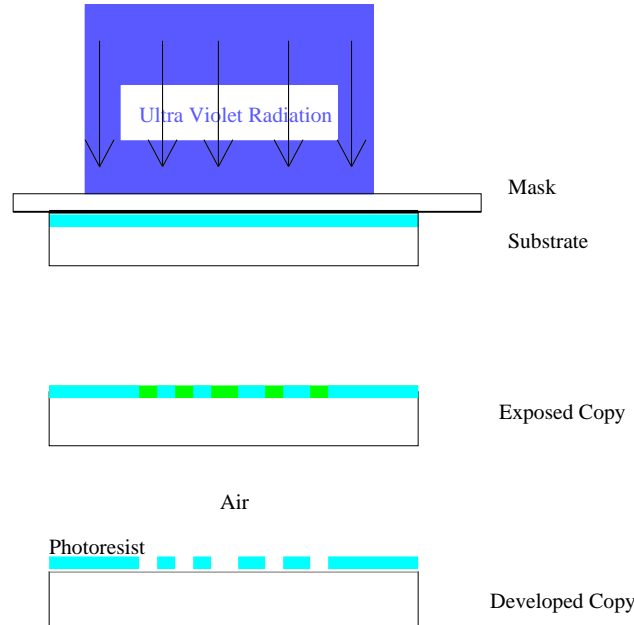


Figure 3.3: A typical contact copying process.

The light travelling through an area with a thick photoresist layer has its phase retarded compared with light travelling through a region with a thin photoresist layer. The phase difference is important to the final performance of the holographic element and it is important to control this depth accurately.

Figure 3.4 shows two paths of light through the hologram. Path A travels through the full thickness of the photoresist whilst path B travels through part of the resist that has been exposed and developed. For a binary phase hologram the desired phase depth is π which corresponds to a difference in the optical path length between paths A and B of $\lambda/2$. The optical path difference between paths A and B arises because of the differing path lengths through the resist and the difference between the refractive index of the resist and the surrounding medium (air). The required etch depth of the resist, D , is given as

$$D = \frac{\lambda}{2(n_{\text{resist}} - n_{\text{air}})} \quad (3.1)$$

where n_{resist} is the refractive index of the resist and n_{air} is the refractive index of air.

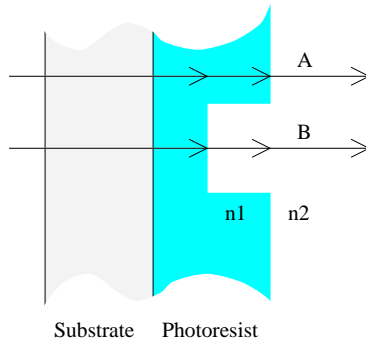


Figure 3.4: Different optical paths through developed photoresist.

For a hologram design for $\lambda = 633\text{nm}$ (He-Ne wavelength) and fabricated using the Shipley resists mentioned above ($n_{\text{resist}} = 1.6$) the required etch depth is approximately $0.5\mu\text{m}$.

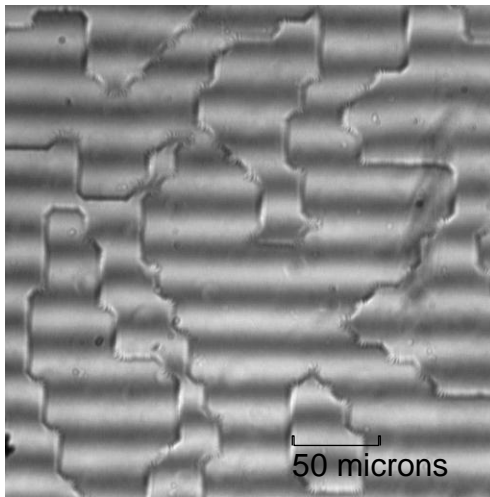
Controlling the phase depth is particularly important when high efficiency is required as much light is diverted into the zero order when the phase depth is not ideal [32].

In practice the depth was controlled by making test exposures of a grating mask and measuring the depth by measuring the displacement of fringes using a Mach-Zender interference microscope. It is quicker and easier to find the correct depth by adjusting the exposure and development conditions than it is by adjusting the thickness of the resist. However it is easier to control the thickness of the resist from one substrate to another. It follows that controlling the exposure is preferred when making one copy and controlling the resist depth is preferred when making many copies.

Figure 3.5 shows two interferograms of sections of holograms fabricated in photoresist. The phase depth was controlled by controlling the thickness of the resist. These pictures were taken using a CCD camera attached to a Mach-Zender interferometer. The first was taken using a mercury source, $\lambda = 546\text{nm}$, and shows interference fringes used to measure the phase depth of the photoresist. The second shows white light interference fringes used to ensure that the phase depth is less than 2π .

The phase depth was estimated using the interferograms as $148^\circ \pm 5^\circ$ at $\lambda = 546\text{nm}$ and $127^\circ \pm 5^\circ$ at $\lambda = 633\text{nm}$.

Interferogram of hologram



White light fringes

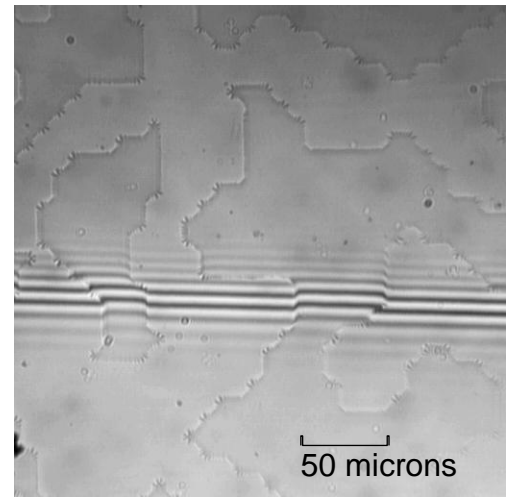


Figure 3.5: Interferograms of hologram made with interference microscope.

Chapter 4

The direct-search method

4.1 A tailored search technique for the computer design of holograms

The direct-search method presented in this thesis is a tailored optimisation technique. It shares similarities with some general optimisation techniques such as simulated-annealing but has been specifically designed to take advantage of the particular properties of the hologram design problem.

4.2 Direct-search and simulated-annealing

In many ways the direct-search method is similar to simulated-annealing¹. The structure of the two algorithms is very similar, the main difference being that where as in a simulated-annealing algorithm changes are kept with a probability dependent on the effect of the change and another variable, normally referred to as the temperature, in the direct-search method the changes are kept if the effect of the change is considered “good” and rejected if it is considered “bad”. Both of these algorithms use a “cost” or “merit” function to assess whether a change leads to an improvement of the image.

Apart from this the additional differences between the two algorithms can be considered as details that are manipulated in the direct-search algorithm to tailor it to the specific problem of hologram design.

4.2.1 Simulated-annealing

Simulated-annealing is discussed in some detail in Section 2.3.3. The most important feature of a simulated-algorithm is that it can accept bad changes with a probability that decreases as the algorithm progresses. The purpose of accepting bad changes is to reduce the probability of ending up with a poor local solution rather than the globally optimal one. It can be shown, given the right optimisation conditions, that the probability of the final solution being the globally optimal solution tends to 1 [49]. These conditions are computationally prohibitive and do not necessarily result in significant gains in terms of solution quality [16].

¹This algorithm is also referred to as the ‘Monte-Carlo’ method.

4.2.2 Direct-search

The direct-search method has no intrinsic way of avoiding poor local solutions. Thus it is important to design the algorithm so that poor local solutions are as far as possible avoided, reducing the chance of the algorithm getting trapped. This can be done by carefully considering how the cost is calculated, how the changes are selected, the initial starting conditions and the stopping conditions. The key to the success of this technique is the design of the function used to calculate the cost associated with the solution. This must obey three simple rules, it must admit as many good solutions as possible, it must reject bad solutions effectively and it must permit progress from bad solutions to good solutions. Careful consideration must be given to all aspects of the design of the algorithm so that ready progress from bad to good solutions is not impeded.

Figure 4.1 shows a simple flow chart of the direct-search method,

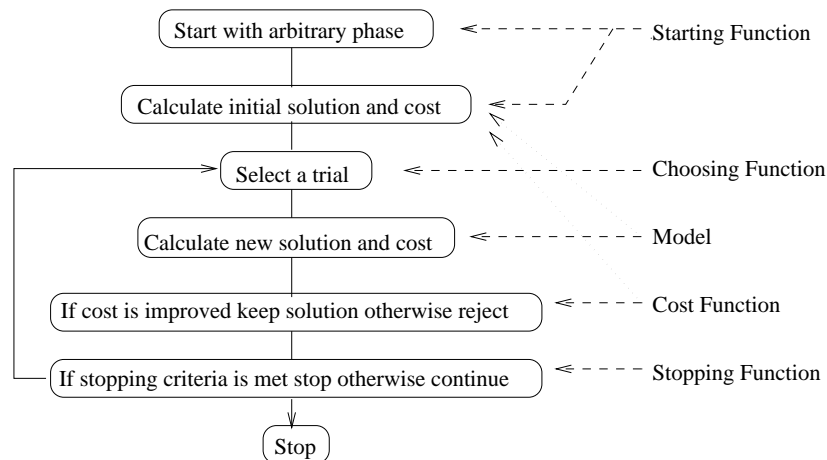


Figure 4.1: Flow chart of the direct-search method showing the algorithm divided into starting function, choosing function, model, cost function and stopping function.

4.2.3 Computer generated hologram solution space and its impact on simulated-annealing and direct-search methods

With quantised-phase computer designed holograms there are typically N equivalent permutations giving roughly the same solution (see Section 4.3) and any of these permutations can be reached from any other permutation without any significant drop in the quality of the solution. This can be thought of as a degeneracy at each solution. This degeneracy is the same for all solutions, so the solution space maybe thought of as being N times smaller than it would be without the degeneracy. However it is argued in Section 4.3 that the distribution of these degenerate solutions in the solution space may be more significant than the actual number of degenerate solutions.

As the degeneracy is the same for all solutions the probability distribution of solutions for simulated-annealing algorithms is unaffected and thus the performance of the simulated-annealing algorithm is unaffected. However this degeneracy (caused

by the lack of significance of the absolute phase of the solution) may help the convergence of direct search algorithm.

In practice the large amount of data that must be processed to design a computer generated hologram with a high enough space bandwidth product (see Section 4.6) to make a useful hologram means that simulated annealing can only be used for simple holographic configurations, e.g. Fourier transform holograms with simple images [55] and does not perform better than direct-search algorithms. Jennison, Allebach and Sweeney [56] found that simulated-annealing only marginally improved the performance of their direct-search method for computing Fourier transform holograms at the expense of a large increase in computation time [56]. They conjectured that this was because of the large number of solutions with near optimal performance. Hoptroff *et al.* found that their simulated-annealing program always returned reasonably good solutions even when the initial temperature was not high enough for the algorithm to return the globally optimal solution [16].

The cost function that is used to determine whether the change is good or bad can be carefully designed to reduce the chance of the direct-search method returning a bad local solution. This may hinder a simulated-annealing algorithm by increasing the time taken for the algorithm to reach equilibrium between temperature drops [49].

4.3 The nature of the computer generated hologram solution space

The computer generated hologram solution space has a structure that helps the direct-search method to converge reliably to good solutions without being trapped in poor solutions. The structure of the solution space helps to explain why general optimisation algorithms such as simulated-annealing do not necessarily out perform the direct-search method in terms of solution quality and tend to require considerably more computation time than the direct-search method.

Consider a hologram made up of N discrete pixels, each of which can take P different phase levels. After an exhaustive search through all of the possible P^N solutions the globally optimal hologram phase distribution, ϕ_{optimal} is found. If the particular problem concerns only the *intensity distribution* of the image then the *absolute phase* of the image has no physical significance. Hence any change in the absolute phase of the image, ϕ_{abs} , has no effect upon the intensity distribution of the image. Assuming that this solution to the hologram phase distribution, ϕ_{optimal} , represents the optimal coding of the optimal wavefront; it is possible to construct another wavefront by adding a constant absolute phase, ϕ_{diff} , to the optimal wavefront. Assume that ϕ_{diff} is chosen such that after optimally recoding the new wavefront the solution to the hologram phase distribution is the same as the original solution except that one hologram pixel has changed by one phase value. This process has then generated a slightly sub-optimal coding of the original optimal wavefront. This is another combination of hologram pixel phase values that represents the same overall solution. This process can be repeated to generate PN different combinations of hologram pixel phase values that all represent different codings of the same overall solution. After N changes the solution returns to its original form but with every

pixel one phase level higher than the original solution². These solutions are exactly the same as the original and are a “distance” in the solution space of N moves away from each other. After PN moves the solution returns to the exact coding sequence of the original.

This sequence can be thought of as a closed loop of high-quality solutions spread out on some complicated path through the problem space. This argument can be repeated for each instance of a good solution representing a different wavefront rather than for each solution representing a different coding scheme of the same wavefront. These solutions can be thought of as lying on a “string” through the solution space rather than being distributed randomly, evenly, or clustered together.

An additional effect of the redundancy of the absolute phase of the image is that each good solution can be represented by PN combinations of hologram pixel phase values, this may be thought of as making the solution space PN times smaller. However this reduces the size of the space by an insignificant amount compared with the number of possible combinations³.

4.4 An appropriate model for use with the direct-search method for the design of computer generated holograms

Figure 4.2 show typical working layouts for the sort of computer generated holograms designed in this thesis. In these working layouts there are no additional optical elements and the hologram must therefore supply both the imaging information and the focal power to the incident wavefront so as to form the image. The image is not restricted to two dimensions as demonstrated in Section 6.4.

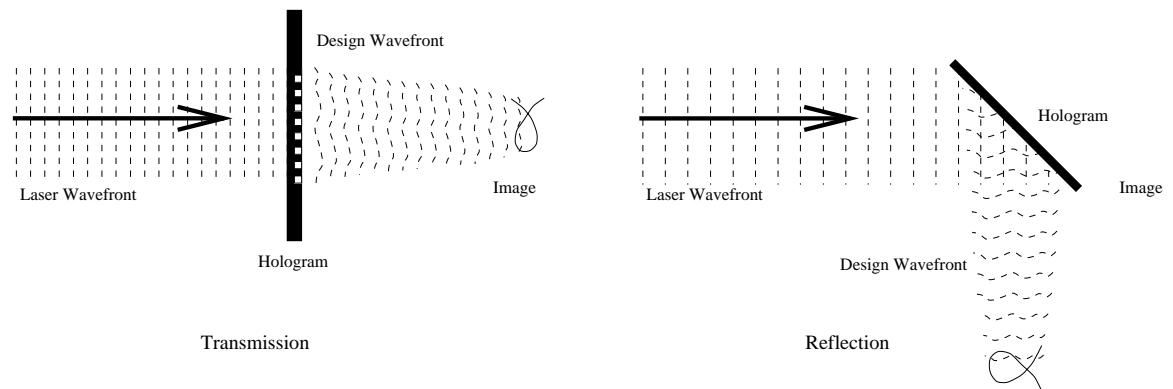


Figure 4.2: Schematic diagram showing a typical working layout for computer generated hologram applications. The laser light travels from the left to the hologram where the phase of the wavefront is modulated by the hologram design phase. The resultant wavefront then propagates to form a useful image.

²The phase values are $\text{mod}(\text{no. of phase values})$ and wrap back to zero accordingly.

³It changes the number of possible combinations of hologram pixel phase values representing a solution from the very large P^N to $P^{(N-1)}/N$, which is also very large (typically $N > 128^2$).

4.4.1 Polarisation, laser modes, scalar theory and the direct-search method

The following model assumes scalar theory for the light propagation and this sets certain limitations on the type of problems that can be tackled. One of the most important of these limitations is the polarisation of the incident light. This is often not a problem as most lasers are carefully designed to produce *linearly* polarised light (usually with a Gaussian beam profile). However for some applications notably *high power* laser machining applications the laser may be designed to produce more complex modes with more complicated polarisation distributions. An example of this is the “doughnut” mode, a hybrid mode consisting of a TEM_{01} and a TEM_{10} mode coupled together. This has a circularly symmetric intensity distribution and has a “hole” or dark patch at its centre. This mode is often used in high power applications because it has a high average power density over the laser cavity. This mode can have many different polarisations including linear, tangential and radial subject to the condition that the fields on the opposite sides of the centre of the distribution are out of phase. This results in a spiral phase dislocation [35] at the centre of the distribution.

If the light from the laser has linear polarisation a scalar approximation of the field can be taken (but the phase of the incident wavefront must be supplied to the design algorithm). However if the polarisation is complicated, separate coupled solutions may be found for each orthogonal polarisation.

4.4.2 Other models

The direct-search method is not restricted in usefulness to the following model. It can be used with other models but the quality of the solutions is of course restricted by the quality of the model in the context of the design problem.

Less rigorous models maybe more appropriate for simple design problems, for instance when the design problem can be adequately solved with an array of points of light and the optics permit the use of a Fourier transform lens. This may result in increased computational efficiency.

More rigorous models may be more appropriate for more complicated and demanding problems. These may require the hologram feature sizes to become very small, the focal length to become very short or that a complicated polarisation dependence be tackled. This would result in additional computational effort.

4.5 The underlying model used with the direct-search method

The model described here is firmly within the scalar regime, limiting its usefulness in extreme cases. Fortunately this model suffices for most CGH applications and represents an accurate account of the optical process in the direct-search method. This is essential or the method would converge to a non-physical solution which of course would not perform well optically.

The chosen model uses the Fresnel-Kirchhoff integral (referred to subsequently as the Kirchhoff integral). In Appendix A, this integral is derived from the Helmholtz-Kirchhoff diffraction integral so that the notation can be defined and the necessary

approximations can be examined.

Working from the usual form of the Kirchhoff integral

$$U_i(\vec{r}_i) \cong -i\lambda \oint_{\sigma} U_s(\vec{r}_s) \frac{e^{-ikR} [\cos \theta_s - \cos \theta_i]}{R} d\sigma \quad (4.1)$$

where $k = 2\pi/\lambda$, λ is the wavelength of the source radiation, \vec{r}_s is the position on the closed surface σ (see appendix A), U_s and U_i are the amplitudes at \vec{r}_s and \vec{r}_i respectively, θ_s and θ_i are the angles between \vec{r}_s and \vec{r}_i and the normal to the surface of integration \hat{n}_σ respectively, R is the distance between \vec{r}_s and \vec{r}_i and σ is the surface of integration. The time dependence of the amplitude is neglected for clarity.

This may be rewritten as

$$U_i(\vec{r}_i) \cong \frac{-ik}{4\pi} \oint_{\sigma} U_s(\vec{r}_s) \frac{e^{-ikR}}{R} [\hat{n}_\sigma \cdot \hat{n}_s - \hat{n}_\sigma \cdot \hat{n}_i] d\sigma \quad (4.2)$$

where \hat{n}_i and \hat{n}_s are unit vectors in the direction of \vec{r}_i and \vec{r}_s respectively.

By carefully choosing the surface of integration, equation 4.2 can be rewritten, approximately, as an integral over the finite surface of the hologram. All the other parts of the surface, σ , are considered to have a zero or a vanishing contribution to the integral. Simplifying equation 4.2 further, the computer generated hologram is considered to add only a design phase distribution, ϕ_d , to the source wavefront, $U_s(\vec{r}_s)$, given by $U_s(\vec{r}_s) = U_s(\vec{r}_s) e^{-i\phi_s(\vec{r}_s)}$ where $U_s(\vec{r}_s)$ is the amplitude distribution of the incident illumination and $\phi_s(\vec{r}_s)$ is its phase. A third phase term arising from the factor kR can be rewritten as ϕ_r , the phase between the hologram and image points. Substituting into equation 4.2 and rearranging gives

$$U_i(\vec{r}_i) \cong \frac{-ik}{4\pi} \oint_{\sigma} U_s(\vec{r}_s) \frac{e^{-i[\phi_r(\vec{r}_s, \vec{r}_i) + \phi_d(\vec{r}_s) + \phi_s(\vec{r}_s)]}}{R} [\hat{n}_\sigma \cdot \hat{n}_i - \hat{n}_\sigma \cdot \hat{n}_s] dS \quad (4.3)$$

The computer generated hologram will be assumed to be made of a collection of discrete pixels, each of which imparts a uniform phase to the incident wavefront (see figure 4.3). This is introduced so that an efficient numerical approximation can be made to equation 4.3. It does also in part represent the limitations of the fabrication processes that can be used to make computer generated holograms. This approximation is subject to the condition that the change in the optical path length introduced by a pixel does not change appreciably over the pixel (this must be less than $2\pi/N_p$ where N_p is the number of phase levels used to encode the hologram). This condition is usually met provided the correct hologram sampling conditions are met (see Section 4.6). Equation 4.3 can be then approximated by

$$U_s(\vec{r}_s) \cong \frac{-ik}{4\pi} \sum_{pixels} U_s(\vec{r}_s) \frac{e^{-i(\phi_r + \phi_d + \phi_s)}}{R} [\hat{n}_\sigma \cdot \hat{n}_i - \hat{n}_\sigma \cdot \hat{n}_s] \Delta S \quad (4.4)$$

where ΔS is the area covered by one pixel. This approximates the diffraction from a pixel device with diffraction from a collection of point sources with one point source placed at the centre of each pixel. Each point source emits the same energy as the pixel it replaces with a phase given by the source phase plus the design phase, $\phi_s + \phi_d$.

The effect of reconstructing the hologram with finite square pixels rather than

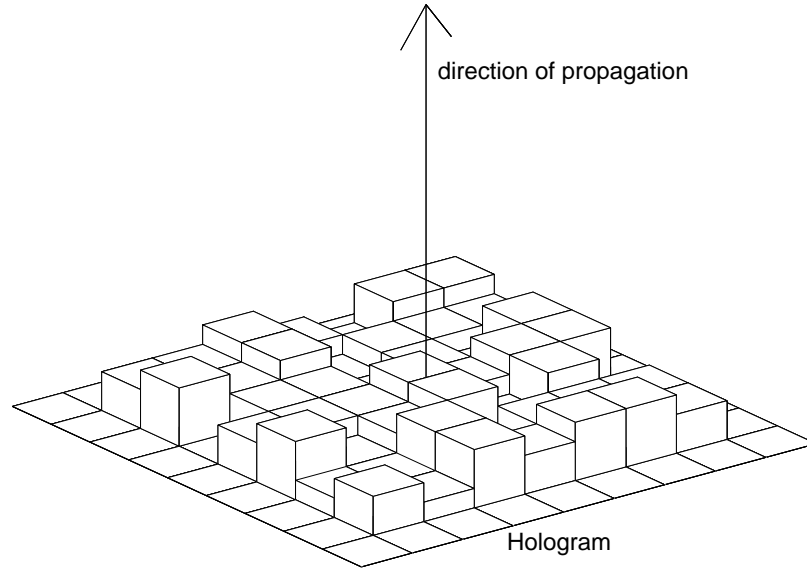


Figure 4.3: Schematic diagram showing the pixellated hologram phase distribution.

point sources is discussed in Appendix B. This has little effect on the design algorithm.

The obliquity factor term $[\hat{n}_\sigma \cdot \hat{n}_i - \hat{n}_\sigma \cdot \hat{n}_s]$ from equation 4.4 can be rewritten as

$$O = \frac{[\cos \theta_1 - \cos \theta_2]}{2} \quad (4.5)$$

where θ_1 and θ_2 are defined in figure 5.1. The obliquity factor is usually taken as unity (see Section 5.1.1).

The contribution of one pixel in the hologram to is given by

$$\Delta U_s(\vec{r}_s) \cong \frac{-ik}{4\pi} U_s(\vec{r}_s) \frac{e^{-i(\phi_r + \phi_d + \phi_s)}}{R} [\hat{n}_\sigma \cdot \hat{n}_i - \hat{n}_\sigma \cdot \hat{n}_s] \Delta S \quad (4.6)$$

Nothing in the above model restricts the surface of the hologram to a plane, the hologram could lie over any reasonable surface for instance a lens which would allow the hologram and the lens to “share” the focal power required to form the image.

The distribution of the incident illumination will be apparent to the model and consequently to the direct-search method, as a consequence the solutions returned by the direct-search method will be “tailored” to the illuminating intensity distribution and phase and the hologram aperture. This is demonstrated in Section 6.6.

Nothing restricts the image to the surface of a plane. Three dimensional image intensity distributions are possible. A single hologram can be designed to produce multiple two- and three-dimensional images at different working distances (focal lengths).

The ability to design three-dimensional images is demonstrated in Section 6.4. This also includes a discussion of the three-dimensional sampling requirements not discussed in the next section.

4.6 Sampling and space bandwidth product

The hologram and the image are sampled so that the numerical calculations required for the direct-search method can be carried out efficiently.

The sampling conditions are critical to the integrity of the model underlying the direct-search method. The optimisation will converge to a solution regardless of the model, if the model is faulty then the solution returned by the direct-search method will also be faulty.

When setting the sampling conditions consideration must be given to the fabrication and computational limits as well as the ideal optical performance of the hologram. Furthermore it is sensible to ensure that the approximations used in the direct-search model are still valid after the sampling conditions have been determined. If these approximations do not appear to hold for the sampling conditions it would be prudent to redesign the optics or consider upgrading the model thus eliminating either the challenging design conditions or the suspect approximations.

It is usual to determine the sampling conditions from the requirements of the image and the optics. These usually define the size or extent of the image, D_i , the smallest image feature size or image resolution, d_i and the working distance (or focal length) between the hologram and the image, F (see figure 4.4).

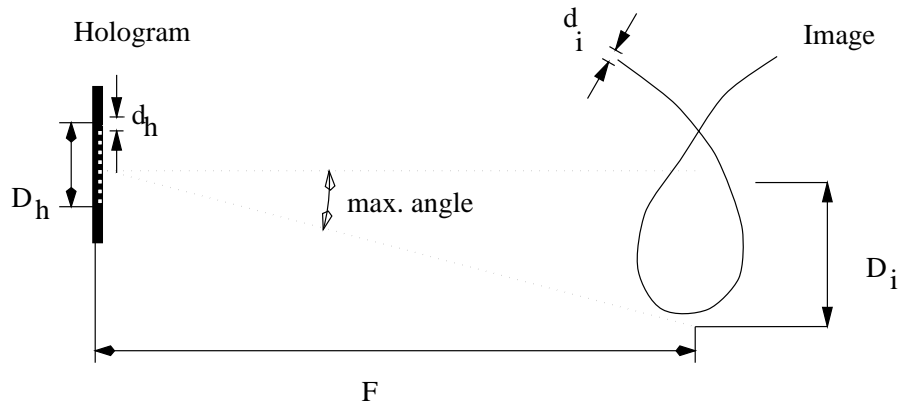


Figure 4.4: Schematic diagram showing the parameters used to determine the sampling conditions. The hologram is at the left of the figure and the illumination travels from the left, through the hologram and then propagates a distance F to the image.

The following sampling conditions are rules of thumb; for instance in figure 4.4 the maximum angular deviation of the light is shown from the centre of the hologram to the edge of the image, this is sufficient for most problems, but when working at high NAs or with off-axis images it might be necessary to use another construction to find the highest angular deviation of the light in order to prevent under-sampling of the hologram.

Working from figure 4.4 the smallest image feature size is related to the computer generated hologram numerical aperture (NA) by

$$d_i \approx \frac{1}{2} \frac{\lambda}{NA} \quad (4.7)$$

where the numerical aperture, NA is given by

$$NA \approx \frac{D_h}{2F} \quad (4.8)$$

thus the minimum hologram size can be expressed using the working distance, F , the image resolution, d_i , and the wavelength, λ , thus

$$D_h \geq \frac{\lambda F}{d_i} \quad (4.9)$$

The maximum pixel size at the hologram can be determined from the size of the image and the distance from the hologram to the image. These determine the maximum angular deviation, θ_{\max} that the hologram has to give to the incoming wave and therefore the highest spatial frequency, ν_{\max} , of the phase. If the hologram is to contain p_l phase levels, (and each fringe is to be successfully encoded using all levels) the hologram pixel (sample) size is given by

$$d_h \leq \frac{1}{p_l \nu_{\max}} \quad (4.10)$$

This can be rewritten in terms of the working distance, F , and the size or extent of the image D_i giving

$$d_h = \frac{\lambda F}{p_l \frac{1}{2} D_i} \quad (4.11)$$

The space bandwidth product is a measure of the total amount of information that can be carried by a system. For a computer generated hologram the space bandwidth product is the number of pixels (samples) contained within it. The space bandwidth product of the image will be taken as the number of separate resolvable image points that it can contain. This can be determined approximately by dividing the image extent by the image resolution. Working from equations 4.9 and 4.11 the space bandwidth product of the image, D_i/d_i , is the same as the hologram, D_h/d_h

$$\text{SBWP} \approx \frac{D_h}{d_h} \approx \frac{p_l \frac{1}{2} D_i}{d_i} \quad (4.12)$$

When the phase of the hologram is binary (or it is not essential to use all available phase levels in every fringe) equation 4.12 becomes

$$\text{SBWP} \approx \frac{D_h}{d_h} \approx \frac{D_i}{d_i} \quad (4.13)$$

The image must be sampled sufficiently finely to ensure that the reconstructed image is continuous. This means that continuous regions must be made up of overlapping samples and the maximum distance between samples in a continuous region is $d_i/2$, i.e., half the size of the smallest image feature (equation 4.7, see Section 6.1.1).

If the image is required to be an array of separated points, the separation of the points is must be greater than the resolution of the image. When this condition is met the phase of the image samples can be considered independent of each other. This can provide an extra degree of freedom with which the direct-search method

can optimise the image. Designs for arrays of points are shown in Section 6.5.

4.7 Cost, starting, stopping and choosing functions

The quality of the final solutions and the reliability⁴ of the algorithm depend to a large extent on the three functions that are used to regulate the algorithm during optimisation. These are the “cost” function, the “stopping” function and the “choosing” function. They control which changes are accepted, when the algorithm should stop and how the changes are selected respectively. These functions must be carefully designed so that the simple rules of optimisation, of accepting good solutions, rejecting bad solutions and allowing progress from bad to good solutions are ensured.

The cost function is responsible for selecting between good and bad changes. It must also behave in a simple clear fashion in distinguishing between bad and good solutions so that the chance of being trapped in poor local minima is reduced.

The cost function plays a vital role in the application of the three optimisation rules.

The starting function determines the initial state of the solution and algorithm before optimisation begins.

The stopping function is responsible for allowing the algorithm to continue optimisation whilst there are still reasonable gains to be made. It is therefore responsible for ensuring that the algorithm reaches a satisfactory solution before it terminates and ensuring that valuable computing resources are not wasted.

The choosing function regulates how the changes are made, it is responsible for selecting the hologram samples for examination and selecting the change parameters.

The stopping and choosing function play important roles in permitting progress from bad solutions to good ones. They can also have a significant impact on the computation efficiency of the algorithm by permitting progress from bad to good in a direct fashion. This also reduces the chance of the algorithm getting stuck in a poor solution by reducing the amount of problem spaced traversed during optimisation.

4.7.1 Cost functions

The design of cost function will influence the computation time both through the time it takes to calculate the effect of each change and how quickly it allows the algorithm to converge. The way in which it will reject poor solutions and admit good solutions will affect the quality of the final solutions and the reliability of the algorithm. It is also important that the cost function is stable, such that it will not converge to a solution that favours one part of the image to the detriment of other parts.

The principle cost function designs used in this thesis are investigated in Section 6.3.

⁴The reliability can be thought of as how often the algorithm returns a satisfactory solution.

Target-based cost functions

Most authors [16, 55, 56] using simulated-annealing, direct-search type algorithms or other optimisation techniques have generally reported using a simple cost function in which the cost was given as the sum of the squares of the differences between the actual intensity of the image points and the target intensity of these image points, thus

$$\text{Cost} = \sum_{\text{image points}} (I_{\text{actual}} - I_{\text{target}})^2 \quad (4.14)$$

The algorithm would then try to select a solution that minimised this cost.

Function 4.14 will be considered the starting point for investigation into cost functions. Some other, less useful, cost functions are discussed in Appendix D.

These *target-based* cost functions and variations and enhancements are discussed in Chapter 6.

Target-based cost functions, as their name suggests, require a target in order to work. This is problematic as a degree of knowledge and intervention is required to find the optimal value of the target for a specific problem. The solution to this is to allow the algorithm to vary the target (a dynamic-target). Target based cost functions are investigated in Section 6.3.1.

The behaviour of a target-based cost function also varies according to how close to the target(s) the calculated intensities are. Control of noise in areas of the image with a calculated intensity far below the target value is not good. This often occurs at early stages of the optimisation and can cause noise to be “locked” into the image, frequently in the form of phase dislocations.

It is difficult to predict the behaviour of target-based cost functions. The value returned by the cost function strongly depends on the target value. This can make it difficult to combine target-based cost functions with other cost functions. This can become even more difficult if the target value is allowed to vary during the optimisation (see Section 6.3.1).

Setting the target value to an appropriate level is also difficult as the algorithm is very sensitive to the target and predicting the optimal value for a given design is difficult⁵.

State variables cost functions

To counter some of the problems associated with target-based cost functions, a new type of cost function has been developed. This “state variable” based cost function aims to describe the distribution of image intensity by use of variables associated with the state of all of the image points. The *mean intensity*, \bar{I} , and the *standard deviation of the intensity*, σ_I , at the image sample points are suitable variables for the design of computer generated holograms. The former gives a measure of the strength of the image and the latter a measure of the noise in the image. These are linearly combined to give a single cost function thus

$$C = -a\bar{I} + b\sigma_I \quad (4.15)$$

where a and b are both positive numbers and are cost balancing factors which allow the relative importance of the two terms to be adjusted.

⁵This is likely to be set by trial and error.

This type of cost function has distinct advantages: the noise is controlled throughout the optimisation, it is easy to add in other cost parameters and there is no target to set. This is discussed in detail in Section 6.3.2 along with a discussion about the role of the image phase distribution as part of the cost⁶.

Computational considerations

The cost function should ideally be quick and easy to calculate. In the direct-search method the majority of the computational time is taken up with calculating the effect on the image of the changes introduced to the hologram phase distribution. Calculating target and state variable based cost functions requires a small amount of computation load compared with calculating the effect of the changes. Certain other cost functions may increase the computational load considerably and in such cases it is important to determine whether the benefits of the cost function are worth the extra computational load.

Gray scales and background light

The above cost function only require that the image is sampled at the points where light is required (the foreground). It is found that the areas where no light is required become darker as the energy is diverted into the brighter areas. This has two significant benefits. Firstly the number of image samples required during optimisation is reduced and secondly the background is “unrestrained”. The saving in calculation incurred by not sampling the dark areas can be considerable especially for image distributions requiring patterns of lines or dots. The lack of need to restrain the background tends to “free up” solutions, increasing solution availability.

Gray scales can easily be introduced with the fixed target cost function by varying the target intensities accordingly. Gray scales can be achieved with the state variable cost function by multiplying the intensity at each image sample with a weight proportional to the reciprocal of the required intensity.

4.7.2 Starting functions

The direct-search method “inherits” the noise inherent in the initial solution and tries to reduce it during optimisation. The initial image and its cost value must be calculated before the optimisation can begin, this can represent a substantial fraction of the required computational time⁷. Starting with all the hologram pixels “off”⁸ can help to reduce the inherited noise and save the computation required to establish the initial image (see Section 6.2.1).

This approach can be thought of as allowing the optimisation to begin as soon as the algorithm begins.

This does complicate the algorithm slightly as initially the solution must be permitted to be a *phase* and *amplitude* hologram and at some point the algorithm must force the solution to become a *phase only* hologram. This can be done by forcing the algorithm to turn on those remaining dark pixels after all the pixels have had a chance to be turned on. This is usually a small fraction of the total.

⁶The use of the image phase distribution as part of the cost was originally intended to help prevent some of the “locked in” noise apparent with target-based cost functions.

⁷Typically 20 percent of the total optimisation time.

⁸This means not emitting any light.

This may help convergence at the early stages of optimisation by allowing the algorithm to avoid poor local solutions.

4.7.3 Stopping functions

The stopping function determines whether the algorithm continues to optimise or terminates. The consequences of a poorly designed stopping function are stopping before the optimisation routine has reached the limit of its capabilities or continuing to try to improve an optimal solution beyond the limits of the optimisation routine thus wasting computational effort.

Three basic types of stopping strategy were considered: stopping after a fixed optimisation time, stopping when a fixed-target has been met and stopping when the solution stagnates.

The solution starts in an arbitrary condition consequently the time taken (in changes or trails) can vary significantly between different arbitrary starting conditions (see Section 6.2.1).

Thus a stopping function based on permitting the algorithm to run for a fixed time or number of iterations will prove unreliable and / or wasteful of computing resources.

The quality of the final solution(s) to any given problem is very difficult to predict. This means that a stopping function based on reaching a fixed-target of image quality will also prove unreliable.

Consequently it was found that stopping the algorithm when the solution stagnates produced reliable and predictable results. This type of stopping function is used throughout this thesis and plays a key role in making the direct-search method robust and reliable.

This stopping function is implemented by monitoring the rate of hologram pixels accepted and rejected, to provide an estimate of the probability of accepting a change at a moment in the optimisation. When the probability of accepting a change becomes smaller than a predetermined stopping probability, P_{stop} , the algorithm terminates. This is discussed in Section 6.2.2 along with a simple method of predicting the efficiency of the final solution from the stopping probability.

4.7.4 Choosing functions

The choosing functions are used to select the sequence in which the hologram pixels are changed (Section 6.2.3) and what value its phase is changed to.

These “choosing” functions can have considerable impact on the quality of the final solutions, the direct-search algorithm’s reliability and the computational efficiency. They play a key role in allowing progress from bad solutions to good solutions and must be carefully designed so that they do not block paths through the solution space as this may prevent access to solutions, increasing the likelihood of the algorithm getting trapped in poor solutions and increasing the number of optimisation cycles required.

The examination of the role of the choosing functions is complicated because of the effect the choosing functions can have on the rate of change of the solution and thus on the stopping function⁹.

⁹A very good example of this is demonstrated in Section 6.2.3.

Chapter 5

Methods

The direct-search method presented here is intended to be robust and practical. In order that it could produce CGH designs that were useful in practice it was important that the underlying model gave an accurate representation of the physical diffraction process to the optimisation algorithm. It was important to ensure the approximations used in the model and its derivation were appropriate to the regime in which the CGHs were to be used.

Numerical accuracy is important to the direct-search method. The method requires the addition of many small numbers to build up the image intensity distribution. It was important to ensure that this could be done with sufficient numerical accuracy¹.

5.1 Direct-search approximations

The model used through out this thesis for the design algorithm uses the *Fresnel-Kirchhoff* [4] scalar theory which treats light as a scalar field (see Appendix A).

The light is assumed either to be made up of just one polarisation component or separable non-interacting orthogonal polarisation components and neglects any interaction between these components. The assumption that light can be treated as a scalar field is valid so long as the diffracting aperture is large compared with the wavelength and the propagation distance is large compared with the wavelength [67] (Appendix A) The complex amplitude at the image samples is calculated using a form of the *Fresnel-Kirchhoff* diffraction integral (the Kirchhoff integral). The Kirchhoff integral also assumes that locally a *plane wave* approximation can be made(see Appendix A). This means that the model breaks down if very small features are present in the hologram.

The model is used with a couple of additional approximations, the first is that the obliquity factor, O , is taken to be unity (see Section 4.5), secondly the Kirchhoff integral is replaced for the sake of computation with a summation over the hologram pixels.

5.1.1 The obliquity factor

¹This problem can be particularly significant when using an optical system rather than a computational system to perform the direct-search optimisations [65]. It is particularly challenging to optically detect the effect that a single pixel change can have on the image intensity distribution especially when the hologram is made up of many pixels.

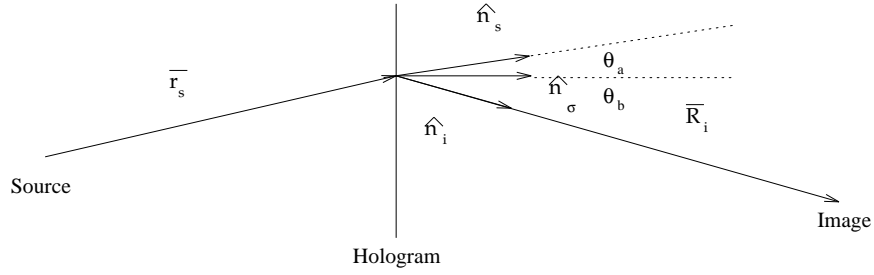


Figure 5.1: θ_a and θ_b used in the calculation of the obliquity factor.

The obliquity factor O can be given as

$$O = [\hat{n}_s \cdot \hat{n}_\sigma - \hat{n}_\sigma \cdot \hat{n}_i] \quad (5.1)$$

this can be rewritten as

$$O = \frac{1}{2}(\cos \theta_a - \cos \theta_b) \quad (5.2)$$

where θ_a and θ_b are defined in figure 5.1. Working from equation 5.2 and figure 5.1 it can be seen that unless the hologram has a large numerical aperture then the obliquity factor for a particular image sample is roughly the same for all the hologram pixels. For most applications the obliquity factor can be taken as unity ².

5.1.2 Summation over pixels vs. integration

For the sake of the computation the Kirchhoff integral is replaced with a summation over the hologram pixels. The equation used in computation (see equation 4.4) implies that rather than representing the Kirchhoff integral as a summation over sectional integrals that cover each pixel area, a summation of point sources emitting light with magnitude $|U_s|$ and the phase $\phi_s + \phi_d$ is used.

This is a valid approximation provided that the magnitude and phase of the complex amplitude of the hologram are reasonably uniform over each pixel and the phase between an image sample and the hologram pixel does not vary considerably across the pixel.

In practice limiting the size of the hologram pixels ensures that the phase and magnitude do not vary significantly over them. Provided the pixel size is small enough for the sampling conditions set in equation 4.10 to apply then these conditions are met.

The effect of using finite pixels rather than point sources is discussed in more detail in Appendix B.

5.2 Assessment of the hologram design

Reconstruction of the hologram is necessary in order to examine the resultant image. This can either be done optically or by simulation (digital reconstruction). In order to assess the design optically the hologram needs to be fabricated. Unfortunately this is a comparatively lengthy and expensive task. For the purpose of evaluating

² $\cos x \approx 1 - \frac{x^2}{2}$ if $x \approx 0.2$ radians (± 12 degrees) $\cos x \approx 0.98$.

hologram designs it is preferable to use a digital reconstruction as this can be done quickly and inexpensively. Furthermore digital reconstruction does not suffer from errors arising from the fabrication process and analysis is simplified by being able to locate the original image samples accurately in the simulated image.

5.2.1 Optical reconstruction

Holograms can be fabricated and optically reconstructed so that the resultant image can then be examined using some device to capture the image (usually a CCD camera). This captured image can then be analysed to determine the quality of the image. If the image is planar and of suitable size then it can be positioned directly on to the active surface of a CCD camera (see figure 5.2). This method is

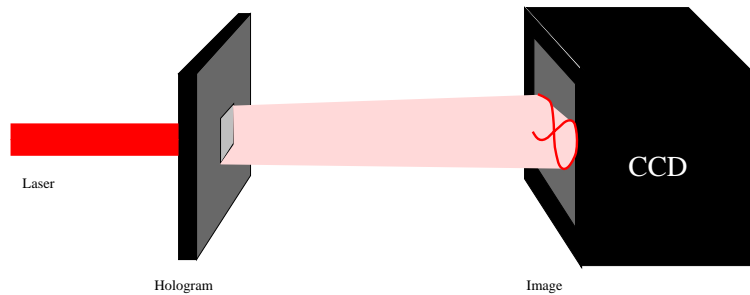


Figure 5.2: Optical reconstruction apparatus. The image from the illuminated hologram is placed directly onto the active portion of a CCD camera. The image may be (de)magnified using a lens if it is too small (too large).

not particularly useful for routine testing of the hologram designs because of hologram fabrication problems and experimental problems. The fabrication process (see Section 3.2) can be slow and is prone to error (especially phase depth error) unless tightly controlled. These fabrication errors can affect the efficiency of the hologram [32] and increase the amount of light going into the zero and other undesirable diffractive orders. Experimental measurement of the image can be difficult even if the hologram has been well fabricated.

In order to assess a design optically the hologram phase distribution must be masked off with an appropriate aperture, so that no light can travel from around the hologram design to the camera. The hologram designs presented below (see Section 6) are physically quite small with the apertures typically being 1mm across. This makes the problem of masking off the hologram aperture difficult especially as it is usual practice to make many hologram designs on a single mask and single substrate to reduce the individual hologram cost.

Simulated reconstruction

By using a simulated reconstruction many of the problems associated with the use of optical reconstructions can be avoided. The incident beam can be specified easily (including the aperture) without the need to construct specific experimental apparatus. The simulated reconstruction can be at the exact working distance, and the simulated image can be aligned with the original image samples (as used in the design process) easily. In order to assess properly the result of a particular design

it is necessary to calculate the image intensity over the entire image plane not just at the image sample points. This allows the analysis of the image to include the background area away from the design image samples.

Calculating the Kirchhoff integral as a summation over pixels (see equation 4.4) is an inefficient way of reconstructing the whole image. Instead the image is calculated using a Fresnel propagation method based on fast Fourier transforms as described in Section 5.2.2.

This method propagates the wavefront at a plane normal to the axis of propagation, to another plane normal to the axis of propagation. If the image lies in such a plane then the image complex amplitude at that plane is readily calculated, however if the image is three dimensional many propagations may be required to calculate the image complex amplitude throughout the volume which encompasses all the image points.

5.2.2 Propagation based on the angular spectrum of plane waves

The complex amplitude over the hologram plane can be represented as an angular spectrum of plane waves [4]. The angular spectrum can be thought of as a set of plane waves leaving the hologram plane and propagating at different angles. The propagation of these waves to the image reconstruction plane can be represented by adding a propagation phase, ϕ_p , to each of them. The image complex amplitude at the reconstruction can then be calculated from this new angular spectrum.

The angular spectrum of the complex amplitude distribution, A_0 , over the hologram plane can be given as,

$$A_0(f_x, f_y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} U_0(x, y) e^{-i2\pi(f_x x + f_y y)} dx dy = \text{FT}(U_0) \quad (5.3)$$

where $U_0(x, y)$ is the complex amplitude over the hologram plane and f_x and f_y are the spatial frequencies in the x and y directions respectively. The angular spectrum at a distance F perpendicular to the hologram plane and the image reconstruction plane, A_F is,

$$A_F(f_x, f_y) = A_0(f_x, f_y) e^{-i\phi_p(f_x, f_y)} \quad (5.4)$$

where the propagation phase ϕ_p is given by,

$$\phi_p(f_x, f_y) = \frac{2\pi F}{\lambda} \sqrt{1 - (\lambda f_x)^2 - (\lambda f_y)^2} \quad (5.5)$$

where F is the distance between the hologram and image planes.

The complex amplitude at the image reconstruction plane can be calculated from the angular spectrum at the image reconstruction plane by taking the inverse Fourier

$$U_F(x, y, F) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} A_F(f_x, f_y, F) e^{i2\pi(f_x x + f_y y)} df_x df_y = \text{FT}^{-1}(A_F) \quad (5.6)$$

5.2.3 Implementation of angular spectrum propagation method using fast Fourier transform

The angular spectrum of the hologram phase distribution is calculated using a fast Fourier transform. The discrete nature of the fast Fourier transform means that the hologram is sampled and the the angular spectrum is bandwidth limited. The hologram design is made up of discrete pixels and these can form the basis for the sampling of the complex amplitude distribution at the hologram plane. The complex amplitude at the hologram plane is given by

$$U_0(x, y) = \begin{cases} |U_s(x, y)|e^{i(\phi_s(x, y)+\phi_d(x, y))} & \text{within the hologram aperture} \\ 0 & \text{outside the hologram aperture} \end{cases} \quad (5.7)$$

where $|U_s|$ is the magnitude of the hologram illumination, $\phi_s(x, y)$ is the phase of the illumination and $\phi_d(x, y)$ is the design phase introduced by the hologram. The region of zero amplitude around the hologram aperture must be include to prevent aliasing. The sampled, bandwidth-limited hologram complex amplitude distribution is given by

$$U_{m,n} = U_0(x, y)\delta(x - \frac{md_x}{M})\delta(y - \frac{nd_y}{N}) \quad (5.8)$$

where d_x and d_y are the size of the hologram sampling region in the x and y direction respectively, $m = -M/2, -M/2+1, \dots, M/2-1$ and $n = -N/2, -N/2+1, \dots, N-1$ and M and N are the total number of samples over the hologram reconstruction plane in the x and y directions. The discrete angular spectrum of the hologram plane complex amplitude distribution is then given by its discrete Fourier transform (DFT),

$$A_{f_m, f_n} = DFT [U_{m,n}] = \frac{1}{\sqrt{MN}} \sum_{m=-M/2}^{M/2-1} \sum_{n=-N/2}^{N/2-1} U_{m,n} \exp -i2\pi \left[\frac{mf_m}{M} + \frac{nf_n}{N} \right] \quad (5.9)$$

where $f_m = -M/2, -M/2 + 1, \dots, M - 1$ and $f_n = -N/2, -N/2 + 1, \dots, N - 1$.

Because of the sampling the DFT of the function $U_{m,n}$ is equivalent to the Fourier transform of the bandwidth limited function $U = U_0(x, y) = U_0(x+d_x, y) = U_0(x, y+d_y)$, $U = U_0(x, y)$ for $-d_x/2 < x < d_x/2$ and $-d_y/2 < y < d_y/2$. The sampling region is effectively tiled across the hologram plane with tiling periods of d_x and d_y in the x and y directions. The sampling region must be large enough so that the propagated waves from each of these periods do not cross into and interfere with each other. The sampling region must be large enough to hold both the hologram and the image.

5.3 Comparison of simulated reconstructions and optical reconstructions

A small number of computer generated holograms were fabricated in order to confirm the similarity between optical and digital reconstructions and to demonstrate the direct-search methods capabilities optically.

The fabrication of some of these elements was not perfect, in particular the phase depth achieved was not exactly 180° . This reduced the efficiency and increased the amount of background light.

The images of the optical reconstructions shown in figures 5.3 and 5.4 were taken with a CCD camera placed at the image plane.

The positioning and size of the images and simulation differ slightly as a result of misalignment of the CCD camera, the aspect ratio and size of the CCD pixels. The optical reconstruction shown in figure 5.4 shows additional background noise,

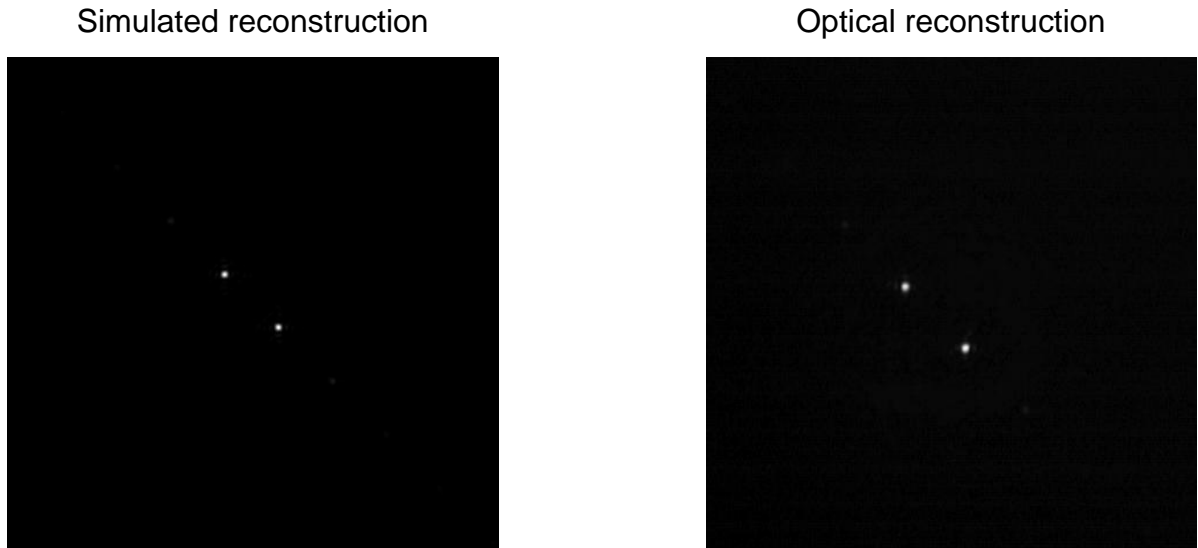


Figure 5.3: Simulated and optical reconstructions for a 2 spot focus element.

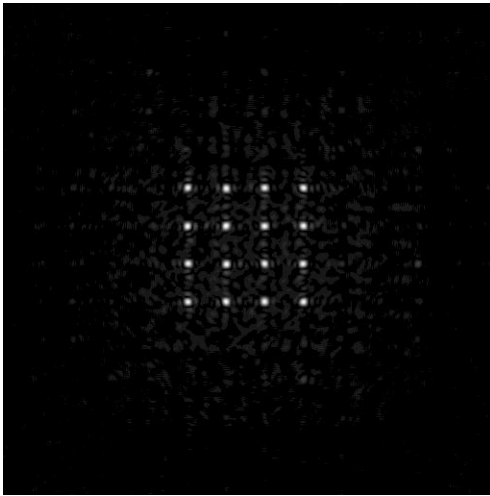
especially to the immediate right of the image. This results from misalignment of the hologram aperture allowing light from around the hologram to pass. This travels to the image plane and interferes with the image. This demonstrates some of the difficulties encountered when trying to use optical reconstruction for routine assessment of the hologram designs.

The aperture was constructed by placing opaque adhesive tape along the edge of the hologram designs. The holograms were 1.28mm square, the tape was positioned within about 0.2mm of the edge designs by hand using a magnified projection of the hologram as a guide. Care had to be taken not to touch the delicate hologram surface with the tape as it was placed. Figure 5.5 shows reconstructions of a multi-line arc hologram design for use in a laser-ultrasound experiment. The spacing between the four lines is 35 microns and the radius of curvature of the lines is 10mm.

5.4 Analysis of simulations

During the direct-search design process the image is sampled only in the bright regions. In order to assess the quality of the finished design, the full image is digitally reconstructed by taking an angular spectrum representation of the wavefront after it has passed through the hologram and propagating it to the image plane (or space in the case of three-dimensional designs). To ensure that the simulations are sensitive to the aperture of the hologram, the hologram wavefront is embedded in a large aperture which contains no light. The resultant simulated image reconstruction

Simulated reconstruction



Optical reconstruction

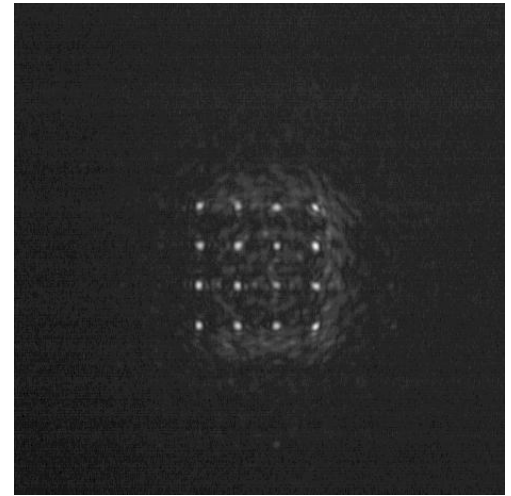


Figure 5.4: Simulated and optical reconstructions for a 4x4 array focus element. There is considerable stray light evident to the right of the image in the optical reconstruction due to misplacement of the hologram aperture.

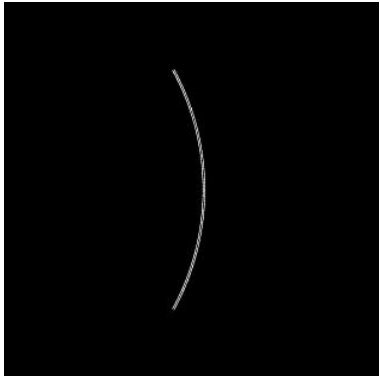
includes complex amplitude information for the dark background as well as bright regions of the image and can be used to assess the quality of the design.

5.4.1 The use of a mask function to divide the image

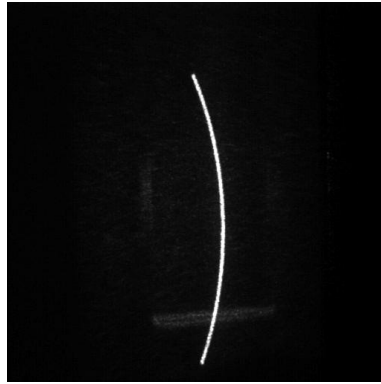
The simulated reconstruction is divided into three regions. The size of these regions is determined approximately from the size of the smallest image feature as determined by equation 4.7. The actual size of the smallest image feature is determined by aperture, focal length and wavefront at the hologram, however equation 4.7 was used for all the experiments regardless of the aperture shape or incident illumination. Most of the experiments used in chapter 6 use the same aperture, focal length and incident illumination (square, 100mm and uniform respectively).

The first region (the foreground) is centred on the image samples and includes all the area around the image sample up to \pm a quarter of the smallest image feature size, the second (the middle-ground) is centred in the same way but uses a distance of \pm a half of the smallest image feature size and the third (the background) is the region that remains after the subtraction of the second from the entire reconstruction (see figure 5.6). The first region represents the central part of the image line and is used to calculate the mean intensity along the line and the standard deviation along the line. The second region is used to calculate the efficiency of the hologram which is taken to be the fraction of energy in the second region divided by the total energy leaving the hologram (this does not take into account absorption and surface reflection losses that may be encountered in real reconstructions). The third region is used to calculate information about the background. In the following experiments the average intensity refers to the mean of the intensity in region 1, the noise refers to the standard deviation of the intensity in region 1 and the efficiency means the energy present in region 2 expressed as a percentage of the total energy leaving the

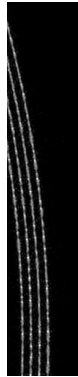
Simulated reconstruction



Optical reconstruction



Detail



Detail

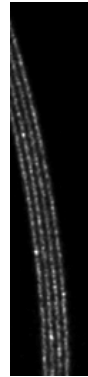


Figure 5.5: Simulated and optical reconstructions for a multi-line arc focus element. The lower images are details showing that the arc is made of 4 separate lines spaced by 35 microns. The hologram used in the optical reconstruction was fabricated by the Department of Electronics and Electrical Engineering at the University of Glasgow.

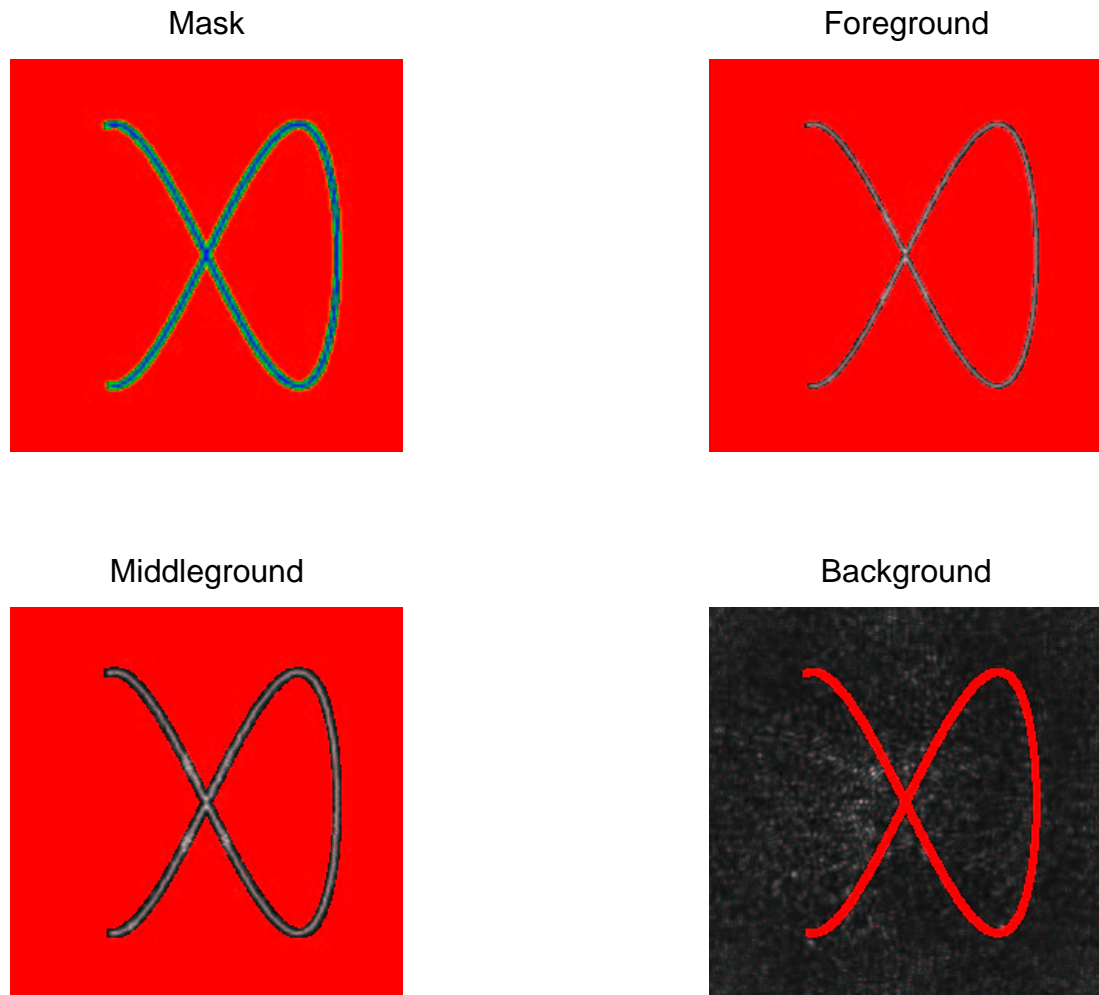


Figure 5.6: The mask function used to divide up the simulated reconstructions. The first picture shows the mask, red indicating the background, green indicating the middle-ground and blue indicating the foreground. The next three pictures show a simulated reconstruction divided into these three regions with red indicating the region that is ignored.

hologram³. For instance, if the hologram is a uniformly illuminated square, the ideal image line may be expected to have a sinc-squared type line profile. Using this method of assessment the average intensity would be approximately 0.83 of the peak intensity at the centre of the line, the standard deviation would be approximately 0.13 of the peak intensity and the efficiency would be 70% (70% of the energy falls in the region between the first minima of the line intensity profile).

In the chapter 6 the typical hologram pixel size was $10\mu\text{m}$ and the typical smallest feature size was approximately $50\mu\text{m}$. The placement of the masking regions can only be accurate to within the sample (hologram pixel) size. Comparison of the noise between experiments with different sampling-rate to image-feature-size ratios must be treated with some caution as it is possible to get jumps in the noise (and intensity) measured with this technique as the mask width will be predominately rounded up or rounded down according to this ratio.

5.5 Computational methods

5.5.1 Round off errors during direct-search optimisation

The most significant round off error encountered in this thesis is the round off error at the acceptance of a change of hologram pixel phase level. A typical double precision floating point number is represented using 64 bits, Typically 11 bits are used to represent the exponent, 1 bit the sign and 52 bits for the mantissa although these values are machine dependent. This gives a round off error in the region of 2^{-52} or 10^{-16} . The image complex amplitude is represented using complex doubles, which consist of a double precision floating point number for each of the real and complex parts. The total error after N changes can be given as

$$\left[\frac{\delta E}{E}\right]^2 \approx N \left[\frac{\delta e}{e}\right]^2 \quad (5.10)$$

where $\delta E/E$ is the total fractional error and $\delta e/e$ is the fractional round off error per change. Assuming that an accuracy of 1 part in a million is sufficient⁴ for the real and imaginary parts, then the number of changes that can be made before the round off error is significant is $> 10^{20}$. The worst case occurs when there is a constant one bit error with the same sign, in this case the number of changes that can be made before the round off error is significant is $\approx 10^{10}$. Even with the worst case the model is good for in excess of one hundred changes for each pixel in a 10000×10000 pixel hologram which is more than sufficient⁵. The round off error encountered when using single precision floating point variables (typically 32 bits with an 8 bit exponent) is of the order of 2^{-22} or about 1 part in ten million, this may suffice for very low band-width holograms but the round off error may well become significant. Another potentially significant error is encountered when calculating the phase along the path between a hologram sample and an image sample. For this purpose, the distance along this path can be taken as approximately F , the distance in the z direction. The phase is given as $|2\pi F/\lambda|_{2\pi}$. If $\lambda \approx 10^{-7}$ metres and the phase must be extracted

³This does not include losses such as surface reflection or substrate absorption.

⁴This means that a single pixel change in a 1000×1000 pixel hologram is significant.

⁵As a rule of thumb each hologram pixel is changed on average less than five times during a design.

to one part in a million then $F < 10^{-7} \times 10^{-6}/10^{-16} \approx 10^3$ metres. So when using double precision numbers there is sufficient accuracy to extract the phase along the path from the hologram pixel to the image sample even for holograms with very long focal lengths (when a far field approximation may be more appropriate).

Chapter 6

The direct-search algorithm

In this chapter the effects of the various parameters on the solution quality and computational efficiency are investigated. The ability of the direct-search generated solutions for various images and applications is demonstrated.

6.1 The effect of basic model parameters

6.1.1 The effect of sampling.

For the purpose of calculations used in the model during direct-search optimisation both the hologram and the image are sampled. It is convenient to lay the hologram samples out on a regular, rectangular grid and because of this the hologram samples are referred to as *hologram pixels*. There is no advantage in laying the image samples out on a regular grid and so these are placed in the required design area spaced out by the required sampling distance, these are referred to as *image samples*.

The effect of image sampling

This section investigates the image sampling rate. If the image is “under-sampled” then the image reconstructed from the resulting hologram will be made of discrete points of light at each sample point. If the image is “over-sampled” then computing time will be spent calculating the complex amplitude at the unnecessary sample points.

In-between the two limits of under- and over sampling is the “critical sampling” rate where the minimum number of samples required for the image to reconstruct continuously occurs. This “critical sampling” distance can be calculated for the hologram by considering the size of the point spread function for the hologram. It is reasonable to assume that the minimum width of the point spread function (psf) at the image is given by the width of a diffraction limited focused point imaged through the same aperture as the hologram. For a square hologram, uniformly illuminated, the width of the psf, d_i , (for a $sinc^2$ psf) is given by

$$d_i = \frac{\lambda F}{D_h} \tag{6.1}$$

where λ is the wavelength, F is the working distance from the hologram to the image and D_h is the hologram aperture size. The resultant image will be continuous if the distance between samples is half the width of the psf.

In the following examples the image resolution was tested using three shapes, an alpha shape, a box shape and a circle (offset from the centre). The three image shapes are shown in figure 6.1.



Plots of image sample points for alpha, box and circle

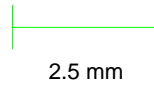


Figure 6.1: Diagram showing the shapes used to generated the image sample data. The alpha shape is an example of an arbitrary curved line with limited symmetry. The box and circle shapes are examples of shapes where simple analytical methods could be used to design the hologram phase distribution.

For this experiment the holograms were designed using a dynamic-target-based cost function with binary phase and 128×128 , $10\mu\text{m}$ square pixels. Apart from the different number of samples the design parameters were identical in all cases.

The designs where made using image sampling distances of 5, 10, 15, 20, 25, 30, 35, 50 and $100\mu\text{m}$. The sampling rate is defined as the reciprocal of the sampling distance.

For these examples the width of the psf was approximately $50\mu\text{m}$ and the estimated critical sampling distance approximately $25\mu\text{m}$. The estimated critical sampling rate is therefore 40000 samples/metre and is marked as a vertical line on the following graphs

Figure 6.2 shows the mean foreground intensity for the three shapes as the sampling rate is varied. To the left of the vertical line the image is under-sampled and the image begins to break up into dots. The intensities for the three shapes are fairly constant to the right of the vertical line indicating that over sampling the image has little effect on the solution.

Figure 6.3 shows the standard deviation of the foreground intensity. When the image is under-sampled (to the left of the vertical line) the standard deviation increases corresponding to dark patches appearing between the image samples. To the right of the line the standard deviation is fairly constant indicating that the noise cannot be further reduced by adding more image samples.

Figure 6.4 was made by averaging the information from all three shapes. The signal-to-noise ratio is made by dividing the mean foreground intensity by the standard deviation of the foreground intensity. This shows that increasing the sampling

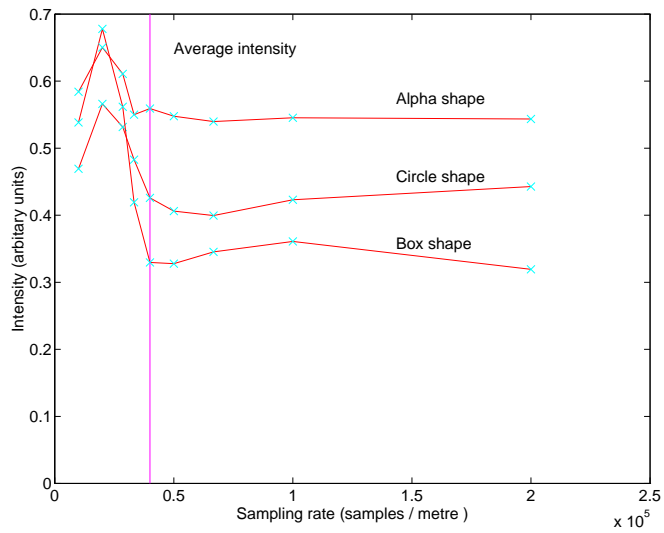


Figure 6.2: Foreground intensity vs. image sampling rate. Each shape returns different intensity levels because of the different area covered by the image sample points.

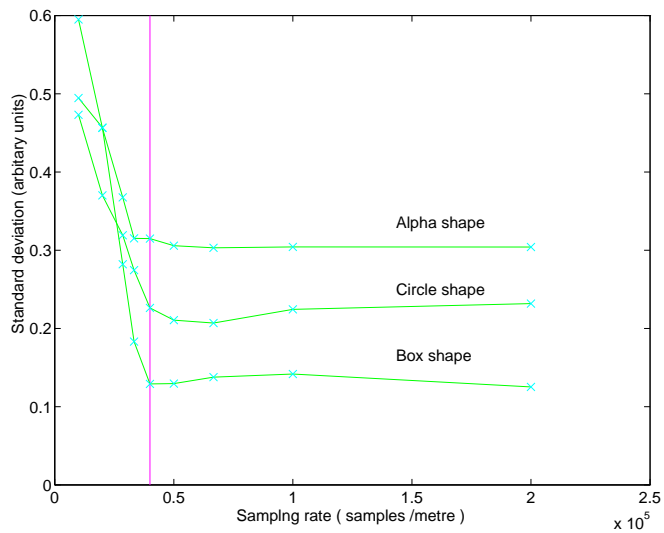


Figure 6.3: Noise vs. image sampling rate.

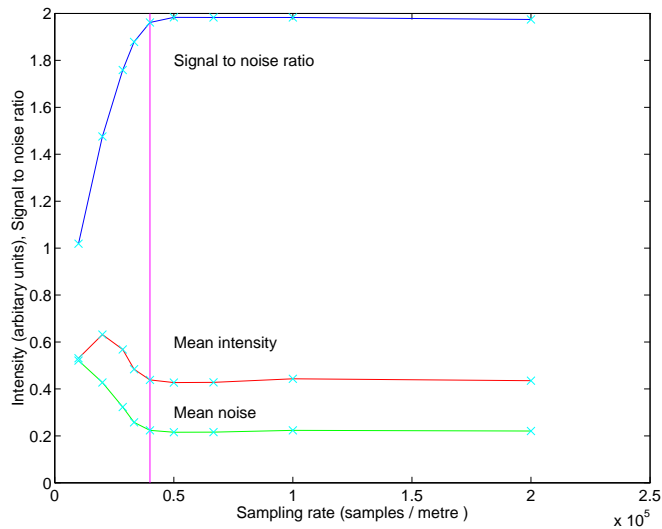


Figure 6.4: Mean intensity, noise and signal-to-noise ratio vs. image sampling rate.

rate beyond the critical rate has little or no effect.

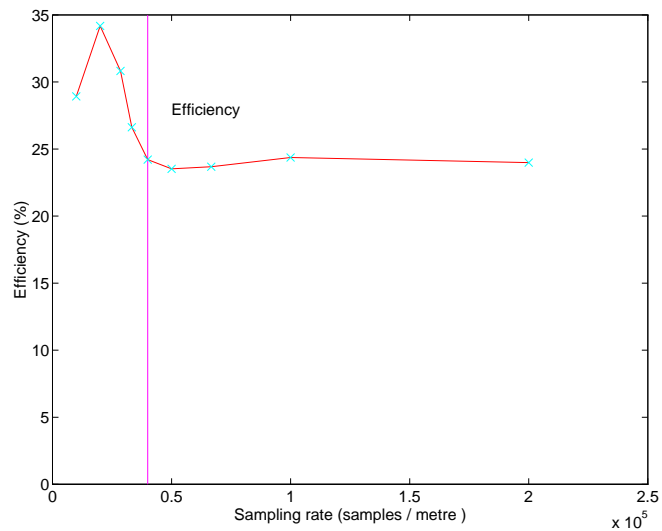


Figure 6.5: Mean efficiency vs. image sampling rate.

Figure 6.5 shows the efficiency of the holograms (again averaging over three different shapes). The efficiency initially increases as the sampling rate falls below the critical rate as it is easier to produce designs of discrete points but then falls as the foreground becomes dominated by dark patches (because of the missing samples).

Figure 6.6 shows the CPU time taken for the three shapes against the sampling rate. The number of sample points is proportional to the sampling rate (this is because the sampled images consisted of lines. If the sampled images were two-dimensional filled areas then the number of samples would be in proportion to the

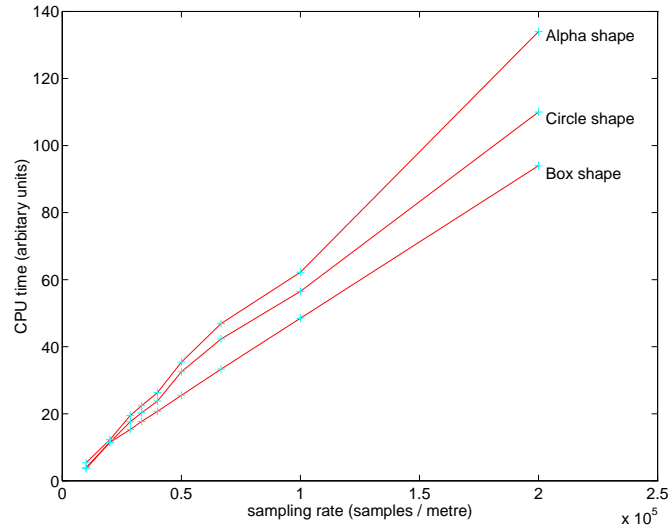


Figure 6.6: CPU time vs. image sampling rate. Each shape requires a different optimisation time because of the different number of image samples required.

sampling rate squared). This graph shows that the CPU time required is proportional to the number of image samples.

The effect of Hologram sampling

The performance¹ of a hologram and of the direct-search algorithm² is dependent upon the hologram resolution (or the number of hologram pixels). The size of the hologram (or the aperture it subtends) is set so that the minimum feature size required for the image can be achieved (see equation 6.1).

The number of pixels required across the hologram aperture is then set so that the highest spatial frequency that can be coded is sufficient to diffract the light to all parts of the image. This is the critical hologram sampling rate. This can be calculated by considering the size of the pixels, d_h , required to encode the highest spatial frequency thus

$$d_h = \frac{\lambda F}{D_i} \quad (6.2)$$

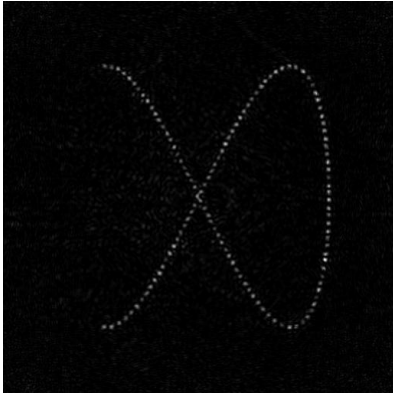
where D_i is the size of the image (assuming the image is centred on the axis of the hologram). If the number of pixels are decreased then the algorithm will not be able to diffract light across all of the image and the performance of the hologram will decrease. When the number of pixels across the hologram is increased the performance of the hologram increases but the amount of CPU time required for the optimisation rises roughly in proportion to the total number of hologram pixels.

In the following examples the image resolution was tested using the alpha shape with the same image size and image sampling for each design. The aperture that the hologram subtended was kept constant whilst the size and number of hologram pixels was varied. Designs were made using 64^2 , 96^2 , 128^2 , 192^2 and 256^2 hologram pixels. The pixels sizes were 20, 13.3, 10, 6.6 and 5 μm squared respectively. In

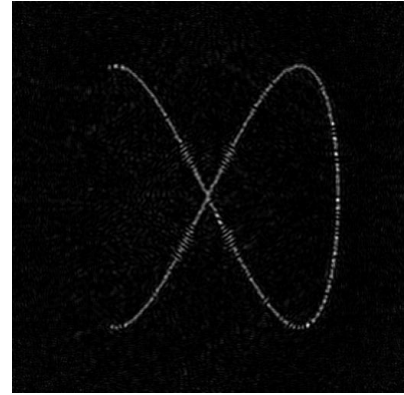
¹Indicated by the quality of the image on reconstruction

²Indicated by the quality of the image and the optimisation time

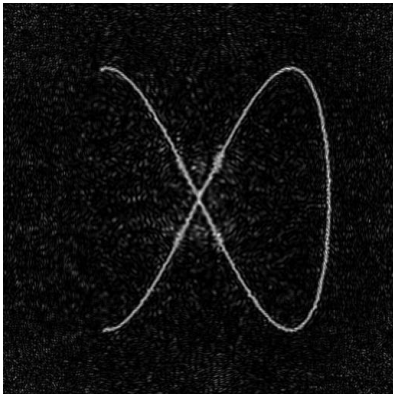
100 microns



50 microns



25 microns



10 microns

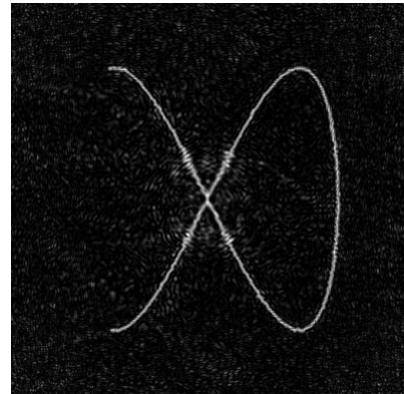


Figure 6.7: Reconstruction of designs made using different image sampling rates. The critical sampling distance is $25\mu\text{m}$.

this example the critical hologram sampling rate occurs for 128 pixels across the hologram. Figure 6.8 shows the average intensity and the standard deviation of the

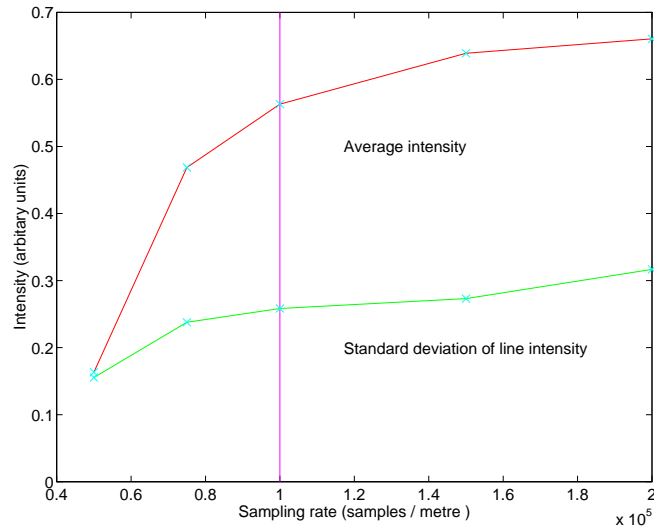


Figure 6.8: Average intensity and noise vs. hologram sampling rate.

intensity for various hologram sampling rates. The vertical line marks the calculated critical hologram sampling rate (128×128 , $10 \times 10 \mu\text{m}$ pixels). To the left of this line the hologram is under-sampled, to the right it is over-sampled. Figure 6.9 shows

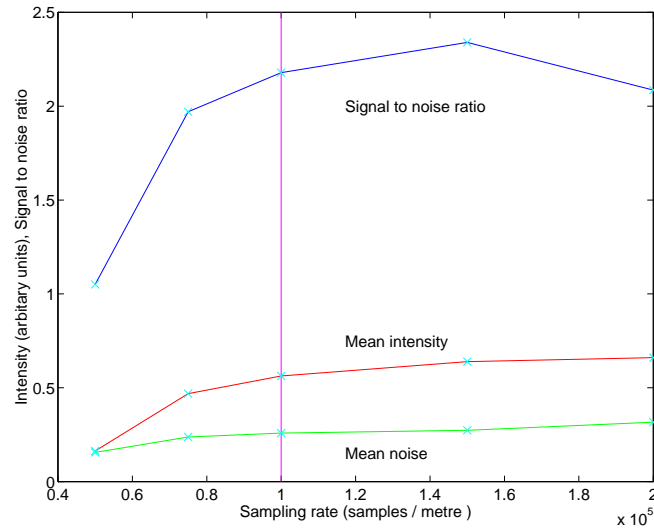


Figure 6.9: Signal-to-noise ratio vs. hologram sampling rate.

the signal-to-noise ratio for various sampling rates. Figure 6.10 shows the efficiency for various sampling rates. It is clear from figures 6.8 to 6.10 that the hologram performance rapidly deteriorates as the hologram sampling is reduced below the critical sampling rate. As the sampling rate is increased the hologram performance increases but more slowly. Figure 6.11 shows the CPU time required for these designs

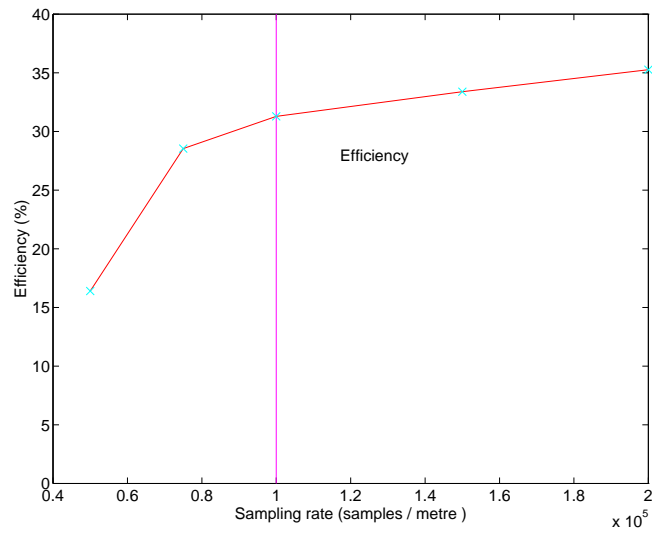


Figure 6.10: Efficiency vs. hologram sampling rate.

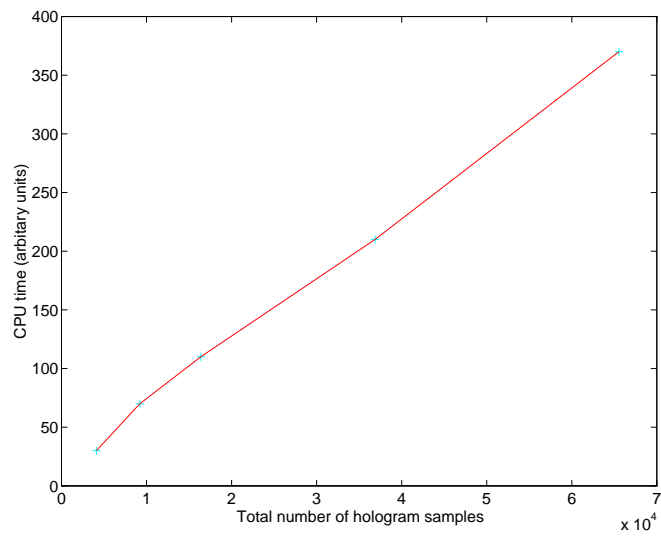


Figure 6.11: CPU time vs. total number of hologram pixels.

against the total number of hologram pixels required at each sampling rate (the total number of samples is proportional to the square of the sampling rate). This shows that the CPU time required is proportional to the total number of hologram pixels and therefore proportional to the square of the sampling rate.

6.1.2 The effect of the number of phase levels

For most practical designs the number of phase levels is fixed by the manufacturing process. It is important to know how the number of phase levels affects the performance of the algorithm. It is also useful to know the significance of the number of phase levels when considering the application and the manufacturing and design processes.

The data for the following graphs were taken from designs for the alpha shape using 2, 3, 4, 6, 8, 12 and 16 phase levels. The manufacturing process (see Chapter 3) means that holograms designed for 3, 6 and 12 levels would probably not be used in practice.

In each of the designs, all the available phase levels were available throughout the design process. The stopping parameter was varied for each phase level because of

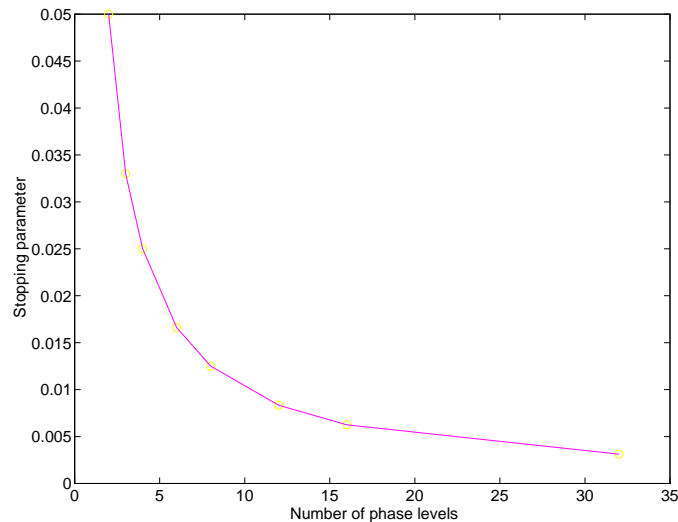


Figure 6.12: Stopping parameter used at different numbers of phase levels.

the way the number of changes accepted and rejected is monitored in the algorithm. Once the algorithm has run for some time most pixels are unlikely to change and furthermore those that are likely to change are likely to change to a “neighbouring” value (i.e., one level up or down from their current value). As a consequence it is reasonable to assume that the rate changes are accepted is inversely proportional to the number of phase levels and that the algorithm would terminate prematurely at high numbers of phase levels unless the stopping parameter was varied. An initial value for the stopping parameter was set at 5% for the binary phase design and then varied inversely proportion to the number of phase levels (as shown in figure 6.12). Figure 6.13 shows the intensity and noise vs. the number of phase levels. Figure 6.14 shows the signal-to-noise ratio vs. the number of phase levels. The slight drop in the signal-to-noise ratio with higher numbers of phase levels is thought to be due

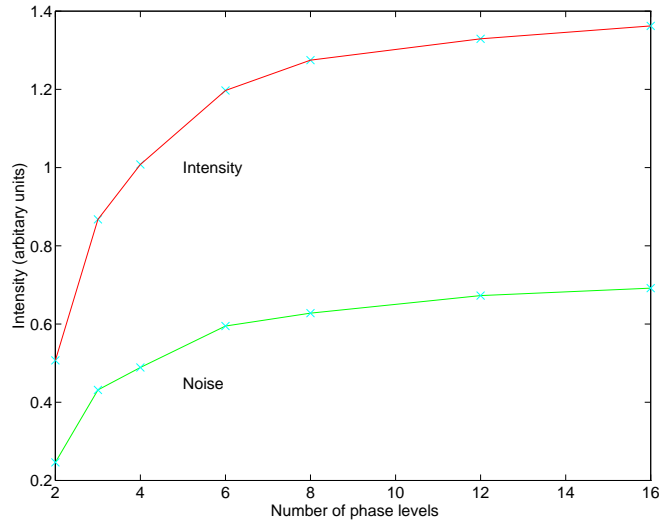


Figure 6.13: Intensity and noise for different numbers of phase levels.

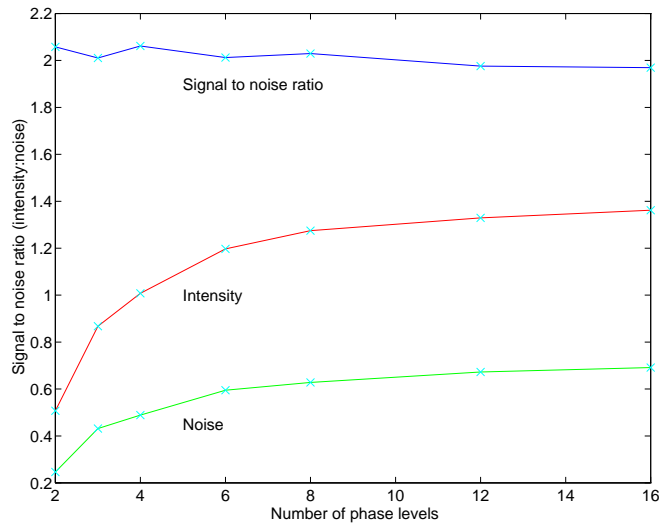


Figure 6.14: Signal-to-noise ratio for different numbers of phase levels.

to narrowing line width in the image rather than an increase in the relative noise levels³.

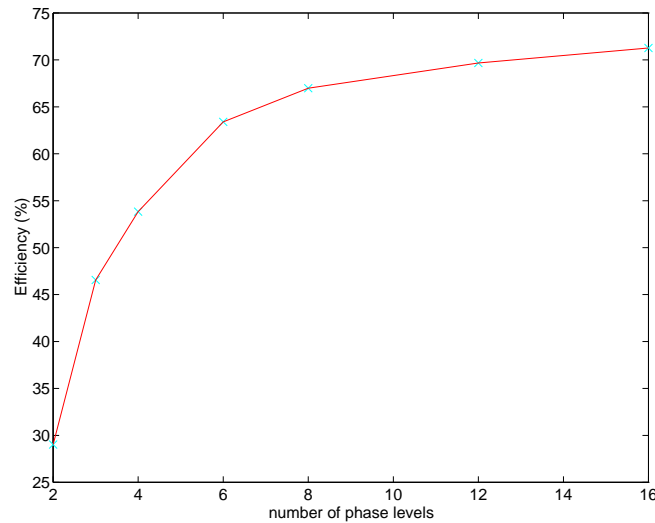


Figure 6.15: Efficiency for different numbers of phase levels.

Figure 6.15 shows the efficiency vs. the number of phase levels. The efficiency is consistent with the values reported in the literature, and with theory [30, 31].

When determining the diffraction efficiency using the method discussed in Chapter 5 the efficiency of designs made with large numbers of phase levels (>8) approaches the efficiency that would be found for a diffraction limited imaging system (limited by the same aperture as the hologram, see Section 5.4.1). This implies that no significant efficiency gains can be made by increasing the number of phase levels once the number of levels rises above eight.

Figure 6.16 shows the CPU time required by the direct-search algorithm for these designs vs. the number of phase levels. The time is roughly linear with the number of phase levels. This is expected as the number of possible changes for each pixel is proportional to the number of phase levels minus one.

6.2 Algorithm design: The effect of the starting, stopping and choosing functions

6.2.1 The effect of the starting function

The starting function is responsible for setting up the initial state of the design prior to optimisation. In order to be reliable the algorithm must be capable of optimising from an arbitrary starting point and should be largely unaffected by the initial state of the design.

³As the number of phase levels increased the line profile appeared to narrow (as it approached a sinc^2 intensity profile) because of the way in which the noise is measured (see Section 5.4.1) this may increase the measured noise slightly.

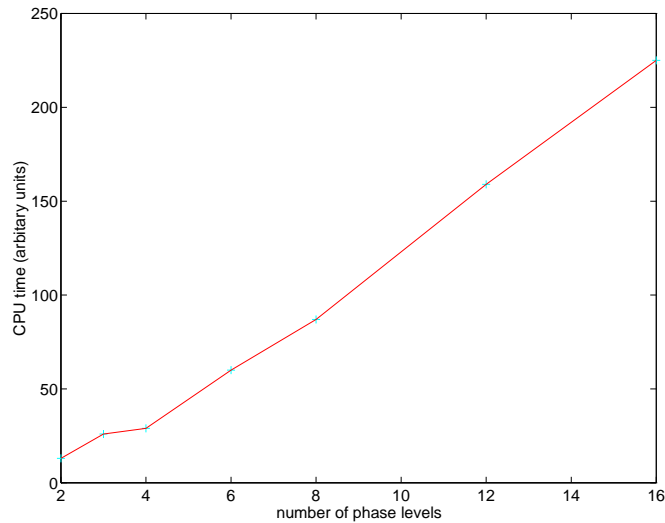


Figure 6.16: CPU time for different numbers of phase levels.

The effect of starting with “light” or “dark” hologram pixels

The direct-search algorithm can be started with either all the hologram pixels “off” or “on”. When a hologram pixel is “off” it is considered not to emit any light. As far as the mathematical model is concerned dark pixels are simply excluded from the summation given in equation 4.4. The algorithm initially tests to see if the solution is improved when a dark pixel is turned “on” to emit light with the chosen phase value. If the solution is improved then the pixel is kept “on” otherwise it remains “off”.

This means that the hologram is initially a phase and amplitude distribution. After a certain time the algorithm must be forced to make all pixels “on” so that the final solution is a phase only hologram.

If the algorithm is started with all the pixels “on” the summation in equation 4.4 must be carried out with all the pixels at their initial phase value before any changes can be made.

During optimisation the vast majority of the CPU time is spent calculating the change in the image complex amplitude due to the change in the phase or amplitude of the hologram pixel under examination. Calculating the change to the image due to turning a hologram pixel “on” is marginally faster than calculating the change to the image due to a hologram pixel phase change. For a hologram with N pixels it takes about the same time to turn “on” all the pixels prior to optimisation as it does to carry out N optimisation cycles.

Starting the optimisation with the hologram in a “dark” state (with all the hologram pixels “off”) can be thought of as allowing the optimisation to proceed during the build up of the initial solution. Starting the optimisation in a “light” state (with all the pixels in the “on” state) requires that the initial solution is built up before the optimisation can begin.

It is possible that by allowing the optimisation to choose the initial phase of the hologram pixels (i.e., starting the optimisation when all the hologram pixels are off) may increase the quality of the final solution because the noise associated with the

initial solution is kept under control. The algorithm does not inherit noise from an arbitrary initial solution which may become “locked” into the final solution.

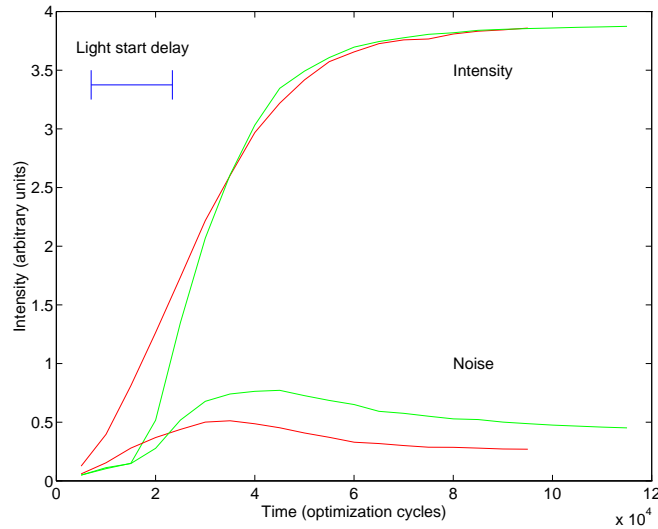


Figure 6.17: Mean foreground intensity of and noise during two optimisations. The red result starts with all the hologram pixels in the off state, the green result has to calculate the initially random solution first.

Figure 6.17 show the development of the intensity and the noise during optimisation for a “dark start” and a “light start” optimisation. The time-axis along the bottom of figure 6.17 shows the number of the optimisation cycles. For the purpose of this graph the turning “on” of each pixel in the light start optimisation (green line) is considered as an optimisation cycle. The blue marker in figure 6.17 shows a time corresponding to 128×128 optimisation cycles; i.e. the number required to turn all the hologram pixels “on”. The “light” start design lags behind the “dark” start design by approximately 128×128 optimisation cycles. This is about 20 % of the required optimisation time. The noise of the resulting “dark” start solution is lower than that of the “light” start solution. This probably occurs because the “dark” start optimisation controls the noise from an earlier stage than the “light” start optimisation.

The effect of the initial state of the design

It is possible to change certain starting conditions and parameters to generate a number of different solutions to the same design problem. These conditions are the initial conditions of the solution or optimisation algorithm as opposed to optimisation specifications such as the sampling or the algorithm conditions such as the cost parameters and pixel choice strategy.

Specifically these include the pixel at which the first choice is made and the particular sequence of pixel phase changes investigated. For optimisations where the underlying pixel phase change sequence is chosen using pseudo-random numbers the sequence can be altered by “seeding” the pseudo-random number generator with different numbers⁴

⁴Random numbers generated on computers follow a sequence which “looks” random, the par-

For very simple image intensity distributions there is likely to be a relatively small number of exceptionally good solutions⁵. With more complicated image intensity distributions the chances of such solutions existing are reduced. With simple image distributions with exceptionally good solutions, the result of repeated optimisations starting with the same design and algorithm conditions is the same form of solution⁶. However with more complicated image intensity distributions very different looking solutions may be returned but it is found that the performance of these solutions is very similar.

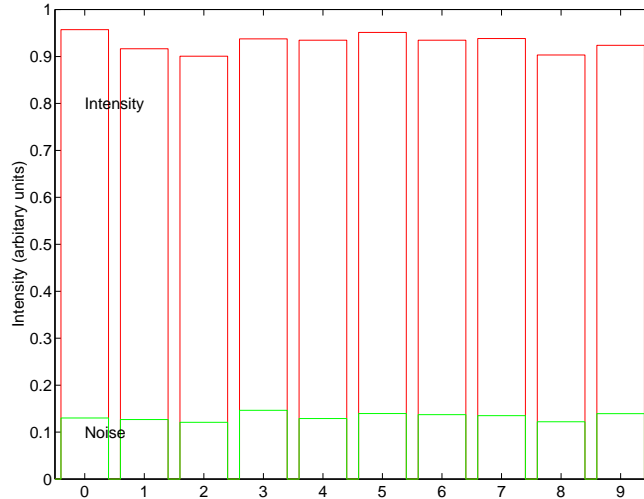


Figure 6.18: Performance of holograms optimised using the direct-search algorithm optimised using different starting conditions.

The chart in figure 6.18 shows the intensity and noise for ten holograms optimised to produce the alpha image design. Each optimisation was started with a different “seed” for the random number generator. The order in which the pixels were chosen was therefore changed between each run. As the alpha image design is not “simple” there are no “exceptional” solutions for the problem and each optimisation returned a different design.

The performance of the designs is similar, the efficiency, mean foreground intensity and mean foreground intensity noise vary by approximately $\pm 2\%$ over the different designs. The chart shown in figure 6.19 shows the CPU time required to perform direct-search optimisations for the same design using different starting conditions. The variation in the required CPU time is approximately 20% of the mean CPU time required. The CPU time required for these designs is not correlated to the design performance. If the initial solution is particularly bad it does not follow that the final solution will be bad. It does however mean that the amount of CPU time taken for the algorithm to complete may be longer.

Figure 6.20 shows the ten designs and simulations used in this section. It can be seen that whilst the designs all appear different to each other the simulated images

ticalar sequence generated may be set using a “seed” number.

⁵Usually as the result of simple symmetries.

⁶For instance if the desired image is just a single point then the result of an optimisation is always a zone plate.

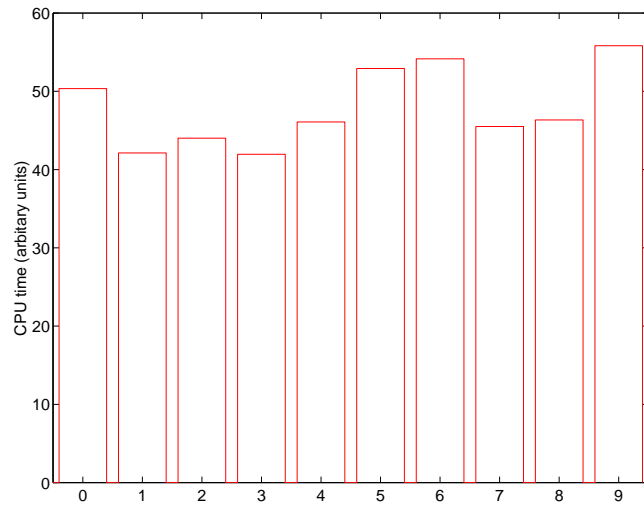


Figure 6.19: Chart showing the CPU time required to perform direct-search optimisations using different starting conditions.

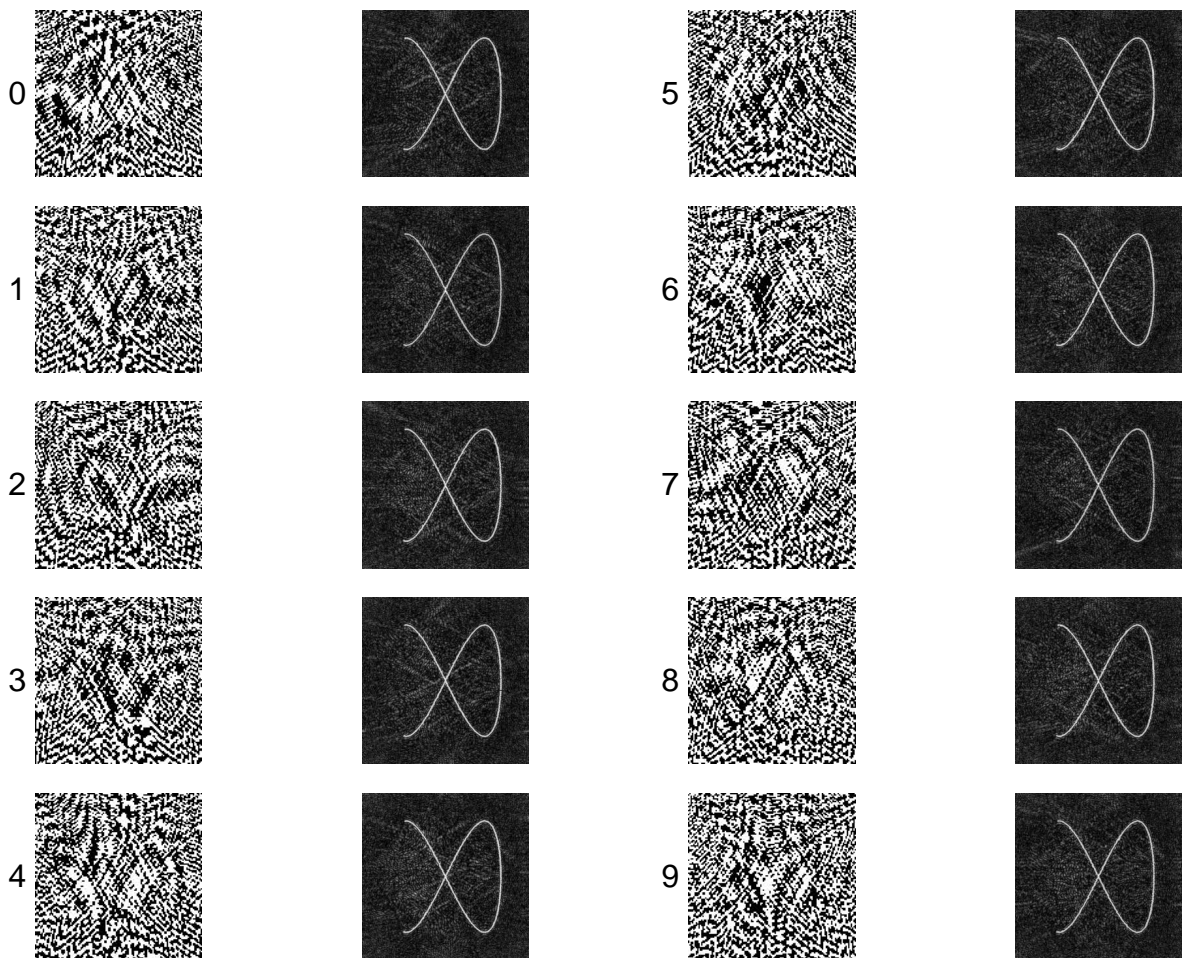


Figure 6.20: Ten holograms and reconstructions designed using different starting conditions.

are all very similar. This demonstrates the direct-search method to be reliable and capable of returning high quality solutions regardless of the initial state of the algorithm.

6.2.2 The effect of the stopping function

Stopping the direct-search method is not straight forward. As it is an optimisation algorithm there is not a well defined number of iterations required to reach an acceptable solution. The direct-search algorithm has an unpredictable run time that is dependent upon each problem and the initial starting conditions for each optimisation (see Section 6.2.1).

Three different stopping strategies have been considered although only one was investigated in detail. These are fixed-time, fixed goal and solution stagnation.

Fixed time

It is not feasible to produce a formula to predict how many iterations or changes must be considered before terminating the algorithm. This would involve many assumptions about the likely performance of the algorithm. The algorithm would be sensitive to any errors in the estimate of the number of iterations; too low and the resultant hologram would not perform well, too high and computing time would be wasted.

Fixed goal

Predicting the exact performance target for the algorithm for a particular design problem is usually impractical. If the performance target is set too low then the best performance of the design will not be reached, if the performance target is too high the design will never reach it and computing time will be wasted needlessly.

Solution stagnation

The following stopping strategy was found to be effective. The algorithm monitors the ratio of hologram pixel changes accepted to the number of hologram pixels examined. This provides an estimate of the probability, P_m , of accepting a change. The estimated probability, P_m , is monitored over a large number of trials (roughly equal to the number of hologram pixels) and updated regularly. When this measure of the probability of accepting a change falls below a pre-set stopping parameter, P_{stop} , the algorithm terminates. This approach helps to ensure that the majority of the hologram pixels are set to the optimal phase level and that time is not wasted pursuing small improvements.

When “fringe following” or pixel preselection is used pixels that are not examined because they are embedded in a fringe are counted as if they had been examined with the change being rejected. Likewise pixels that are not examined because the new phase level would be the same as the existing phase level are counted as if they had been examined and rejected.

It is possible to estimate the likely efficiency of the resultant solution for a given stopping parameter P_{stop} , by assuming that the proportion of pixels not contributing at the final solution is equal to the true probability of a randomly chosen hologram pixel changing.

In using this approach to stop the algorithm, it is possible to estimate the efficiency of the resulting design by making a three simple assumptions. The first assumption is that a pixel that contributes constructively to the image intensity distribution will not change in phase if tested and that a pixel that does not contribute constructively will change its phase if tested. The second assumption is that if the algorithm is run for an infinite time or until no new changes are accepted the efficiency of the resulting solution for the hologram phase distribution, E_∞ , is independent of the initial starting conditions (see Section 6.2.1). This infinite time solution, ϕ_∞ , has all the pixels contributing constructively⁷ and a finite time solution has a fraction of the pixels contributing to the image intensity distribution.

Thirdly it is assumed that the estimated efficiency of a given finite time solution, E_{finite} , is related to the efficiency of the “best” solution⁸, E_∞ , and the true probability of change P_c by $E_{\text{finite}} = (1 - P_c)E_\infty$ ⁹.

The algorithm monitors the measured probability of accepting a change, P_m , during the optimisation as the number of changes divided by the total number of pixels. This includes a number of “null” trials where the phase of a pixel would not be changed¹⁰. The true probability of accepting a change, P_c , does not include these “null” trials. Assuming that in a given solution there are roughly equal numbers of pixels with at each phase level the true probability P_c and the measured probability, P_m , can be related by

$$P_m \approx (1 - 1/p_l)P_c \quad (6.3)$$

where p_l is the number of phase levels. The algorithm terminates when the condition $P_{\text{stop}} > P_m$. This gives the estimated efficiency, E_{finite} for a given stopping parameter P_{stop} (assuming that the algorithm stops when $P_m = P_{\text{stop}}$) as

$$E_{\text{finite}} \approx E_\infty \left[1 - \frac{P_{\text{stop}}}{(1 - 1/p_l)} \right] \quad (6.4)$$

Figure 6.21 shows the completed design efficiency and estimated efficiency (using equation 6.4) for various stopping probabilities. The best efficiency E_∞ is taken from a run with the stopping parameter as 0.001. The measured value tends to be slightly lower than the estimated value as small fluctuations in the measured statistics cause a slight premature termination of the algorithm¹¹.

Figure 6.22 shows the intensity and noise vs. the stopping parameter. When the stopping parameter is about 0.01 or less the algorithm performs well. As the stopping parameter is increased above this level the performance of the resulting designs reduces.

Figure 6.23 shows the signal-to-noise ratio vs. the stopping parameter, the signal-to-noise ratio is almost unaffected by the stopping parameter.

Figure 6.24 shows the efficiency vs. the stopping parameter. As the stopping parameter falls below about 0.01 the efficiency slowly approaches a maximum value, about 35% in this case, and as the stopping parameter increases above 0.01 the effi-

⁷Some pixels may only contribute to a negligible degree to resultant solution and in some problems.

⁸The result of direct-search run for infinite time.

⁹This assumes that all pixels that would not change contribute equally to the solution and those that would change are not contributing to the image.

¹⁰These are not actually computed, but they are counted.

¹¹ E_∞ was taken as 35% see the graph in figure 6.24.

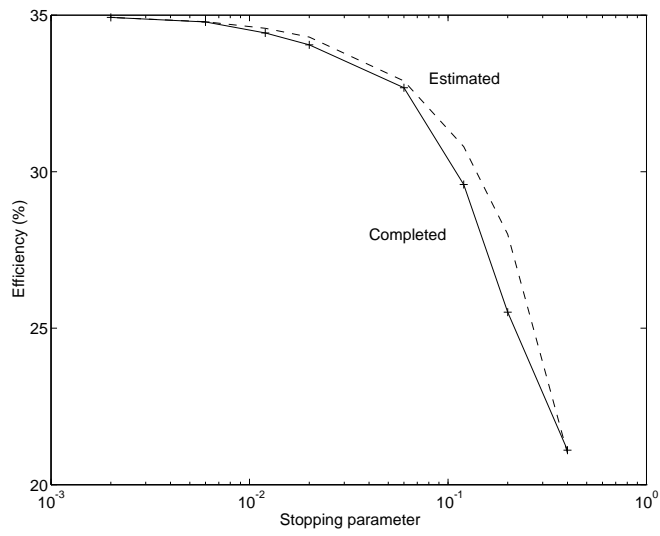


Figure 6.21: Measured final efficiency and estimated efficiency, E_{finite} , vs. stopping parameter (note log scale for P_{stop}).

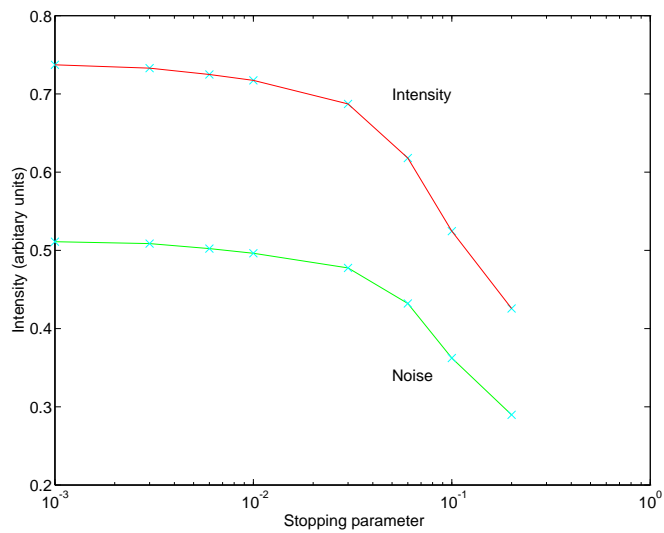


Figure 6.22: Intensity and noise vs. the stopping parameter (note log scale for P_{stop}).

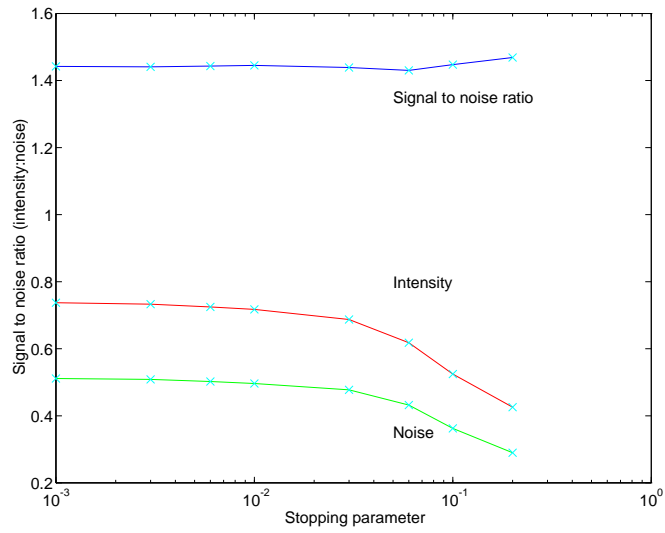


Figure 6.23: Signal-to-noise ratio vs. stopping parameter (note log scale for P_{stop}).

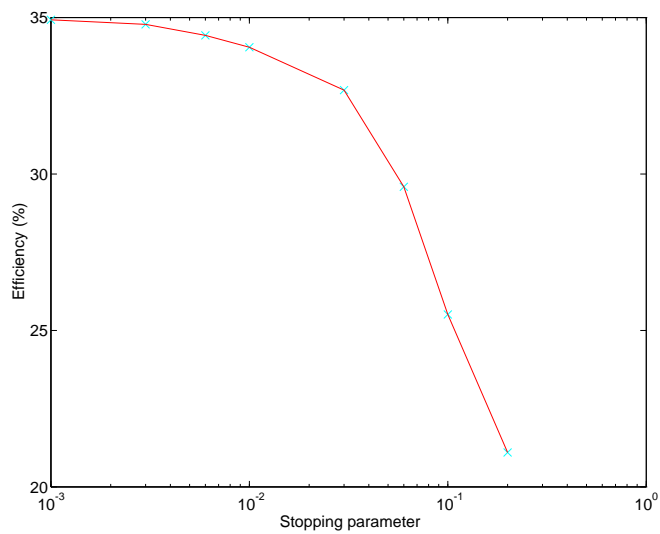


Figure 6.24: Efficiency vs. stopping parameter (note log scale for P_{stop}).

ciency drops rapidly. Figure 6.25 shows the CPU time required to run the algorithm

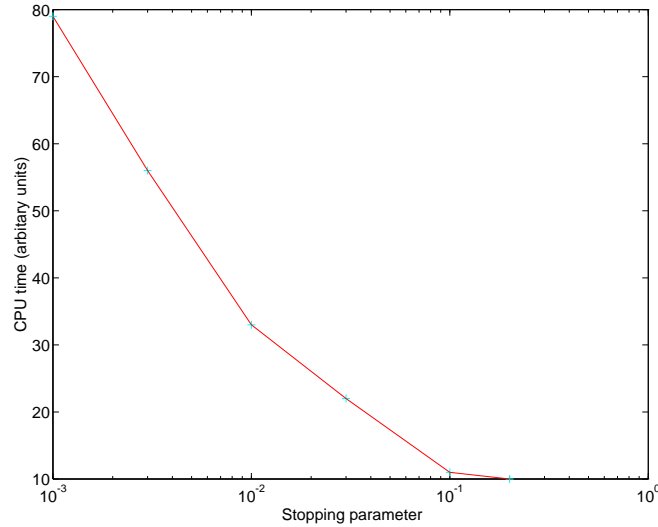


Figure 6.25: CPU time vs. stopping parameter (note log scale for P_{stop}).

vs. the stopping parameter, P_{stop} . As the stopping parameter falls the required CPU time rises sharply. This is to be expected because as the ratio of accepted changes to rejected changes decreases the rate of change of the solution must decrease. This in turn leads to the rate of change of the probability of accepting a change decreasing and the overall optimisation time increases as the stopping probability decreases.

The stopping strategy describe above contributes significantly to the reliability of the direct-search method by adjusting the optimisation time to match each problem.

6.2.3 Choosing functions

The choosing functions are responsible for selecting the trials¹² during optimisation. They can have a significant impact on the reliability of the direct-search method and can be used to enhance its computational efficiency. In this section the order in which the hologram pixels are chosen and the way in which the trial phase level is chosen is examined.

Hologram pixel choice

The order in which the hologram pixels are chosen for change influences the final solution. In an extreme case it is possible to select a combination of choices that would effectively change the “neighbourhood structure” of the solution space¹³. This could be beneficial; for instance trying to choose pixels that are more likely to change may be advantageous (see Section 6.2.3). Even when employing such a strategy the

¹²This include which pixel to change and the new phase level

¹³The neighbour structure refers to the organisation of the solutions in the solution space and how they are interconnected, it is possible to consider two different trial functions, one that places certain solutions in proximity to each other and one that places them far apart. These two schemes have the same number of solutions, but the former allows easy movement between the solutions so is less likely to trap the algorithm at a poor solution.

order in which the choices are made can affect the final solution or the time required by the algorithm.

Three basic pixels selection strategies have been investigated; random selection, random-exhaustive selection, where each pixel is visited once in a random order, and sequential selection.

The likelihood of a pixel changing is related to the length of time since it was last tested. The image is made up of many small contributions from all the hologram pixels. If the time between tests is short then the number of changes that could be made is small and the image is essentially the same¹⁴. If the pixel is re-tested before the solution has had time to change it is unlikely that its phase will change. This period during which pixels are unlikely to change upon re-examination may be thought of as a “relaxation time”¹⁵.

When random selection is used a proportion of the trials will be carried out on pixels that have recently changed (within the relaxation time) and consequently will not change. These trials can be considered wasted and unnecessary. The effect of these trials is to slow the optimisation process.

Sequential choice of pixel means that the pixels are chosen in the order that the direct-search program addresses them, the pixels start in the “top left hand corner” of the hologram and are organised in “rows”. A consequence of this is that during the early part of the optimisation, when say only half the pixels have been tested, the hologram “appears” to have a differently shaped aperture than it will have at the conclusion of the design process.

The random-exhaustive method tests each pixel once in a random order by use of a randomly arranged list of all the pixels and then using entries from this list in sequence. This means that the time between subsequent tests is kept high preventing unnecessary re-tests within the relaxation time. The image gets to “see” the full extent of the hologram aperture early on thus avoiding the problems encountered with sequential choice of pixel.

In the following examples the three basic selection procedures were tested using the alpha shape. Figure 6.26 shows the intensity and noise for the three different types of pixel selection. It shows that the method is tolerant of the pixel selection method and largely unaffected by the order of the changes.

Figure 6.27 shows the CPU time required for each of the three methods. The variation between the optimisations times is within the variation found between runs for the same repeated design (see Section 6.2.1). However these results are typical, with optimisations performed using the random pixel selection and sequential pixel selection techniques taking about 10 – 20% longer than the random exhaustive technique.

Figure 6.28 shows the intensity and noise of the image for the three different types of pixel selection throughout the optimisation process¹⁶. It is clear that choosing the pixels randomly impedes the progress of the algorithm, probably due to the re-selecting recent trials during the “relaxation” time. Choosing the pixels sequentially

¹⁴This may be contrasted with a problem such as the travelling salesman problem where single changes can change the solution by a large amount and can therefore make subsequent trials of the same change more or less independent.

¹⁵The “relaxation time” measured in hologram pixel changes was estimated at around 20% of the total number of hologram pixels for the designs performed in this thesis.

¹⁶Note that the data for the graph shown in figure 6.28 is taken from the design algorithm and only applies to the image sampling points. This gives the graphs a different appearance from those made using a full simulation of the holograms.

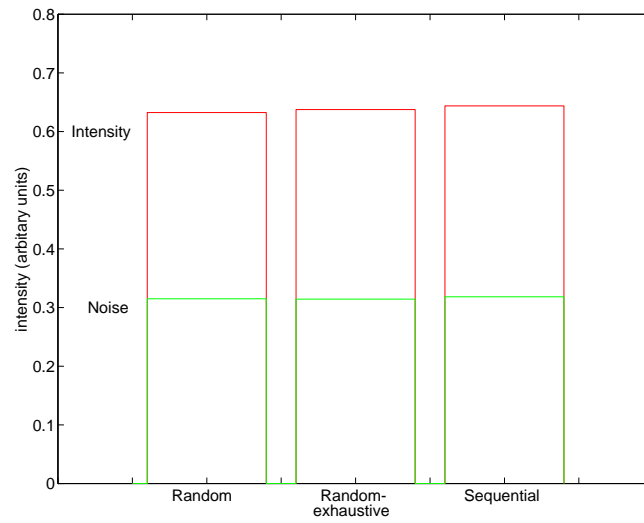


Figure 6.26: Intensity and noise for different pixel selection procedures.

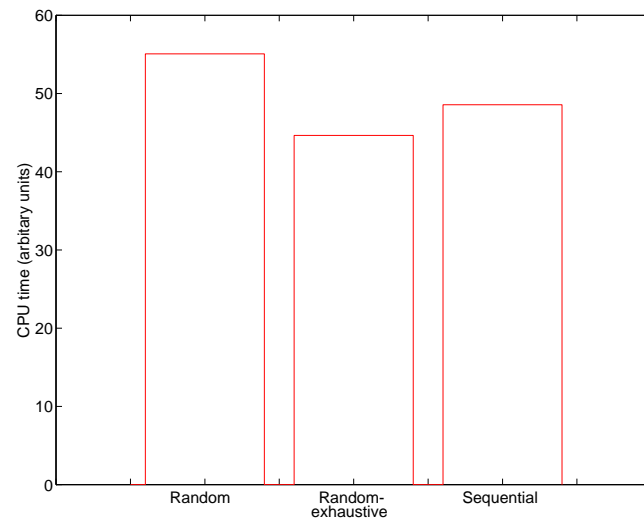


Figure 6.27: CPU time for different pixel selection procedures.

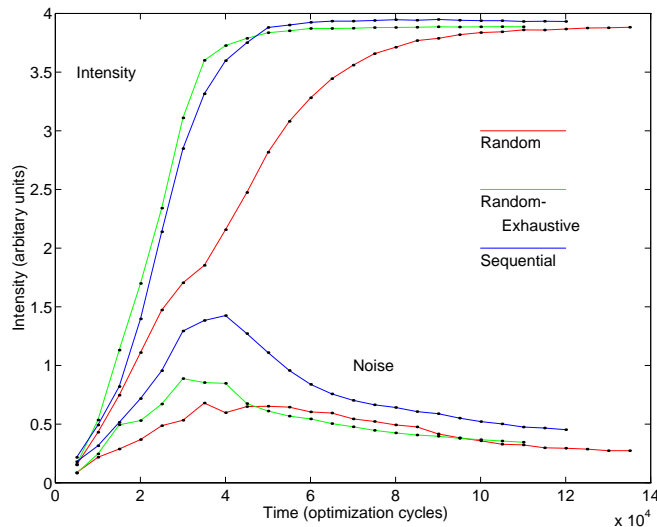


Figure 6.28: Intensity and noise during optimisation for different pixel selection procedures (note: the data for these graphs was taken from the design algorithm and applies to the image sample points only).

does not initially appear to impede the progress of the algorithm; however, as the optimisation gets past the early stages the noise increases¹⁷.

“Fringe following” (using prior knowledge to choose which hologram pixel to change)

If hologram pixels that are unlikely to change if tested can be identified quickly and easily then the direct-search algorithm can be speeded up by reducing the number of trials resulting in no change. A possible consequence of avoiding these trials may be that some important changes are blocked, “locking” the algorithm into returning a poor solution.

A method of predicting which pixels are unlikely to change and avoiding them which does not affect the final image quality has been developed. It relies on two assumptions: that the hologram phase distribution will contain “fringe” type structures and that the solution to the hologram phase distribution will evolve slowly (thus the majority of the changes will take place on fringe boundaries).

It is possible to test whether or not the fringe following strategy affects the quality of the final solution by restricting its use to different periods of the optimisation. Once the optimisation is under way and fringe structures begin to develop it is found that most of the changes accepted are on a phase boundary (i.e. at the edge of a fringe structure). At the beginning of the optimisation the position of any fringe structures move rapidly and it is more difficult to tell if the majority of changes occur on a fringe boundary.

By turning “on” the pixel preselection at different stages of optimisation and observing the quality of the final solutions it has been possible to draw the conclusion that pixel preselection has a negligible effect upon the final quality of the solution.

¹⁷Probably due of the “skewing” of the solution because of the way the hologram aperture is mis-represented at the early stages of optimisation.

The following graphs were made for hologram designs for the alpha shape using three different hologram sampling rates 128×128 , 256×256 and 512×512 pixels. The pixel sizes were $10 \mu\text{m}$, $5 \mu\text{m}$ and $2.5 \mu\text{m}$ respectively. These examples used a stopping parameter of 1%.

The designs were made for various “fringe following levels”, f_f . A fringe following level of 0 means that the preselection of pixels is carried out from the start of the optimisation process. Higher levels means that the preselection process is switched off until the optimisation process has examined every hologram pixel at each phase level f_f times. In the following examples fringe following level $f_f = 5$ means that the algorithm terminates before allowing preselection. This produces virtually the same solution as the algorithm without preselection.

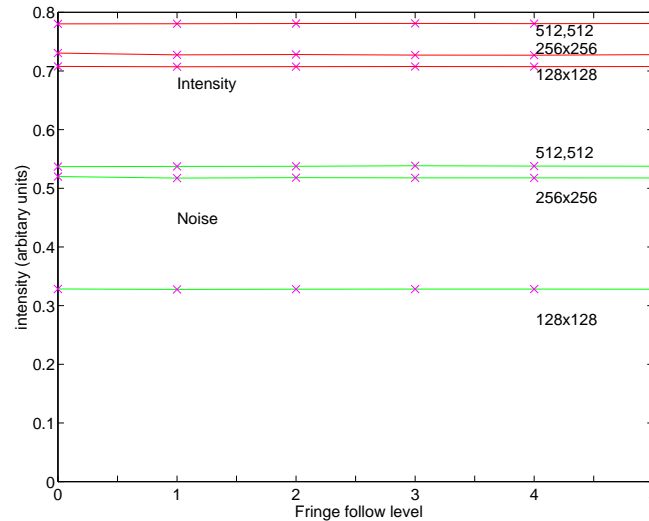


Figure 6.29: Intensity and noise for different fringe following levels.

Figure 6.29 shows the intensity and noise for different fringe following levels using three levels of hologram resolution (numbers of hologram pixels). At a fringe following level of 5 the algorithm behaves as it would without fringe following, at a fringe following level of 0 it only examines pixels on fringe boundaries throughout the optimisation.

It is clear from figure 6.29 that preselecting the pixels on the basis of whether or not the pixel is on a phase boundary does not affect the quality of the final solution.

Figure 6.30 shows the CPU time required for different fringe following levels using three different levels of hologram resolution. These show that as the fringe following level is increased (so less preselection is done) the amount of CPU time required increases. It also shows that the saving in CPU time is more significant when the hologram is sampled using a greater number of pixels. Figure 6.31 shows the CPU times shown in figure 6.30 scaled by the total number of hologram pixels used in each design. It shows that the required CPU time per hologram pixel increases as the fringe following level increases and that the saving in CPU time is more significant when the hologram is sampled using a greater number of pixels.

The “speed up” obtained will depend on the problem. An estimate of the time saved can be made by assuming that all pixels embedded in fringes will not change and that avoiding examining them takes no time.

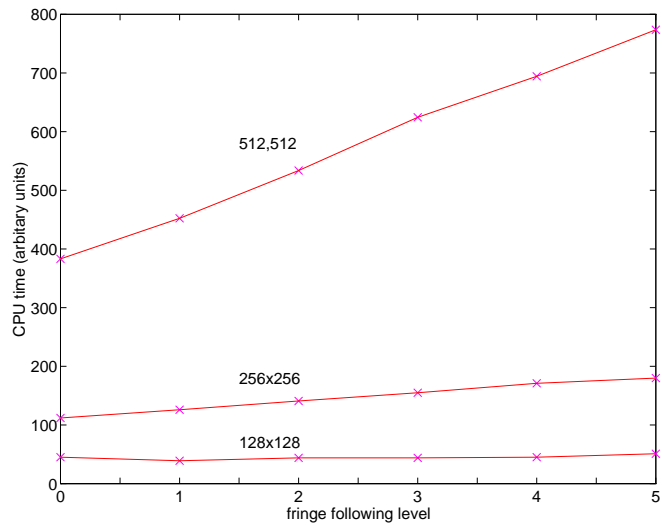


Figure 6.30: CPU time vs. fringe following level.

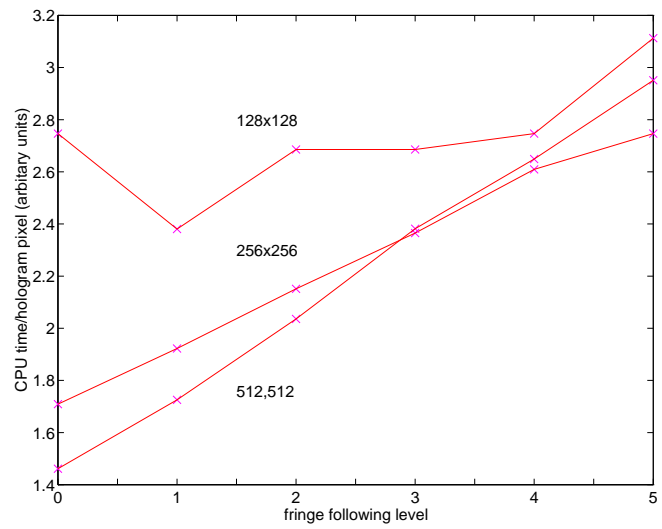


Figure 6.31: CPU time per hologram pixel for fringe following levels.

By assuming a simple model whereby the solution has two stages, stage one where little or no fringe structure is developed and stage two where fringe structure is developed, it is possible to write an expression for the time taken by the algorithm as a function of the number of hologram pixels. For an algorithm without preselecting pixels on phase boundaries the time for the algorithm to complete, T is

$$T \propto aN_{\text{total}} + bN_{\text{total}} \quad (6.5)$$

where N_{total} is the total number of pixels, a is the proportion of time spent without fringe structures and b is the proportion of time spent with fringe structures. With the preselection, the time T is given by

$$T \propto aN_{\text{total}} + bN_{\text{fringes}} \quad (6.6)$$

where N_{fringes} is the number of pixels on fringe boundaries. $N_{\text{fringes}} \propto \sqrt{N_{\text{total}}}$ because the number of pixels on fringe boundaries is proportional sampling rate and the total number of pixels is proportional to the square of the sampling rate. This implies that for a given set of design parameters the time saving will increase with the hologram sampling rate, until the optimisation time is dominated by the first stage where there is little fringe structure. This simple relationship breaks down with increasing numbers of phase levels (as the number of phase levels used increases, the number of fringe edges increase until the number of pixels on a fringe boundary is equal to the number of pixels in the hologram).

It is interesting to compare the hologram sampling experiment (Section 6.1.1, page 62) with and without pixel preselection. The experiment performed in Section 6.1.1 was repeated using pixel preselection. Figure 6.32 shows the intensity and

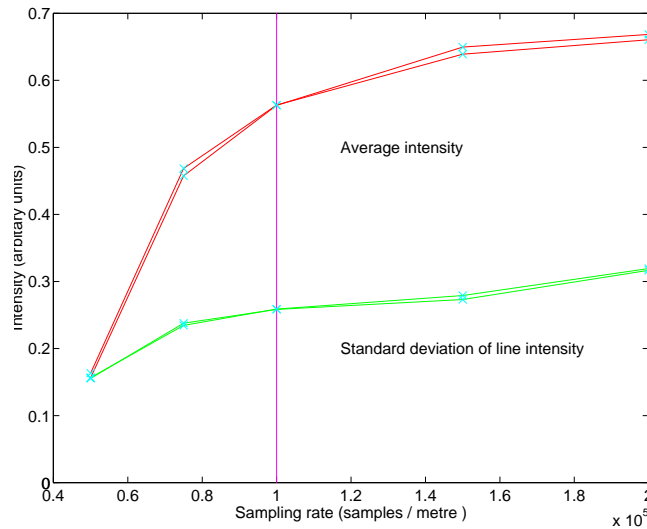


Figure 6.32: Intensity and noise vs. total number of hologram pixels.

noise vs. hologram sampling rate for designs performed with and without pixel preselection. It can be seen that the use of pixel preselection has little effect on the final results. Figure 6.33 shows the CPU time required for these designs against the total number of hologram pixels required at each sampling rate (the total number of

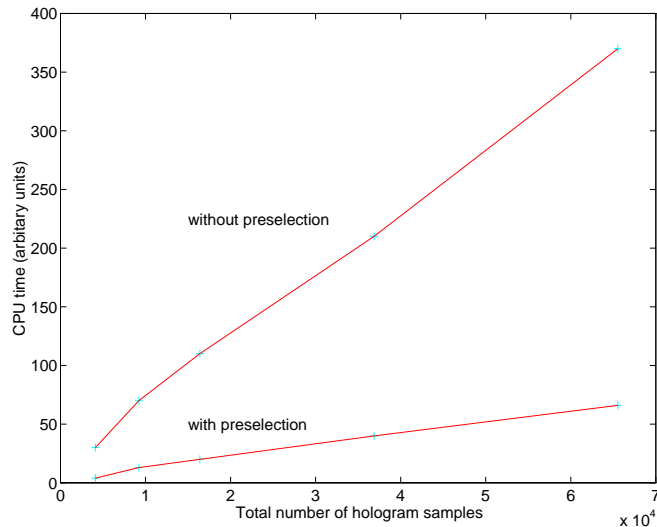


Figure 6.33: CPU time vs. total number of hologram pixels.

samples is proportional to the sampling rate squared) with and without preselection. It can be seen that the CPU time is significantly reduced when preselection is used.

Choosing the pixel phase

Each time a hologram pixel is examined the new phase level which is to be applied to it must be chosen. This new phase level must obviously be different from the existing level and must be a value that belongs to the set of quantised levels allowed in the design.

In this section five different ways of choosing the new phase level are considered. These five methods can be sub divided into two groups: where all the phase levels allowed by the set of quantised levels are available to the algorithm throughout the optimisation (unrestricted access); and where the availability of the levels is changed during the optimisation (restricted access).

For all these methods consider a set of p_l evenly spaced quantised phase levels where $p_l \in 2^n$ where $n \geq 1$ and $n \in$ positive real integers. The phase selection methods studied here could be used with numbers of phase levels that are not powers of two but the selection procedures become slightly more complex. When $n = 1$ there are only two phase levels and consequently there is no choice about which phase level to change to.

Allowing all phase levels all the time (unrestricted access).

This section deals with two methods in which all the allowed phase levels are available to the optimisation process throughout the optimisation. The two procedures investigated are “sequential” selection and “random selection” of the new phase level.

With sequential selection the algorithm tests all the pixels¹⁸ at a each phase level (bypassing those pixels already set to that phase level) and then increments the phase level to the next available level. When all the pixels have been tested at

¹⁸In the case of random pixel selection after most of the pixels have been visited see Section 6.2.3, 77.

all the available levels the process is repeated until the optimisation is stopped (see Section 6.2.2, page 73).

Random selection means that a random phase level, out of the allowed levels (but excluding the existing level), is chosen for each change.

Restricting access to allowed phase levels.

This section deal with three methods of selecting the new pixel phase that allow the optimisation algorithm to alter the number of permitted phase levels at different stages during the optimisation.

It is obviously a disadvantage for the direct-search method to finish the optimisation with fewer permitted levels than it started with. This was not tested. This contrasts with methods like *projection on constraints* (see Section 2.3.1) where there are distinct advantages in starting with a continuous phase distribution and then progressively reducing the number of permitted quantisation levels to the number of levels allowed in the final design.

It is noted from Section 6.1.2 that the image quality and efficiency rise with the number of allowed phase levels. Unfortunately the required optimisation time rises (roughly linearly) with the number of phase levels (see figure 6.16).

It would be desirable to combine the image properties of a high number of phase levels with the rapid convergence of a lower number of phase levels. Three phase selection methods which start with an initially low number of phase levels and then allow the number of phase levels to be increased later in the optimisation, up to the number of allowed phase levels, before the optimisation ends were examined. These experiments showed that restricting the number of phase levels early on actually reduces the performance¹⁹ of the direct-search method rather than improving it.

The number of phase levels that the direct-search method can use at a given moment during the optimisation will be referred to as the *permitted number* of phase levels as opposed to the *allowed number* of phase levels determined by the design. The permitted number starts initially at two and then increases during the optimisation until it reaches the allowed number.

The basic mechanism for determining when the number of permitted phase levels is increased is similar to the mechanism used to stop the algorithm (see Section 6.2.2). The probability of accepting a change, P_m is monitored and if it falls below a preset value, the *phase increment probability* P_{inc} , then permitted number of phase levels is increased. The phase increment probability, P_{inc} , must be higher than the stopping probability, P_{stop} or the algorithm will terminate before the permitted number of phase levels reaches the allowed number.

The three methods considered were “sequential phase” and “random phase” selection as described in Section 6.2.3, using the permitted levels rather than the allowed levels and “phase stepping” where the new phase is allowed to take either the next permitted level above or below the existing level (the selected level is allowed to “wrap around” to either the highest level or lowest level when the existing level is the lowest or highest respectively).

The levels are chosen so that they cover the range from $0 \rightarrow 2\pi$ as evenly as possible with the number of permitted levels, for instance when the number of allowed levels is 8 the number of permitted levels will be 2, 4, and 8. These levels will be 0 and π ; 0, $\pi/2$, π and $3\pi/2$; 0, $\pi/4$, $\pi/2$, $3\pi/4$, π , $5\pi/4$, $3\pi/2$ and $7\pi/4$

¹⁹Performance includes the quality of the final solution, the speed of convergence and the reliability of the method.

respectively.

Comparison of phase choice methods

The designs made in this section all had 128×128 , $10\mu\text{m}$ pixels and all had 16 allowed phase levels. The five different methods are listed in table 6.1. The phase increment probability, P_{inc} , was set to 0.01 and the stopping probability was set to 0.00625 (see Section 6.1.2).

The value of the phase increment probability was set, by experiment, to the lowest value that allowed the algorithm to finish optimisation with all the phase levels being used. Values of P_{inc} greater than this produced smaller effects until when $P_{\text{inc}} > 0.1$ when very little effect was observed at all (the probability of accepting a change quickly falls below 0.1 so all the phase levels are available for nearly all of the time see figure 6.38).

Name	Selection of new level	Permitted levels
sequential (all levels)	sequential	equals allowed levels
random (all levels)	random	equals allowed levels
sequential with inc.	sequential	increment when $P_m < P_{\text{stop}}$
random with inc.	random	increment when $P_m < P_{\text{stop}}$
stepping	one level up or down	increment when $P_m < P_{\text{stop}}$

Table 6.1: Summary of the five different phase selection procedures.

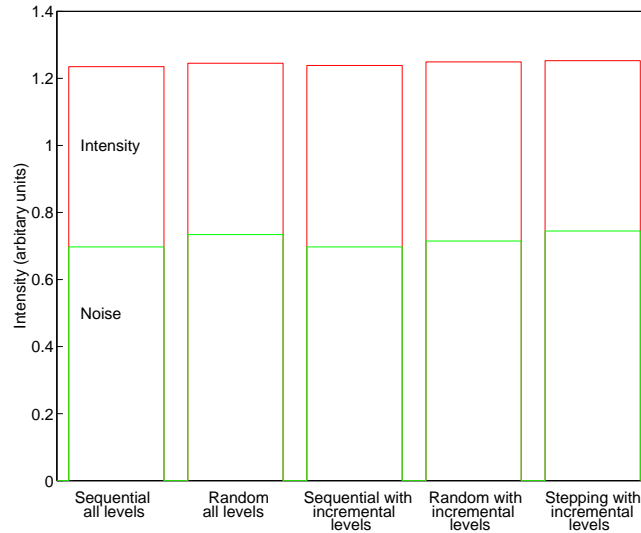


Figure 6.34: Intensity and noise for five different phase selection procedures.

Figure 6.34 shows the mean intensity and noise for optimisations performed using the five different phase selection methods shown in table 6.1. It shows small variations between the five different phase selection procedures.

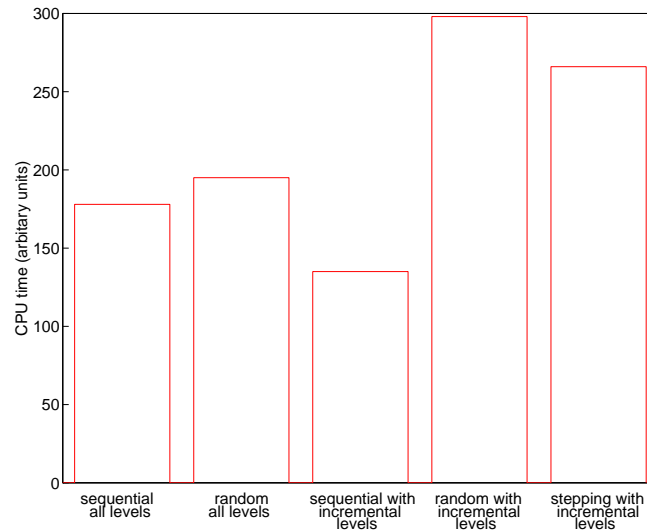


Figure 6.35: CPU time required for five different phase selection procedures.

Figure 6.35 shows the CPU time required to perform optimisations using the five different phase selection procedures shown in table 6.1. The graph shows significant variation between the required optimisation times.

The CPU times for the designs optimised using restricted access to the phase levels available required *more* time for the optimisation with the exception of the method using *sequential phase with incremental numbers of phase levels*, however, this is a special case and is discussed below in Section 6.2.3.

Figure 6.36 shows the mean intensity of the image sample points as during the optimisation for the five different phase selection methods. Note that for the three methods using incremental numbers of phase levels the intensity levels off and then rises sharply three times. These discontinuities correspond to the times when the number of phase levels was increased.

Figure 6.37 shows the standard deviation (noise) of the intensity at the image sample points as calculated by the optimisation algorithm during the optimisation for the five different phase selection methods. Note the second prominent peak in the noise for the method of sequential phase choice using all available phase levels. Figure 6.38 shows the probability of accepting a change during the optimisation for the five different phase selection methods. The stopping probability is shown by the horizontal line. It is important to note the change of horizontal axis scaling between the graphs.

The probability of accepting a change shown in figure 6.38 can sometimes fall below the stopping probability for a short period without the algorithm terminating²⁰.

Note the three sharp peaks in the probability of accepting a change for the three methods using incremental numbers of phase levels²¹. These occur immediately after the number of phase levels is increased. Also note the periodic oscillations in the probability of accepting a change in the case of sequential phase choice with

²⁰The data output from the optimisation routine has a different sampling period from the data used by the stopping function.

²¹These are masked to a degree in the case of sequential phase with incremental numbers of phase levels because of the periodic oscillation in the probability of accepting a change.

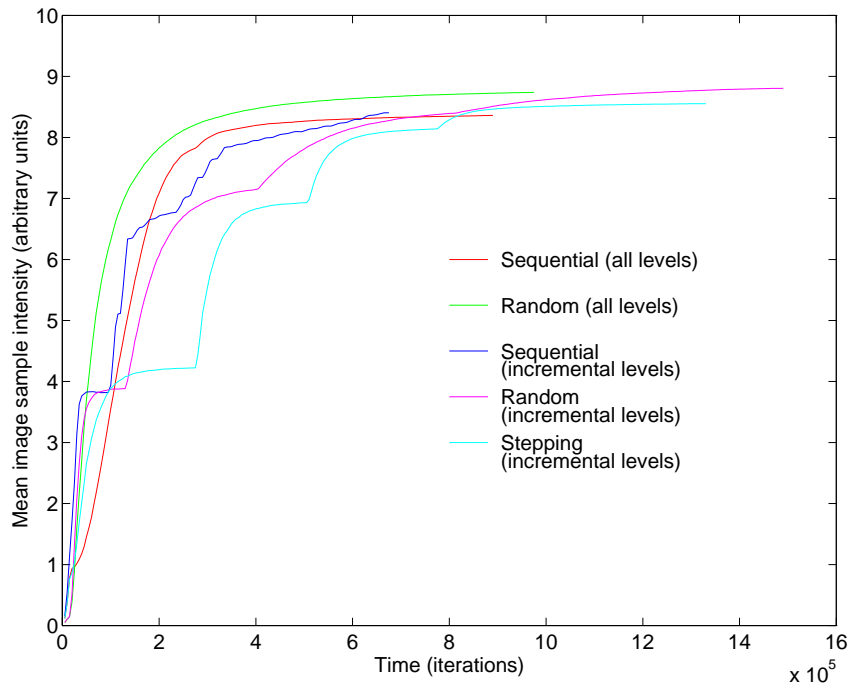


Figure 6.36: The mean image sample intensity during optimisations for five different phase selection procedures.

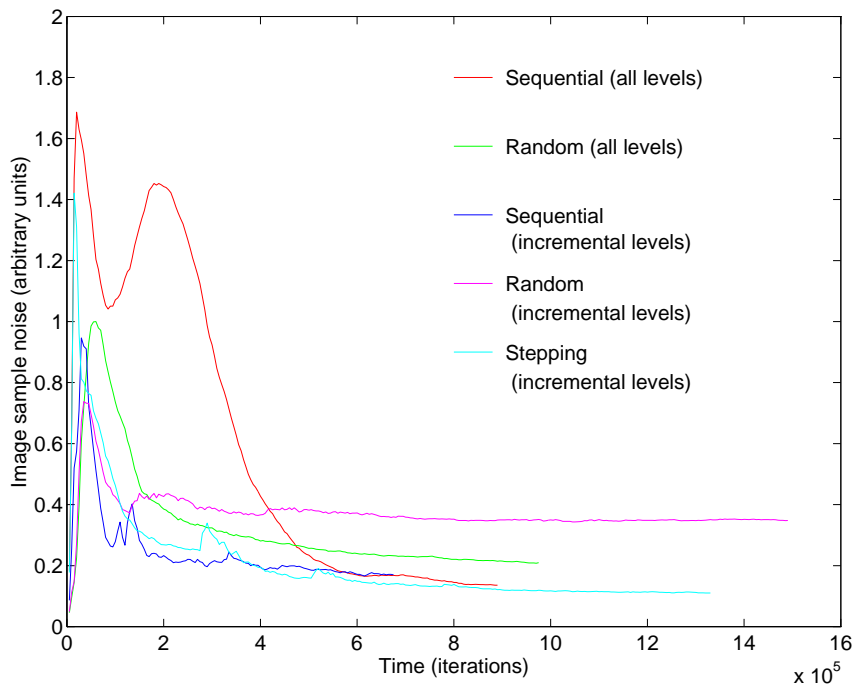
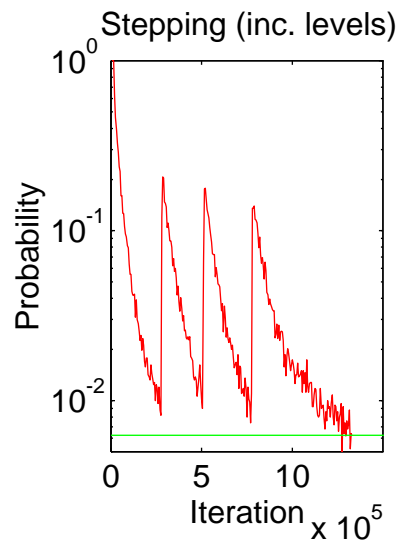
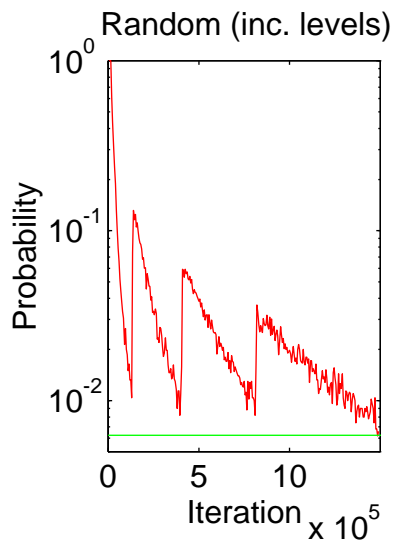
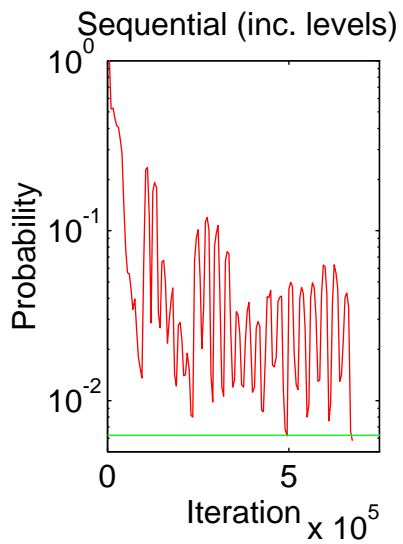
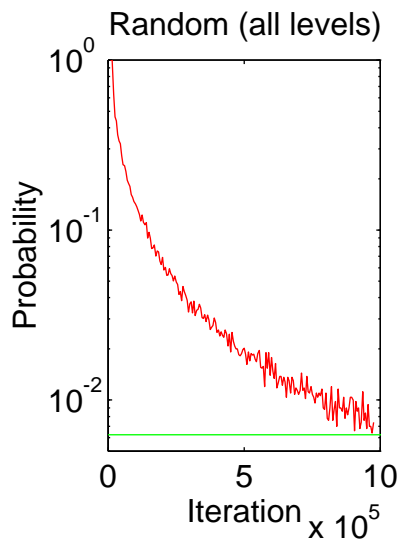
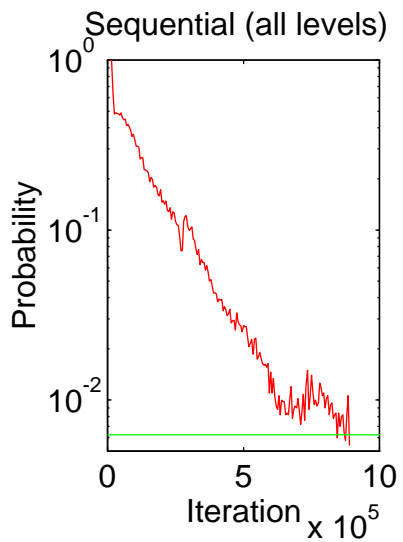


Figure 6.37: Image sample noise during optimisations for five different phase selection procedures.

Figure 6.38: The probability of accepting a change during optimisations for five different phase selection procedures.



incremental numbers of phase levels, these are discussed in Section 6.2.3.

It can be seen from figures 6.36 to 6.38 that random phase selection with all the available levels permitted throughout the optimisation produces the simplest behaviour during optimisation. The intensity (figure 6.36) rises smoothly approaching the maximum without any complex behaviour, whilst the noise (figure 6.37) initially rises and then falls off. A consequence of this is that the probability of accepting a change starts high and decays throughout the optimisation this results in predictable stopping of the direct-search algorithm (see Section 6.2.2, page 73).

The use of sequential phase selection with all the available phase levels permitted throughout the optimisation can lead to some slightly unpredictable behaviour. As the initial phase level is zero the resultant solution after all the pixels have been examined contains approximately 50% pixels with a phase of zero, this can result in some variation in the probability of accepting a change as the optimisation progresses, and this can result in slightly premature termination of the algorithm. Furthermore it can be seen from figure 6.37 that sequential choice of phase level can lead to complicated behaviour of the noise.

Methods which change the number of phase levels during the optimisation generally require longer optimisation times than methods that do not. The exception here is the method of sequential phase with incremental numbers of phase levels. This particular method causes the probability of accepting a change to fluctuate violently leading to premature and unpredictable termination of the algorithm.

Sequential phase choice with incremental numbers of phase levels

This method produces large periodic fluctuations in the probability of accepting a change.

This follows from initially only allowing phase levels 0 and π . This results in the majority of the pixels being initially set to these levels and a subsequent reduction in the probability of accepting a change whenever the new phase level is either 0 or π .

This causes the large fluctuation in the probability of accepting a change which can be seen in figure 6.38. This also causes premature termination of the algorithm. The size of the fluctuations depends upon the phase increment probability, P_{inc} . As $P_{\text{inc}} \rightarrow P_{\text{stop}}$ the size of the fluctuation in the probability of accepting a change increases and consequently the required CPU time decreases as a result of the increasing probability of premature termination of the algorithm.

In this experiment the algorithm did not terminate particularly early. The algorithm has been observed to terminate much earlier in other trials using this method of phase selection. Thus this method of choosing the new phase level makes the direct-search algorithm unreliable because of the effect it can have on the stopping function.

Summary of phase selection procedures

Restricting access to the phase levels slows down the algorithm and adversely affects its reliability because it can cause fluctuations in the probability of accepting a change.

Sequential phase choice can also cause fluctuations in the probability of accepting a change.

Thus random phase selection of any of the allowed phase levels is the preferred method of choosing the new phase level on the grounds of computational efficiency and reliability.

6.3 The effect of cost function design

The cost function is of prime importance to the direct-search method. It determines which changes are kept and which are rejected. It has a key role in discriminating between good and bad solutions and permitting progress from bad to good. It is obvious that a cost function that permits bad solutions or rejects good solutions will not allow the algorithm to perform well.

In this section a distinction is drawn between two different types of cost function; the first type being “target” based cost functions and the second type being “state variables” based cost functions.

Target-based cost functions are widely reported and commonly used in optimisation algorithms (see Section 4.7.1). They normally rely on minimising the difference between a calculated signal and a target signal. If the target is thought of as the ideal signal then the target-based cost can be thought of as an error to be minimised.

State-variables cost functions use variables to describe the state of the system (in this context the image), these are combined to give a measure of how good a particular system is (by convention the better the system is, the smaller or more negative the cost is).

6.3.1 Target-based cost functions

Target-based cost functions usually take the form

$$C = \sum_{\text{image samples}} f(I, \text{Target}) \quad (6.7)$$

where C is the cost, I is the intensity at the image sample point. The function f should have a minimum at $I = \text{Target}$ and the minimum of the cost function is therefore found when the intensity of all the sample points is the same as the target intensity at those points. This allows different relative intensities to be specified by using different target values at different image sample points.

The target-based cost functions considered in this section are typical and take the form

$$C = \sum_{\text{image samples}} (I - T_f)^2 \quad (6.8)$$

where I is the intensity at the image sample point and T_f is the target. The aim of the direct-search algorithm is to minimise the cost. The target can either be fixed-throughout the optimisation or allowed to vary. It should be noted that other target-based cost functions are possible but are rarely used in practice, these are discussed in Appendix D.

Fixed target cost functions

In the following examples, equation 6.8 was used as the cost function and the target, T , was set at the beginning of the optimisation and was not allowed to vary during

the run. A central value of the target was roughly determined by making a small number of experimental runs and noting how the algorithm performed. Additional optimisations were performed using 25, 50, 75, 100, 110, 125, 150, 200, 300, 400, 600, 1000, 1800 and 3000 % of this central value.

The alpha shape was used and the all other parameters were kept the same. A vertical line marks the position of the first estimate of the target (100%).

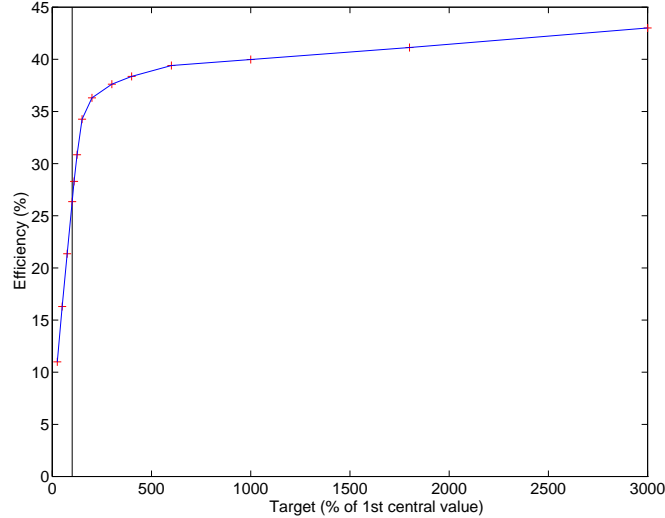


Figure 6.39: Graph showing the efficiency vs. target.

Figure 6.39 shows the efficiency of the hologram vs. the target. To the left of the graph the efficiency drops rapidly as the target intensity is far less than that that the direct search algorithm can achieve. To the right of the vertical line the efficiency climbs to about 35% after which it climbs much more slowly. Here the target intensity is significantly higher than the intensity that the algorithm can achieve for the whole image.

Figure 6.40 shows the average intensity and the standard deviation of the intensity vs. the target value. To the left of the vertical line the intensity and the noise behave in the much the same way as the efficiency, dropping almost linearly as the target approaches zero. To the right of the line the intensity and noise behave differently. The intensity rising slowly while the noise increases rapidly with increasing target values.

Figure 6.41 shows the signal-to-noise ratio vs. the target value. The signal-to-noise ratio reaches a maximum in the region of the central value and fall steeply to either side.

A target-based cost function has two desirable properties, the first is that the cost reduces as the average intensity approaches the target and the second is that the cost is reduced if the intensity of all the samples approaches the target. The first tends to lead to good efficiency and the second tends to lead to low noise. This can be contrasted with a simpler but much less useful cost function,

$$C = - \sum_{\text{image samples}} I^2 \quad (6.9)$$

which produces very poor results as the intensity distribution in the image “skews”

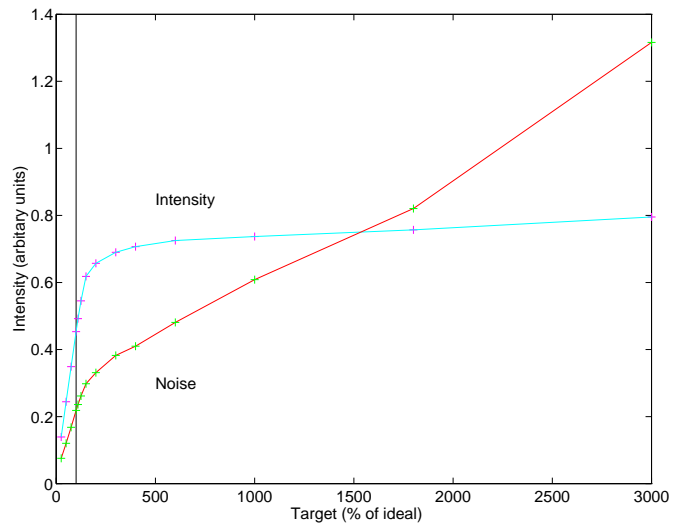


Figure 6.40: Intensity and noise vs. target.

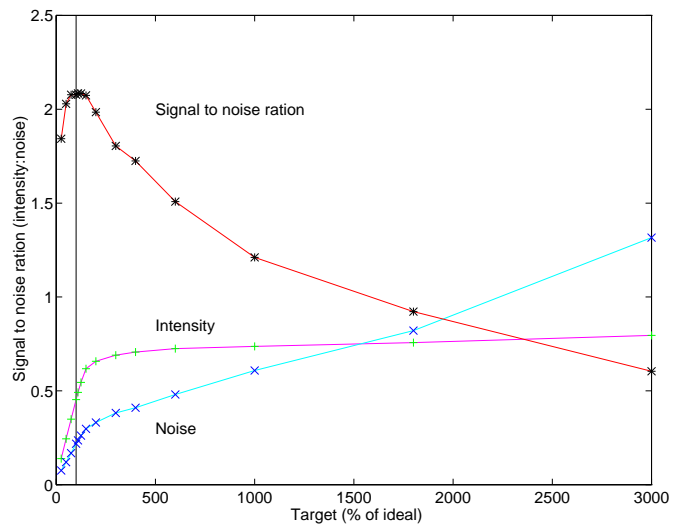


Figure 6.41: Signal-to-noise ratio vs. target.

so that a few of the samples have most of the light and the rest have very little. It is important to design the cost function so that it penalises skewed solutions.

Figures 6.40 and 6.41 indicate that there is only a small region where a fixed-target can produce good results. If the target values are too high the solution will tend to produce a “skewed” image. A target-based cost function such as equation 6.8 prevents skewing of the image intensity distribution by ensuring that an increase in the intensity of a bright sample is less significant than an improvement in the intensity of a dark sample.

Dynamic-target cost functions

Fixed target cost functions are sensitive to the choice of initial target. In many cases the only way of getting the target right is to use experimental runs of the algorithm adjusting the target between runs. This approach is not satisfactory, especially if the particular design problem is large and large amounts of computer time are required.

In order to reduce the significance of this problem dynamic-targets have been developed. These allow the algorithm to adjust the target during runtime. A simple and effective way is to use a cost function based on the fixed-target cost function 6.8 but replace the target T_f with a variable target T_v .

The variable target T_v can be changed, as the algorithm runs, provided the algorithm can recalculate the cost of the previous change using the new value of the target. Otherwise the change in the cost due to the change in the target may dominate any change in the cost due to the change in the image.

A simple rule for setting the variable target is

$$T_v = (1 + dT)\bar{I} \quad (6.10)$$

where \bar{I} is the mean intensity of the image samples and dT is the dynamic-target parameter. Obviously if the target parameter dT is too low then the algorithm will not be able to increase the intensity at each change, if the parameter is too high then the image intensity distribution will skew. An advantage of this approach is that many different types of problem require the same dynamic-target parameter so that the need for experimental runs to identify a good setting for the target is eliminated.

The following graphs show how the algorithms performance varies with the dynamic-target parameter. To the left of the left vertical line, the parameter is too low and the intensity of the image tends to be kept low whilst too the right of the right line the parameter is too high and the noise increases as the image intensity begins to skew. The graph in figure 6.42 shows the intensity, noise and signal-to-noise ratio vs. the dynamic-target parameter. Whilst the dynamic-target is kept between 105% and 200% approximately of the mean intensity the algorithm performs adequately (these points are marked with vertical lines). If the dynamic-target is too high the intensity distribution at the image can skew as the cost will be lowered if more light is diverted into a few image samples rather than shared between all of them.

Whilst the intensity and noise drop to the left of the graph the signal-to-noise ratio stays fairly constant showing that less noisy solutions tend not to be found at the expense of efficiency. This demonstrates that, unless the solution is allowed to skew, the efficiency and noise cannot be traded off against each other, however

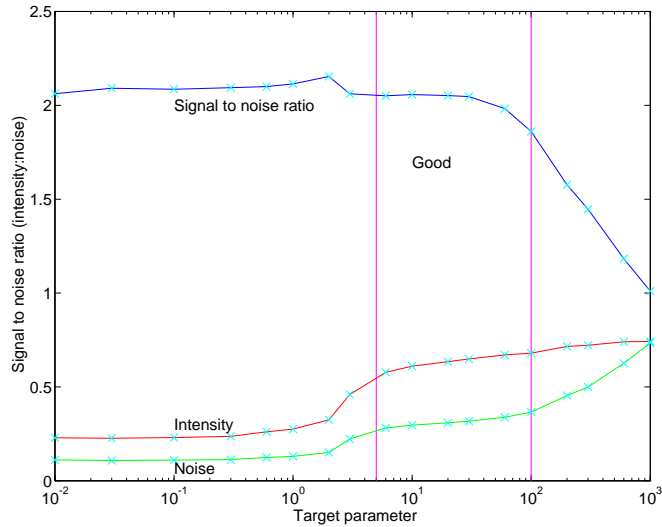


Figure 6.42: Signal-to-noise ratio vs. dynamic-target parameter.

these are dependent upon other parameters such as the number of hologram pixels and phase levels.

6.3.2 State-variables based cost functions

The principle behind the design of these cost functions is to seek to describe the state of the image intensity distribution in terms of a small number of variables. A combination of such variables may be used to describe how good the image is and therefore how good any particular solution to the hologram phase distribution is. It is important that the combination of variables used provide an accurate measure of all the aspects of the image intensity distribution that are important to the application. If not the cost function will not be able to reflect properly the true state of the image and any resultant hologram solutions will not be satisfactory.

It has been found that the state of the image intensity distribution can be adequately described using the mean intensity of the image sample points, \bar{I} , and the standard deviation of the intensity at the image sample points, σ_I .

The cost function is made up of a linear combination of these variables such that the most negative value of the cost function occurs when the state-variables are optimised. A useful form of this type of cost function is

$$C = -a\bar{I} + b\sigma_I \quad (6.11)$$

where \bar{I} is the mean intensity of the image intensity distribution, σ_I is the standard deviation of the image intensity and a and b are positive “cost balancing” parameters. This cost function would obviously give a large negative result if the intensity at all the image sample points were equal to the same large value. The first term is intended as a measure of the signal in the image while the second term is intended as a measure of the noise.

This type of cost function has three major advantages over the cost functions previously discussed. The cost function has no targets to set because as long as the

cost function gets more negative the image intensity distribution is getting better. The behaviour of the cost function is easy to predict, and in comparison with target-based cost functions is well behaved. It is also easier to combine these state-variables with other cost parameters.

It is possible to control the relative intensity of the image sample points by pre-weighting the individual image sample intensities. The weighting value for each image sample point, W_i , is given by the reciprocal of the required intensity. The mean *weighted* intensity \bar{I}_w and the standard deviation of the *weighted* intensity are then used instead of \bar{I} and σ_I to calculate the cost.

It is important when combining two or more parts in a cost function that one part does not dominate the cost preventing the other part(s) from influencing the results of trials. Experimentally, values of $a = 2$ and $3 \geq b \geq 1$ have been found to work well for most problems. If $b < 1$ the first term in equation 6.11 dominates and the resultant image intensity distribution can skew, if $b > 3$ then the second term dominates and the image intensity will be very low. It is easier to combine additional cost terms with state-variables cost functions than with target-based cost functions. This is because the value returned by a state-variables based cost function is better behaved and easier to interpret than the value returned by a target-based cost function. An important additional advantage is that the state-variables “scale” together, that is they maintain their relative importance throughout the optimisation. This helps to prevent the image intensity distribution from “skewing” at the outset of the optimisation. This amounts to allowing the algorithm to “lock out” noise from the outset of optimisation.

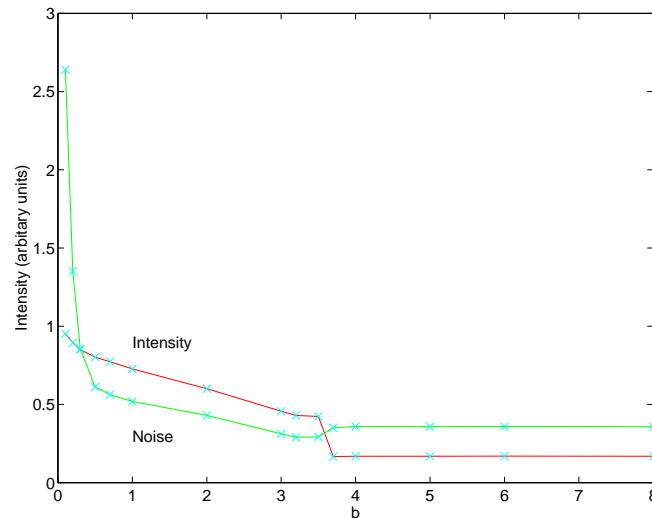


Figure 6.43: Intensity and noise vs. b ($a = 2$).

Figure 6.43 shows the mean intensity and standard deviation of the intensity in the design regions against the cost balancing factor, b (for $a = 2$). When $b < 1$ the cost function does not adequately penalise noisy solutions, resulting in noisy images, however these solutions can be more efficient²² because the first part of equation 6.11 is given a high weight.

²²In terms of the proportion of energy in the design areas.

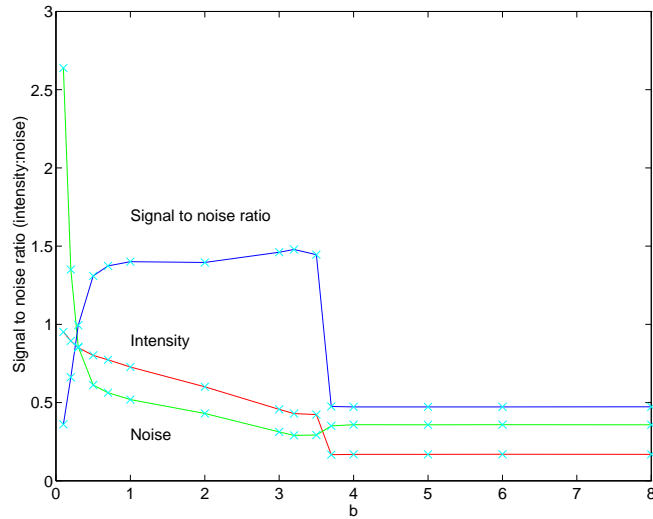


Figure 6.44: Signal-to-noise ratio vs. b .

Figure 6.44 shows the signal-to-noise ratio against the cost balancing factor b (for $a = 2$). The signal-to-noise ratio is high when $1 \geq b \geq 3$, and low elsewhere. This behaviour may seem surprising when the cost factor $b > 3$, however this is because under this condition the cost function places more emphasis upon reducing the noise. However when $b > 3$ the noise term dominates the cost function and does not allow the direct-search algorithm to increase the intensity in the design areas keeping the signal-to-noise ratio low. This is seen also in figures 6.43 and 6.44: when the cost fact $b > 3$ there is a large discontinuity in the graphs. The exact value of b at which the noise term dominates varies between design problems. Figure 6.45 shows the signal-to-noise ratio for three different design problems (using the three images shown in figure 6.1) vs. the cost balancing factor b (for $a = 2$). All three graphs show that the signal-to-noise ratio has a large discontinuity between 3 and 4.

6.3.3 Phase dislocations and cost factors

When using certain types of cost function, most notably target-based cost functions (Section 6.3.1, page 91) it is observed that many of the resulting hologram solutions produced relatively poor images upon reconstruction. A commonly observed defect is the presence of small dark patches in the image intensity distribution. These are usually approximately one image sample in size and are accompanied by a rapidly changing image phase distribution in the vicinity of the dark patch.

These dark patches or holes in the image intensity distribution are caused by *phase dislocations* [35] in the image complex amplitude distribution. These occur where there is a rapid variation in the image phase so that one side of the image defect is “out of phase” with the other. Solutions obtained using target-based cost functions are prone to developing these phase dislocations because of the uneven way the image intensity can build up. Once these have developed in a solution they are virtually impossible for the direct-search algorithm to remove. This is because changing the phase of a single hologram pixel can only move the dislocation slightly.

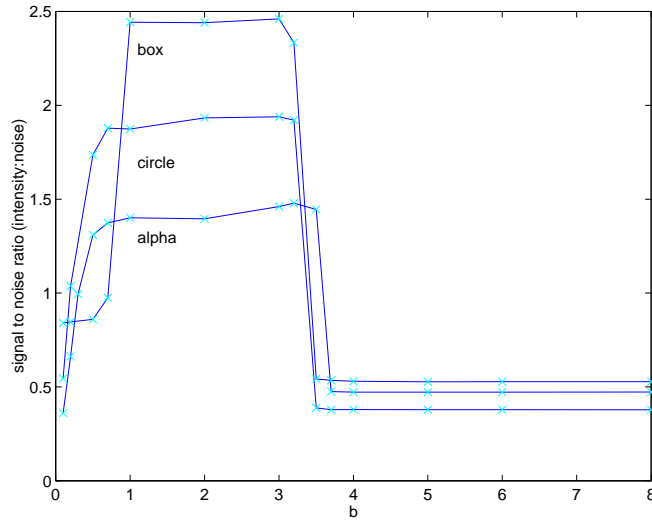


Figure 6.45: Signal-to-noise ratio for three different design problems vs. b .

It is possible for a succession of single pixel phase changes to move the dislocation out of the hologram aperture and hence out of the image intensity distribution but this is very unlikely to happen because the intermediate steps do not improve the cost. The fact that a dislocation may only be moved rather than “filled in” means that the cost will not be improved by a single pixel change. As a result of this the hole becomes “trapped” in the image and remains a feature of the final image intensity distribution.

Early attempts to control the formation of these defects were based on controlling the image phase distribution by including a cost parameter associated with either the image phase distribution or the image complex amplitude distribution in the cost function.

This approach proved exceptionally difficult with both target-based cost functions and dynamic-target-based functions.

An example of a target-based cost function with a phase restraining term is given by

$$C = a \left[\sum_{\text{image samples}} (I - T)^2 \right] + b \left[\sum_{\text{image samples}} (\delta\phi - T_\phi)^2 \right] \quad (6.12)$$

where $\delta\phi$ is the phase difference between adjacent image samples, T_ϕ is a target phase difference and a and b are positive “cost balancing” factors.

It is difficult to predict the value of the first term of equation 6.12 or the approximate change in the first term when the phase of a hologram pixel is changed. It was found that the value of this term changed significantly from design to design and even between repeated optimisations of the same design with different initial conditions (Section 6.2.1, page 70). This makes pre-setting the cost balancing factors a and b difficult.

It was found in practice that at each attempt to use this type of cost function considerable interactive intervention was required to find suitable working values of a and b . If b was set too low then the cost function behaved like equation 6.8 with the first term totally dominating. If b was too high the efficiency of the

resulting holograms was very low. The conclusion drawn was that owing to the lack of reliability the use of phase restraining cost terms with target-based cost functions is unsatisfactory for a general robust direct-search method.

The effect of constraining the phase with state-variables cost functions

Controlling the phase distribution in the image as a means of preventing image defects was also applied to the state-variables cost functions.

Control of the image phase was attempted by adding a target-based phase cost to the state-variables cost function. The relative weight of this cost was controlled by cost balancing parameter, d . The total cost function is given as,

$$C = -a\bar{I} + b\sigma_I - d \sum_{\text{sample pairs}} [(\theta_1 - \theta_2) - T_{(\theta_1, \theta_2)}]^2 \quad (6.13)$$

where θ_1 and θ_2 are the phases of the image samples in each pair and $T_{(\theta_1, \theta_2)}$ is the phase target associated with this pair. The summation is taken over all pairs of adjacent image samples. This phase term is designed to measure the amount of “phase noise” in the image, or the amount of “unnecessary” phase variation²³ between adjacent image samples. The target phase is required in this term because of the necessary curvature of the image wavefront over the image surface. This does not have the problems associated with the intensity target as it is fixed by the design conditions and optical geometry. This term is used to discourage the development of phase dislocation defects in the image.

This phase restraint term is computationally expensive. In the best case, taking about the same time to compute as the change in the image complex amplitude; in the worst case it can be many times longer²⁴. When $T_{(\theta_1, \theta_2)}$ tends to zero over the entire image it is possible to assume that $T = 0$ and use an approximate form of the phase cost term which is computationally much less intensive²⁵.

The experiment was restricted to two-dimensional images and the phase target $T_{(\theta_1, \theta_2)}$, was calculated as

$$T_{(\theta_1, \theta_2)} \approx \frac{-2\Delta r R}{F\lambda} \quad (6.14)$$

where F is the working distance, Δr is the radial separation between a pair of neighbouring image samples, R is the radial distance to the pair of image samples and λ is the wavelength. This assumes a low NA ($F \gg D_i$) hologram.

The phase restraint cost term is designed to have more significance early on in the optimisation in order to discourage the introduction of phase dislocations near the start. As the mean intensity increases the significance of this term is reduced. If it were required to have roughly the same significance throughout the optimisation,

²³This “unnecessary” phase variation may indicate the presence of a phase dislocation.

²⁴The image complex amplitude is stored as real and imaginary parts, in order to compute the phase of the image sample the arctangent of the real and imaginary parts is taken, the time taken to compute this function is about the same as the time taken to compute the change in an image sample, furthermore this function may not *vectorise* and may take 10-20× more time on average to compute (see Appendix C.1).

²⁵This approximates the square of the difference in the phase angle between the two points as $\Delta\theta^2 \approx (\Delta\Re^2 + \Delta\Im^2)/\bar{I}_{1,2}$ where $\Delta\Re$ is the difference between the real and $\Delta\Im$ is the difference between the imaginary components of the amplitudes at the two points and $\bar{I}_{1,2}$ is the mean intensity of the two points.

the cost balancing factor d , could be increased in proportion to the mean intensity during the optimisation.

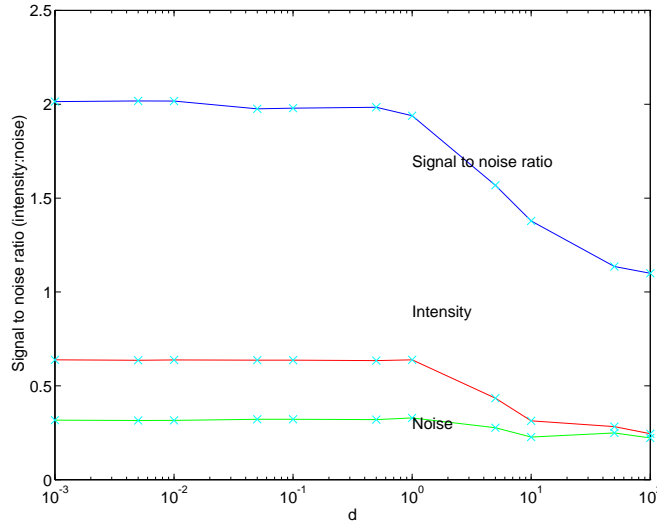


Figure 6.46: Signal-to-noise ratio vs. d .

Figure 6.46 shows the signal-to-noise ratio against the phase restraint cost balancing factor d , there is little variation until $d > 1$, after which the phase restraint term dominates and drives the signal to noise ratio down.

For $1 > d > 0$ the phase restraining term has little effect. The purpose of this term was to discourage the presence of “phase dislocations” in the image which can cause undesirable dark patches and increase the image intensity noise. These often result when using target-based cost functions.

Cost functions based on “state-variables” suppress these defects by controlling the image intensity noise throughout the optimisation. Consequently the addition of a phase restraining term has little or no effect unless it dominates the cost (see Section 6.7).

The conclusion is that a state-variables cost function can control the image noise adequately and the image quality is not improved by the use of an additional phase term in the cost.

6.4 Designing three-dimensional intensity distributions

The direct-search method is capable of designing holograms that produce three-dimensional image intensity distributions. There is no additional computational cost incurred by moving from two-dimensional to three-dimensional images. Each image sample point in a two-dimensional design problem is specified by x and y coordinates and shares the same z coordinate which is the working distance, F . The move to three-dimensions simply involves specifying separate z coordinates for each image sample point and ensuring that the image is adequately sampled in the z -direction as well as the x and y directions.

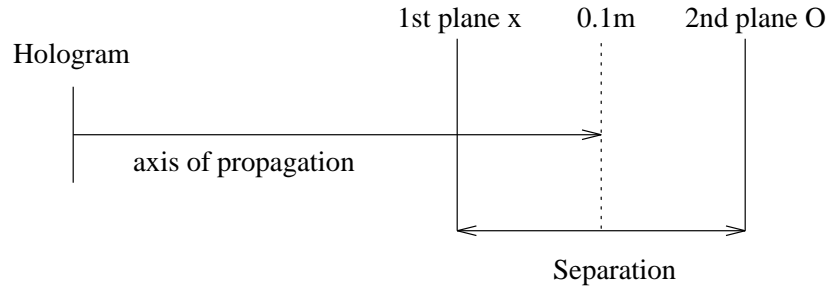


Figure 6.47: Diagram showing reconstruction configuration.

6.4.1 Three-dimensional sampling

The sampling distance in the x and y directions is set so that the distance between adjacent image samples is within half the size of the image point spread function expected for the hologram NA . The sampling condition in the z -direction is set similarly, the maximum separation in the z -direction is set to 1/2 the expected size of the image point spread function in the z -direction which is given by [67]

$$d_z \approx \frac{4\lambda F^2}{D_h^2} = \frac{\lambda}{(NA)^2} \quad (6.15)$$

6.4.2 Focal length multiplexing

It is possible to “multiplex” two or more two-dimensional images at different working distances into the same holographic element. This is particularly simple form of three-dimensional image. In this section two images were used, a cross and a circle each at its own working distance. If the working distances are the same the problem is equivalent to a planar image intensity distribution containing both the cross and the circle designs.

In the following example the working distances for the cross and the circle were initially set to 0.1 metres and a design was made, this design was then used as the zero separation flat design to be compared with the focal length multiplexed designs.

The design process was repeated five times with the working distances separated by differing amounts up to a separation of 0.05 metres, half the initial working distance. At this separation the working distance for the cross was 0.075 metres and for the circle 0.125 metres. Figure 6.47 shows the working layout.

The depth of focus for these designs, given by equation 6.15, was approximately 3.5mm. A pair of reconstructions were made for each design, at the design working distances of the cross and the circle. For comparison the “flat” or zero separation design was reconstructed at each of these distances.

Figure 6.48 shows the reconstruction pairs for the flat and three-dimensional designs at each of the separations. It can be seen from figure 6.48 that the flat design is well focused when the reconstruction separation is zero and that the images lose focus quickly as the planes are separated. This performance would not be suitable for many applications. The three-dimensional designs produce well focused images at the appropriate reconstruction planes and would be adequate for applications requiring more than one working distance. The results for the cross are indicated with “x” and the results for the circle indicated with “o”.

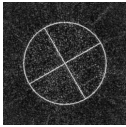
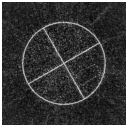
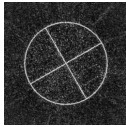
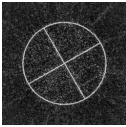
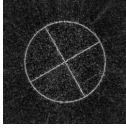
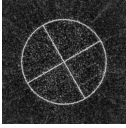
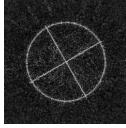
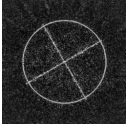

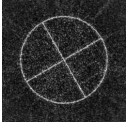
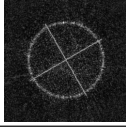
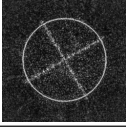
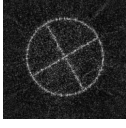
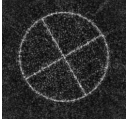
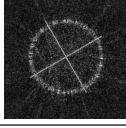
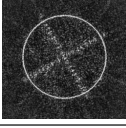
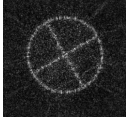
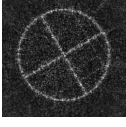
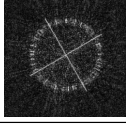
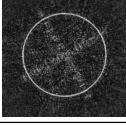
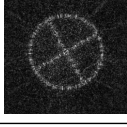
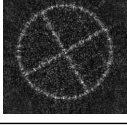
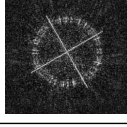
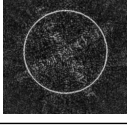
separation	1st Plane	2nd Plane	1st Plane	2nd Plane
0mm				
5mm				
10mm				
15mm				
20mm				
25mm				

Figure 6.48: Reconstructions of a 2-d design at various planes (left side) and of tailored 3-d designs at the same planes (right side).

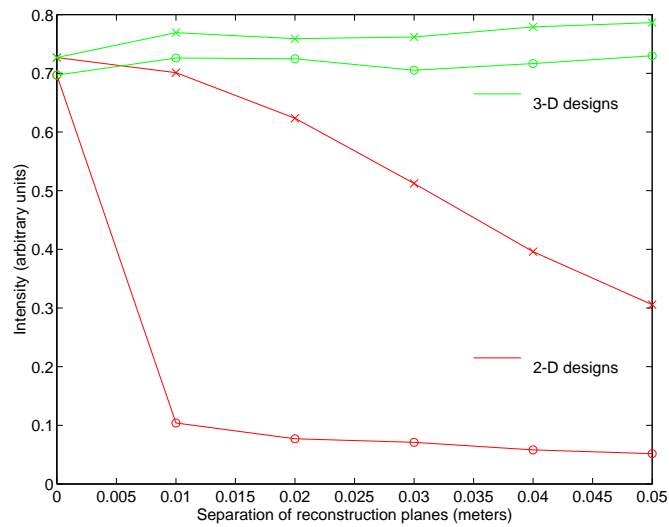


Figure 6.49: Mean intensity vs. separation of reconstruction planes.

Figure 6.49 shows the mean intensity for the cross and the circle reconstructed in their appropriate planes for both the two- and three-dimensional designs. It can be seen that the mean intensities of the cross and the circle reconstructed from the three-dimensional designs remain fairly constant as the reconstruction planes are separated. The mean intensities for the cross and the circle reconstructed from the flat design diminish as the planes are separated.

The mean intensity of the circle drops very quickly as the planes are separated because the circle reconstructed from the two dimensional design increases in radius as the planes are separated, consequently the energy falls outside of the design region (see Section 5.4.1).

The mean intensity of the cross falls slowly as the reconstruction planes are separated because the “arms” of the cross are radial and the majority of the energy still falls within the design region when the reconstruction distance is reduced.

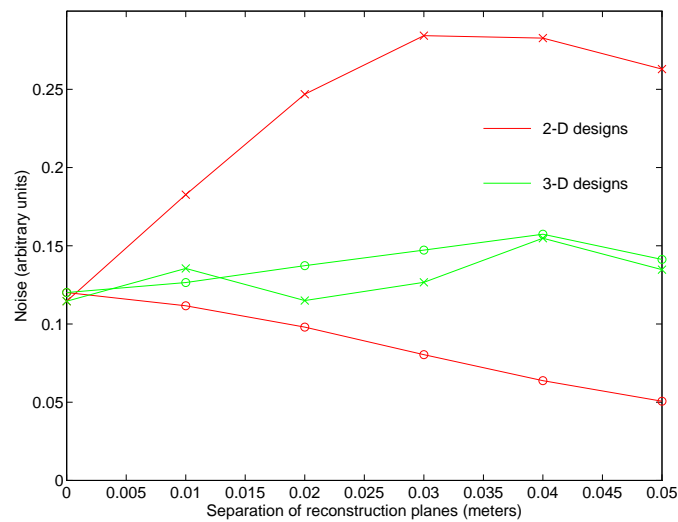


Figure 6.50: Noise vs. separation of reconstruction planes.

Figure 6.50 shows the image noise for the cross and the circle reconstructed in their appropriate planes for both the two- and three-dimensional designs. The noise levels for the cross and circle reconstructed from the three-dimensional designs remain fairly constant as the reconstruction planes are separated. The noise level rises rapidly for the cross reconstructed from the two-dimensional design as the planes separate. The noise level for the circle reconstructed from the two-dimensional design diminishes slowly as the reconstruction planes separate, this however is a result of the rapid decrease in the mean intensity of the circle as the circle moves outside of the design region (see figure 6.49 and 6.51).

Figure 6.51 shows the signal-to-noise ratio for the cross and the circle reconstructed in their appropriate planes for both the two-dimensional and the three-dimensional designs. The signal-to-noise ratio for the two-dimensional design decreases rapidly as the planes are separated. The signal-to-noise ratio for the three-dimensional designs remains fairly constant as the planes are separated.

Figure 6.52 shows the efficiency for the cross and the circle reconstructed in their appropriate planes for both the two-dimensional and three-dimensional designs. This graph also shows the combined efficiency for the three-dimensional designs and

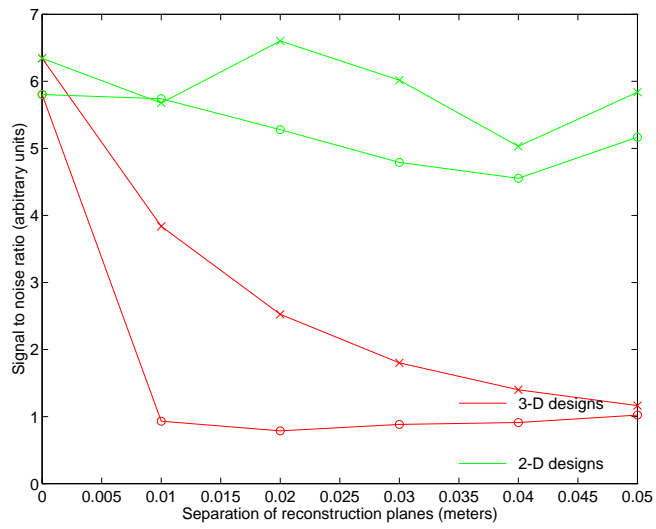


Figure 6.51: Signal-to-noise ratio vs. separation of reconstruction planes.

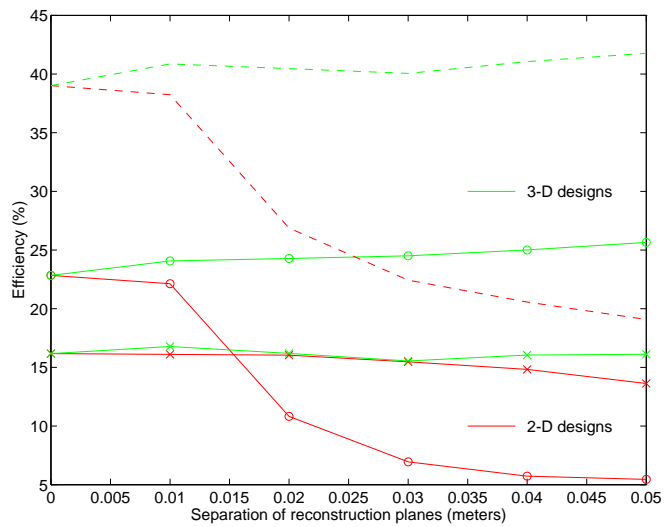


Figure 6.52: Efficiency vs. separation of reconstruction planes.

the two-dimensional design. These were calculated by adding the efficiencies for the reconstructions of both the cross and the circle.

The efficiencies for the cross and the circle differ in the zero separation design and in the three-dimensional design because the design algorithm was intended to make the intensities of the cross and the circle equal. As the foreground area covered by the circle was greater than that of the cross the proportion of the energy diverted into the circle is greater. It is also noted that as the reconstruction planes are separated the hologram NA becomes greater for the cross as it is closer to the hologram and the becomes smaller for the circle. This means the expected line width and hence the area covered by the image falls for the cross and rises for the circle. This is shown as a slight rise in the efficiency for the circle and a slight fall for the cross reconstructed from the three-dimensional designs as the reconstruction planes are separated.

The combined efficiencies (dotted lines) are higher than seen for previous designs because the “overlap” of the circle and cross image design regions results is a small amount of the energy being counted twice (once for the cross and once for the circle). The overall combined efficiency for the three-dimensional designs remains high. It drops with increasing separation of the planes for the two-dimensional design.

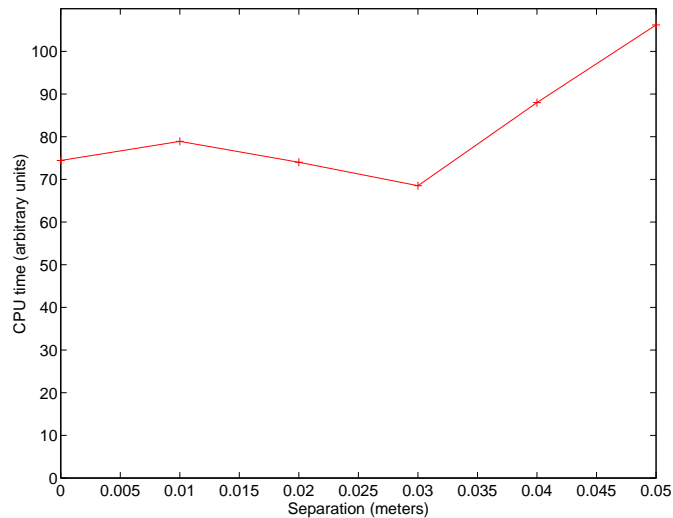


Figure 6.53: Optimisation time vs. separation of reconstruction planes.

Figure 6.53 shows the CPU time required to perform these optimisations vs. the design plane separation. The graph does not show any significant trend outside of the variation expected in the optimisation times (see Section 6.2.1). This slight variation is to be expected as at each separation the algorithm is set a slightly different problem.

6.4.3 True three-dimensional designs

True three-dimensional image intensity distributions require the working distance to be specified for each image sample. In the following example the alpha shape used previously was designed to be focused over a curved surface. The chosen surface

was a paraboloid of revolution. The working distance for each image sample, F_i , was calculated as

$$F_i = 0.1 + 2500 \times R^2 \text{ metres} \quad (6.16)$$

where R is the radial distance from the axis to the image sample. The required depth of the image or difference between the working distances of the image sample nearest to the hologram and the image sample furthest from the hologram was 0.0115m. The holograms designed in this section had 256×256 $10\mu\text{m}$ pixels. The hologram aperture used was twice the size of the apertures used in the previous examples and was chosen to increase the holograms NA and therefore decrease the hologram depth of focus. With this hologram NA the depth of the image was about 8 times the depth of focus.

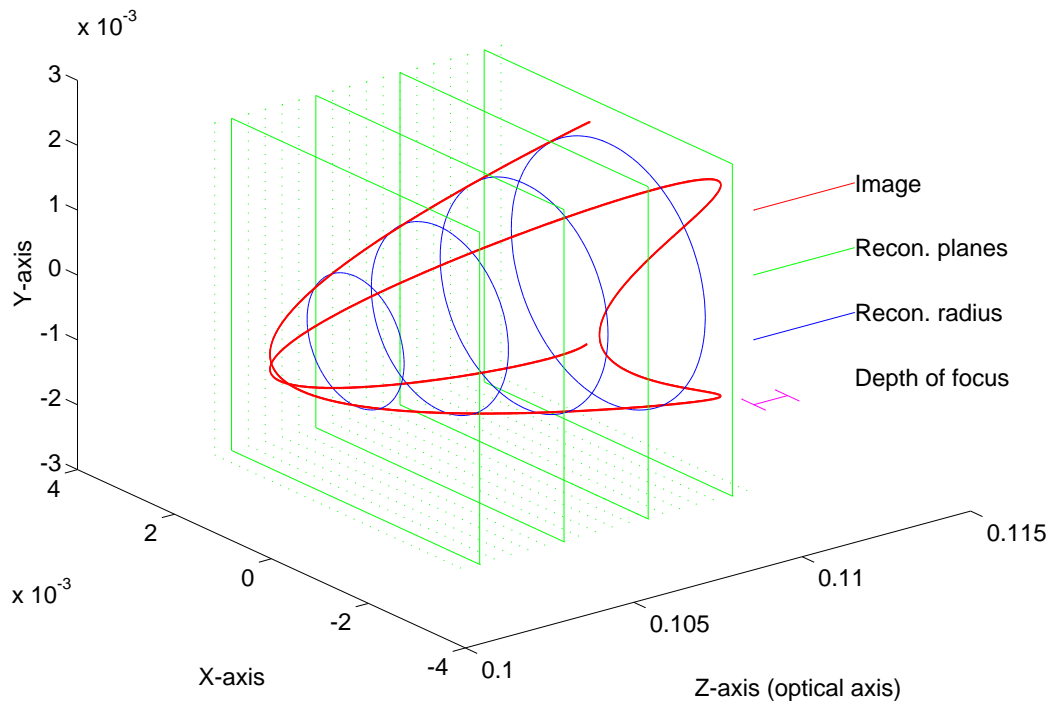


Figure 6.54: Diagram showing the image samples (red), the reconstruction planes (green) and (some) reconstruction radii (blue).

The optimisation was repeated using two-dimensional image sample data with a working distance, $F = 0.1\text{m}$ to provide comparison with the three-dimensional design.

Figure 6.54 shows the image volume. The image samples are shown in red, the position of the chosen reconstruction planes shown in solid and dotted green. The reconstruction over the paraboloid was assembled using data from these reconstruction planes. The radius of the paraboloid specified in equation 6.16 is shown in blue. The depth of focus is shown as a blue marker and is considerably smaller than the depth of the image.

The resulting hologram designs were then reconstructed over the entire image

design volume ²⁶. The sample spacing along the optical axis was 500microns. This sample spacing was significantly less than the depth of focus for the hologram and was required give a continuous reconstruction of the image over the parabolic surface because of the radial movement of the image rather than defocus.

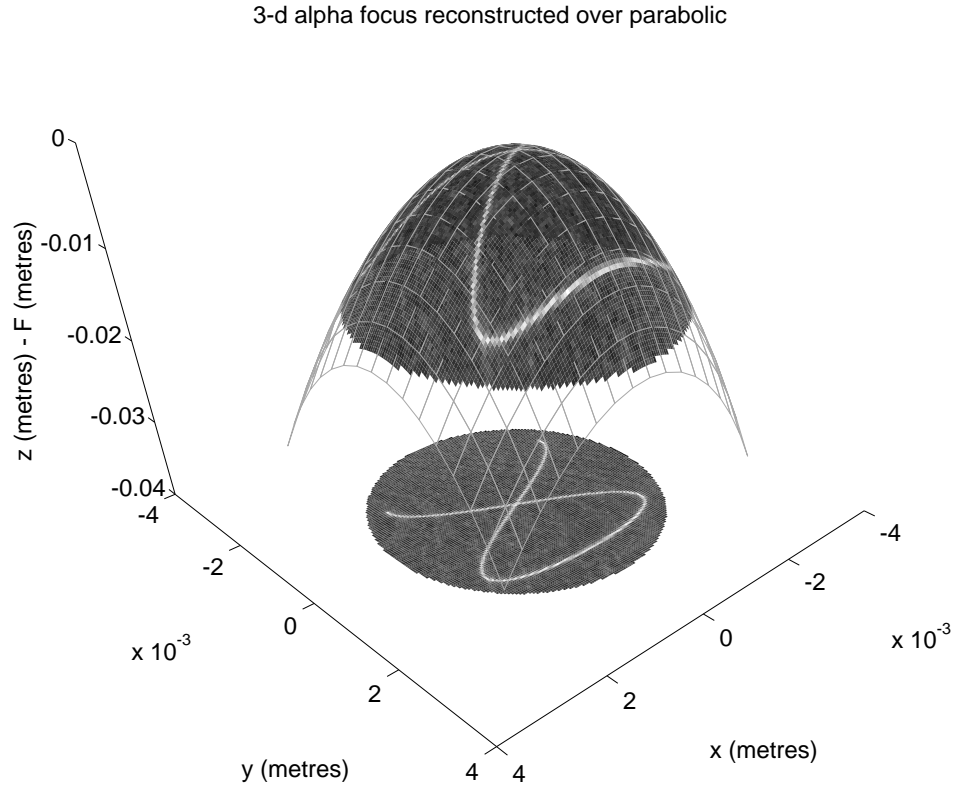


Figure 6.55: A 3d design reconstructed over its design surface.

The reconstruction of the three-dimensional design over the parabolic surface is shown in figure 6.55. For clarity a projection of the reconstruction over the surface is shown on a flat plane under the paraboloid.

For comparison a reconstruction of a flat design over the surface is shown in figure 6.56. It is clear from figure 6.55 that the image reconstructed from the three-dimensional design stays in focus over the paraboloid despite the depth of the image exceeding the depth of focus. Figure 6.56 shows that the two-dimensional image designed for a focal length of 0.1m is in focus at the centre of the parabola but quickly goes out of focus away from the centre as the reconstruction focal length increases.

This example clearly demonstrates the ability of the direct-search method to design holograms for the production of three-dimensional image distributions.

6.5 Designing arrays of points

When the direct-search method is used to design arrays of points and the points are well separated, the phase of the field at a given image sample point can be

²⁶A cuboid containing all the image samples in the three-dimensional design.

2-d alpha focus reconstructed over parabolic

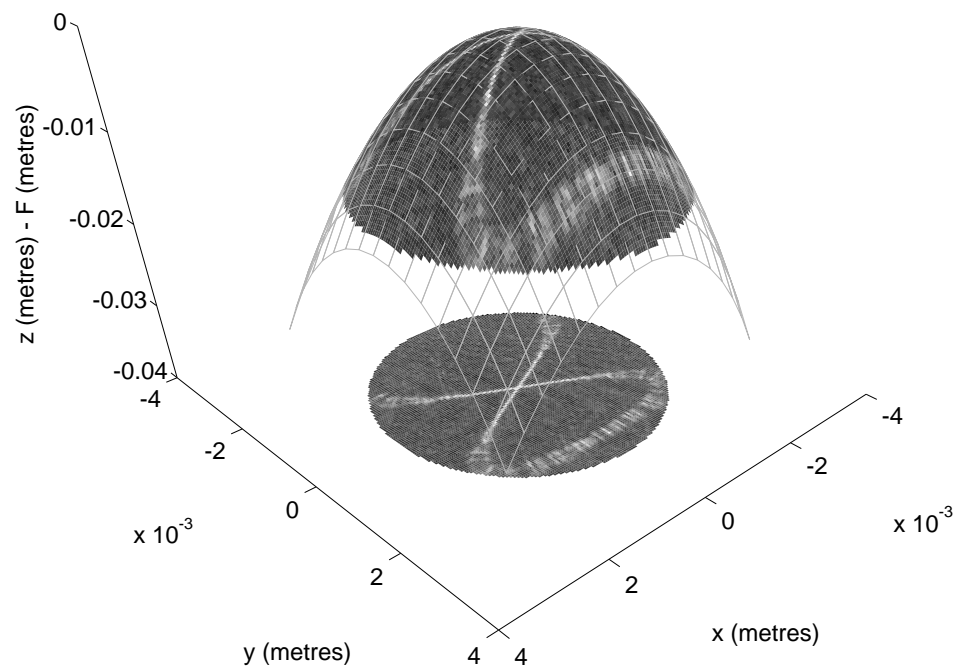


Figure 6.56: A 2-d design reconstructed over the parabolic surface shown in figure 6.55.

considered independent from the phase of all the other image sample points. This results in the phase at each image point being a degree of freedom for the design and an “easier” design problem.

Three array sizes were considered with 2×2 , 4×4 and 8×8 image points laid out on a regular grid. The working distance for all these designs was 0.1m and the number of hologram pixels was 128×128 . The hologram designs and reconstructions are shown in figures 6.59, 6.60 and 6.61. The size of the points in the reconstructions is determined by the *NA* of the entire hologram (see Section 6.1.1). This may be contrasted with arrays of lenslets that may also be used to produce arrays of points. An array of lenslets divides the whole aperture into the required number of sub-apertures each of which contains a small lens (a lenslet) used to produce a single point. Thus for an array of lenslets the more points that are required the larger

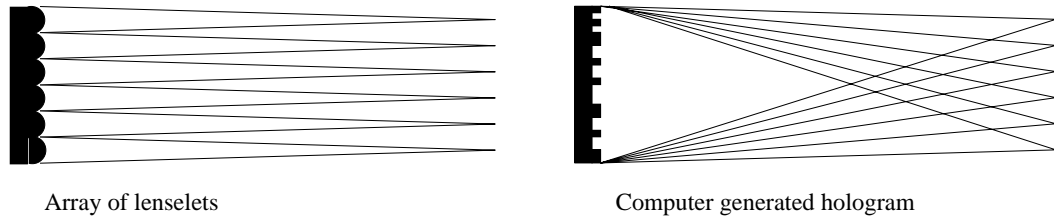


Figure 6.57: Different apertures operating for a computer generated hologram used to produce an array of points and an array of lenslets used to produce an array of points.

the image point spread function gets (for the same total aperture). Thus the image energy is spread over a greater area and the image intensity is reduced.

For a computer generated hologram producing a regular array of $N \times N$ points the peak intensity of the points may be expected to vary as $1/N^2$. For an array of lenslets with the same total aperture size the peak intensity may be expected to vary as $1/N^4$

Computer generated holograms are not suitable for some applications, for instance as wavefront sensors²⁷, as each image point contains information relating to the entire aperture whereas the image points generated from an array of lenslets only contain information relating to their own sub-aperture. Applications which would benefit from the use of computer generated holograms to generate arrays of points rather than arrays of lenslets include optical interconnects and optical clock distributors.

The hologram designs in this section are self-focusing and require no additional optics. They may also be tailored to the specific input illumination and thus provide an even intensity array of points from uneven illumination.

Figure 6.58 shows a scatter plot of the intensities at the image sample points for the three array designs. It can be seen that the intensities are highly uniform and that the intensity is inversely proportional to the number of image sample points as expected. The standard deviation of the intensity at the image sample points is less than one percent of the mean image sample intensity for each design. It can be seen from figures 6.59, 6.60 and 6.61 that the size of the image points is *not* related to the number of points.

²⁷E.g Shack-Hartmann wavefront sensor.

As the phase at the image sample points can be considered independent of each other the phase at each point might be expected to be random. The diagram shown in figure 6.62 shows the simulated reconstruction of the 8×8 array design, superimposed over each sample point is an arrow. The length of the arrow indicates the intensity at the image sample point whilst its direction indicates the phase. It can be seen that the direction of the arrows does appear random, while the lengths of the arrows are very nearly the same as each other.

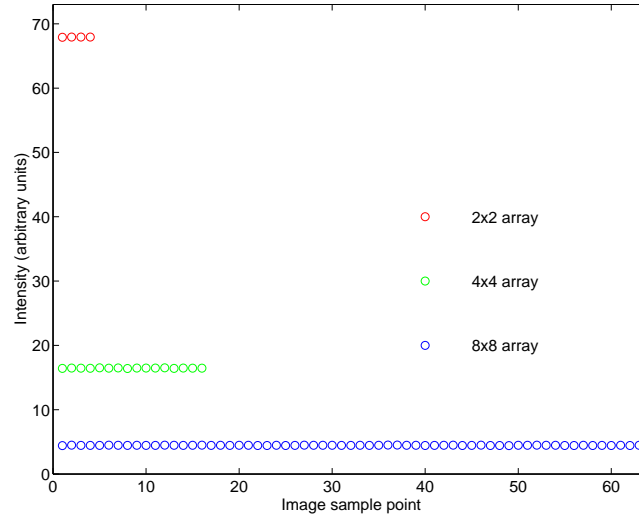


Figure 6.58: Image sample intensity for array designs.

6.6 Tailoring the design to the input illumination

As mentioned in previous chapters the direct-search method allows tailoring of each hologram design to the incident illumination. To demonstrate this the direct-search method was used to design five holograms using different apertures and different illumination profiles. Apart from the differences in illumination, the design specification was identical for each design.

6.6.1 Tailoring the design to the hologram aperture

The aperture over which the optimisation was performed will be referred to as the “design aperture”.

If the design is not tailored to the incident illumination the performance of the hologram suffers. To demonstrate the effect of this the five tailored designs were each reconstructed five times, once with each different aperture. The aperture with which the reconstruction is performed is referred to as the reconstruction aperture. Where the reconstruction aperture transmits light but the design aperture does not, light is transmitted with the phase of the incident illumination. Where the design aperture is taken to transmit light but the reconstruction aperture does not, no light is transmitted. Where both transmit then all transmitted light is modulated by the hologram phase distribution (where both of the apertures do not transmit light then no light is transmitted).

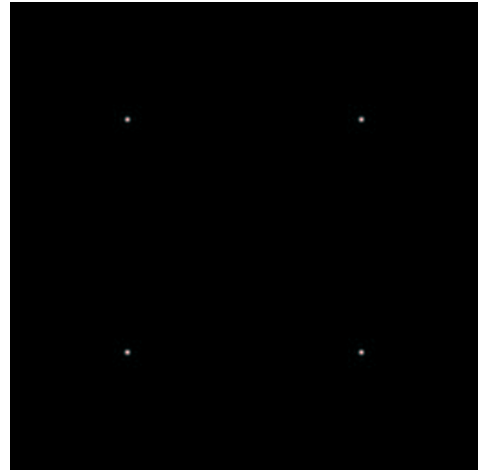
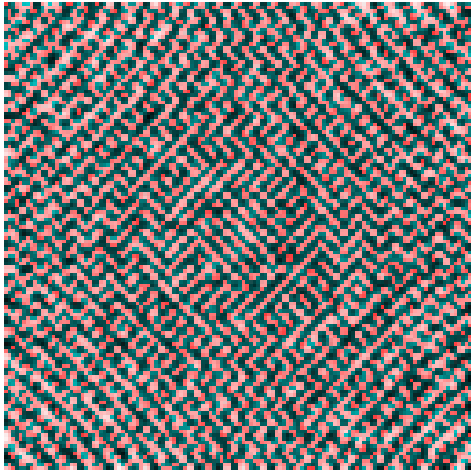


Figure 6.59: Hologram and reconstruction for “2x2 array” design.

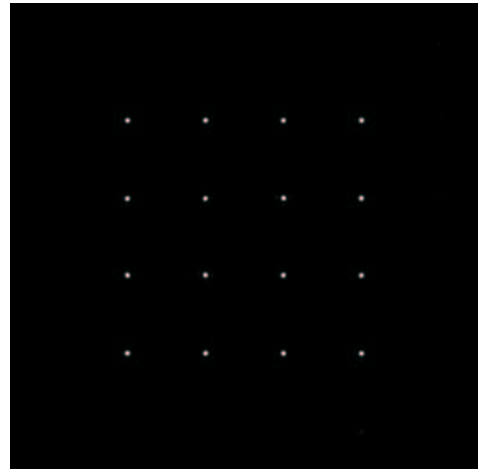
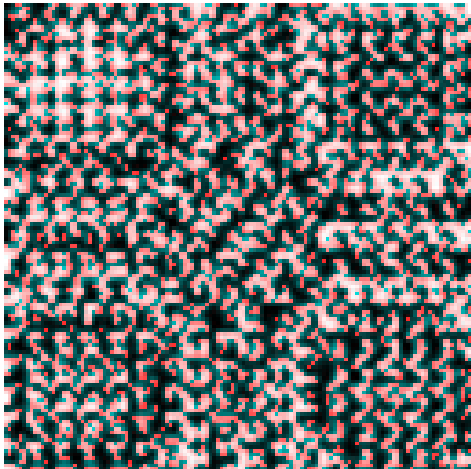


Figure 6.60: Hologram and reconstruction for “4x4 array” design.

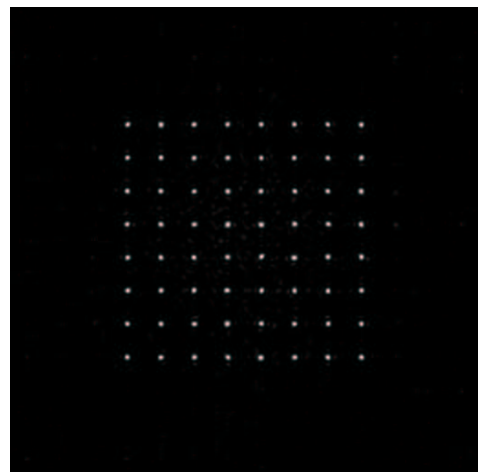
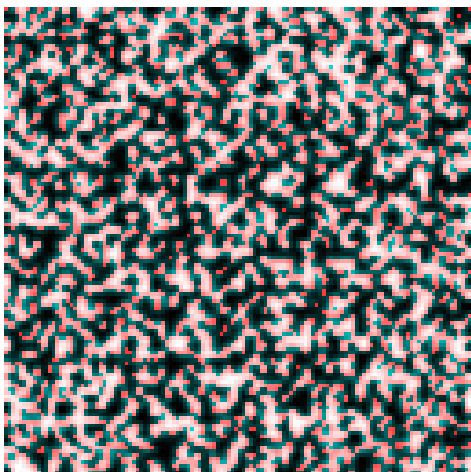


Figure 6.61: Hologram and reconstruction for “8x8 array” design.

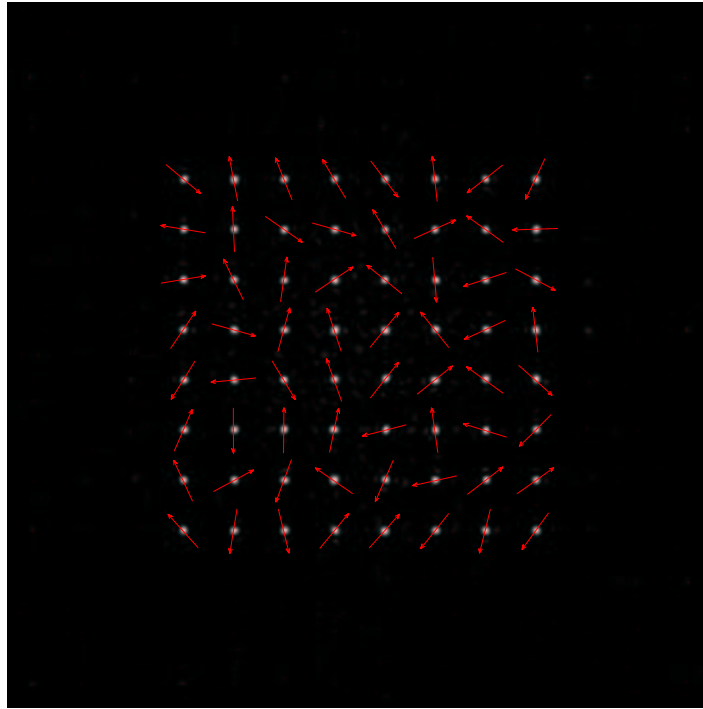


Figure 6.62: Reconstruction of “8x8 array” design with arrows indicating the complex amplitude at the image sample points.

Figure 6.63 shows the five holograms designs reconstructed using the five different apertures. The design aperture is shown down the left-hand side of the diagram and the reconstruction aperture for the simulation is shown along the top. The five different apertures were; square, square with a central square stop, circular, circular with a central circular stop and two wide rectangular slits. Each aperture could be contained within the 128×128 pixel square aperture. It can be seen from figure 6.63 that the best performance for each design occurs when it is illuminated using the same aperture for which it was designed (leading diagonal).

The effect of reconstructing a hologram with a different aperture from the design aperture is complex. A number of different effects are involved. The areas where extra light is transmitted give rise to extra features in the image, this is evident in the reconstructions shown in figure 6.63. Areas which are removed from the hologram design result in loss of information which degrades the quality of the reconstruction, however, the effect of this is complex and very problem specific. In addition, the removal of areas the changes the shape and size of the image point-spread function²⁸. The effective aperture is a combination of the transmitting areas of both the design and the reconstruction apertures. Area which is added to the design aperture does not usually contribute to the image and therefore does not affect the image point-spread function significantly. Figure 6.64 shows the signal-to-noise ratio for the simulated reconstructions shown in figure 6.63. The signal-to-noise ratio for each design is highest when it is reconstructed using the aperture for which it was designed (leading diagonal). The effect of reconstructing with a different aperture depends not only on the change in the area of the aperture but also on its change in shape.

²⁸The exact effect of this is again highly problem specific.

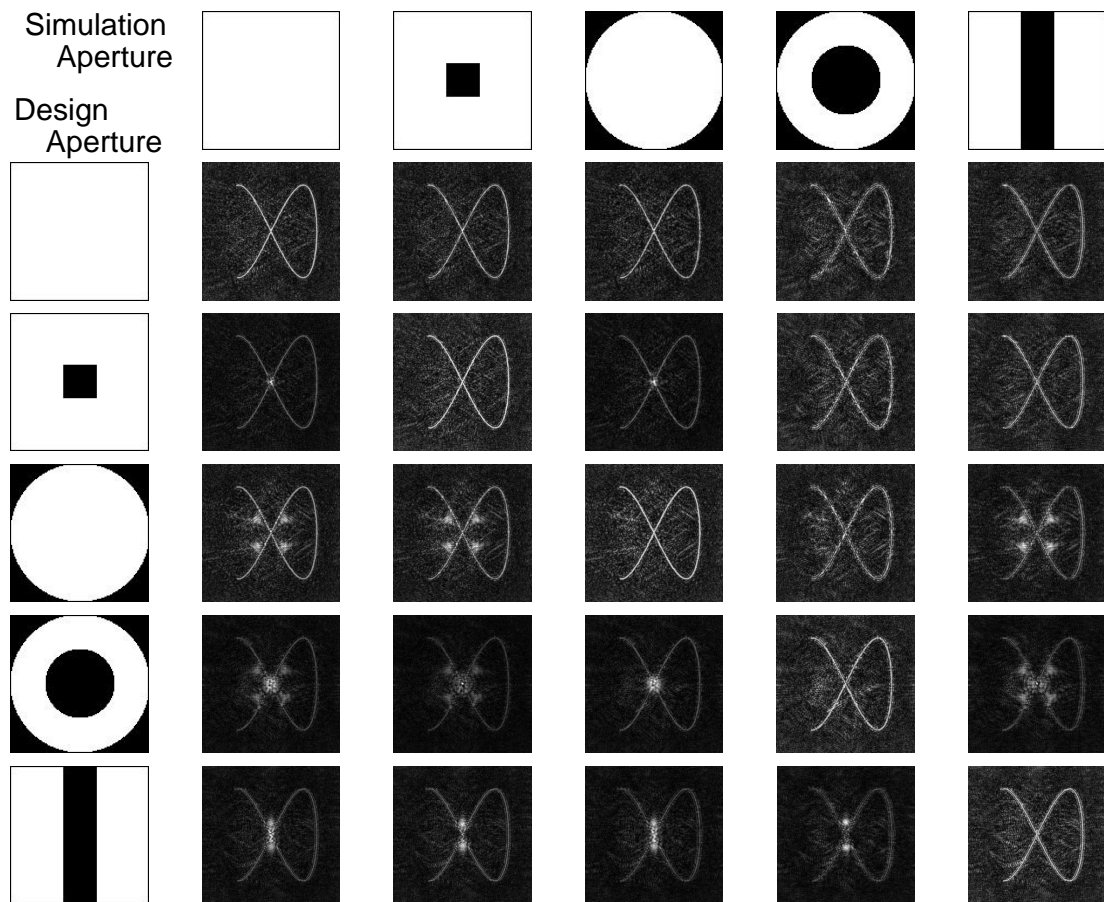


Figure 6.63: Diagram showing designs tailored to an aperture reconstructed using different apertures.

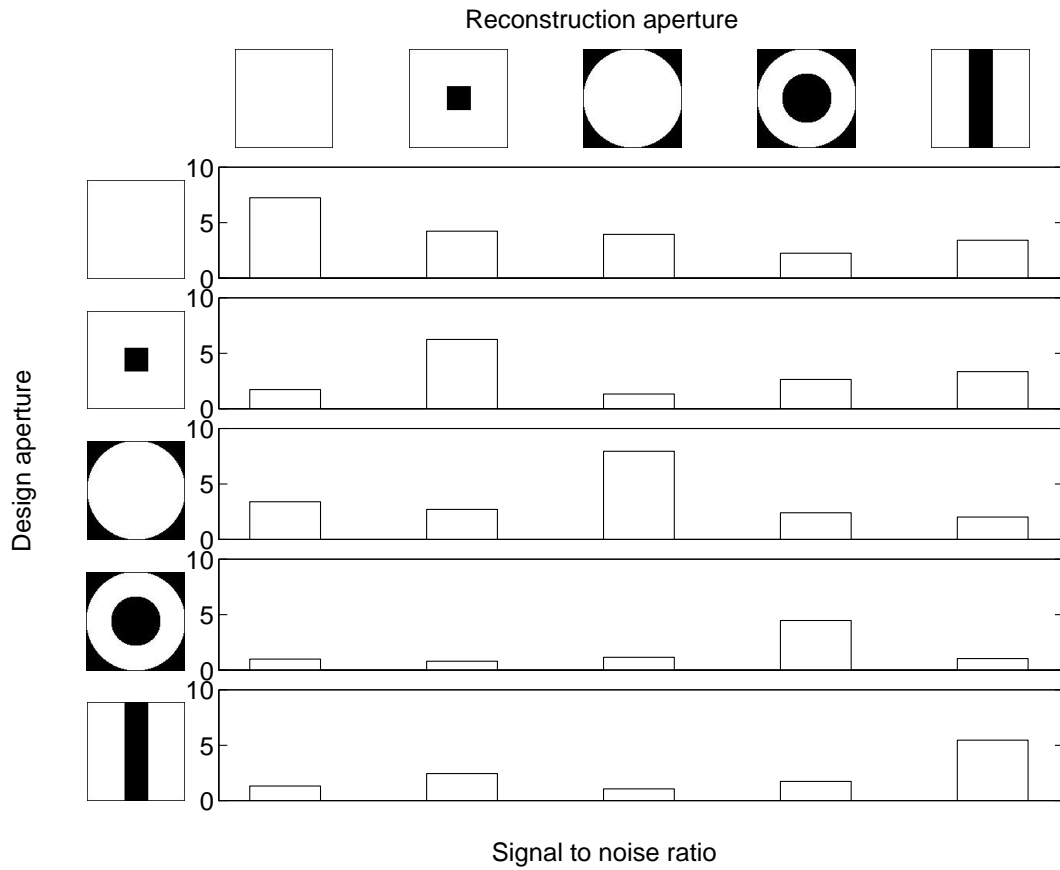


Figure 6.64: Diagram showing signal-to-noise ratio for designs tailored to specific apertures reconstructed with different apertures.

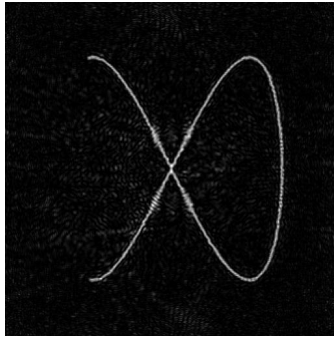
6.6.2 Uniform and Gaussian input illumination

The direct-search method can be tailored so that the hologram is designed for a specific input illumination. The examples thus far have been for a uniformly illuminated aperture. This is a convenient arrangement for laboratory examples. For laser machining applications the total efficiency is important, measured as the proportion of the energy leaving the laser ending up in the required place. In this case the hologram should be able to cope with illumination directly from the laser. This would commonly be a circularly symmetric Gaussian beam profile. For a Gaussian beam the algorithm can be tailored to the input illumination by setting the amplitude factor $U_s(\vec{r}_s)$ in equation 4.6 to be

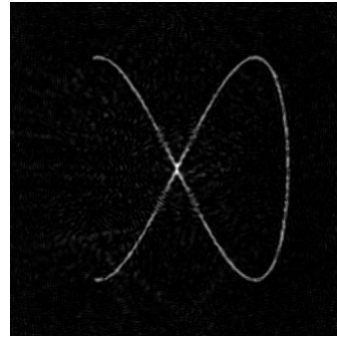
$$U_s(\vec{r}_s) = U_{pixel} = A_0 e^{-\frac{r^2}{2w^2}} \quad (6.17)$$

where A_0 is the amplitude at the centre of the Gaussian beam, r is the distance from the centre of the hologram to the pixel and w is the $\frac{1}{e^2}$ intensity radius. In the following example the alpha shape was designed for a square aperture with uniform illumination and the same aperture with Gaussian illumination. The $1/e^2$ width of the Gaussian was set to $1/2$ of the width of the aperture and the beam was centred at the centre of the aperture. Figure 6.66 shows the intensity distribution along the

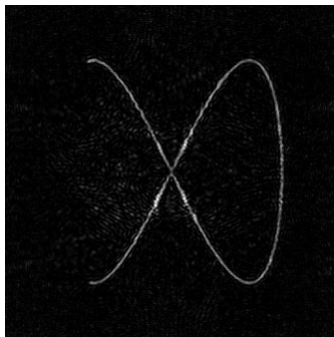
Design: plane, illumination: plane



Design: plane, illumination: Gaussian



Design: Gaussian, illumination: plane



Design: Gaussian, illumination: Gaussian

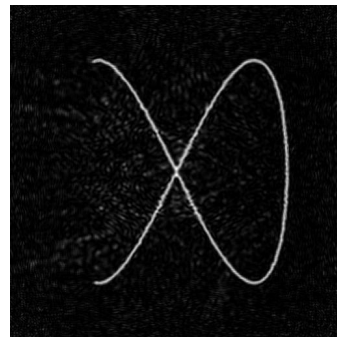


Figure 6.65: Comparison of results for uniform and Gaussian illumination reconstructed using uniform and Gaussian illumination.

line of the reconstructed alpha images shown in figure 6.65.

It can be seen from figures 6.65 and 6.66 that the intensity variation along the image lines is much greater for the holograms reconstructed with illumination different from the design illumination. In particular spikes and dips in the intensity are evident around samples 100 and 400. These occur at the centre of the image where the line of the alpha crosses over itself and can be seen in figure 6.65. It

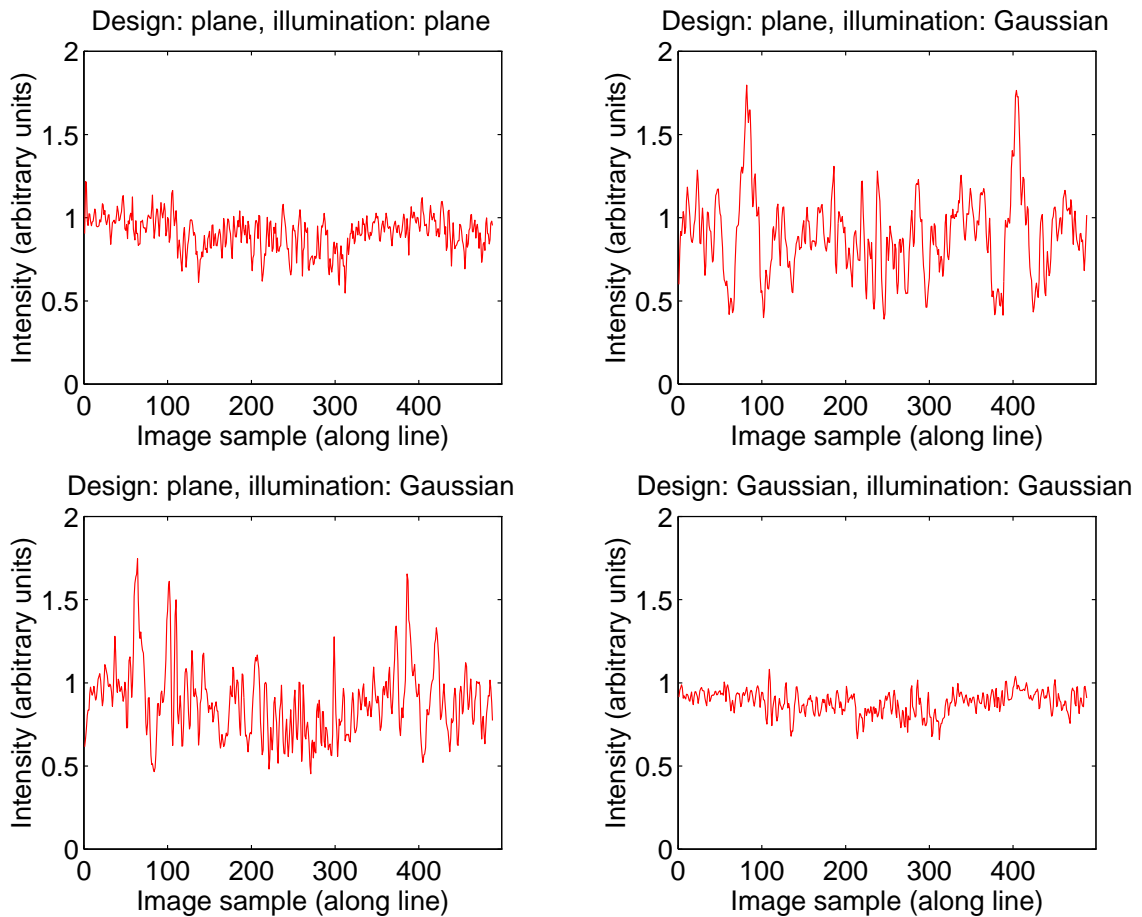


Figure 6.66: Plot of the intensity along the line of the alpha shapes shown in figure 6.65.

is clear from figure 6.66 that tailoring the design to the incident illumination can improve the quality of the reconstructed images. This does not incur a significant additional amount of computation.

6.7 Comparison of fixed-target cost function and state-variable cost function based direct-search methods

This section compares fixed-target and state-variable cost function direct-search methods. The results for the fixed-target cost function are shown in Section 6.3.1, page 91 and the results for the state variable cost function are shown in Section 6.3.2, page 95. This section is intended to compare the results of the two methods

and to illustrate the progress that has been made since changing from the use of fixed-target cost functions.

Both the designs shared the same design parameters, basic algorithm and algorithm design.

The results presented in this section for the fixed-target cost function represent the “best” results that could be found by adjusting the target value and repeating the optimisations (see Section 6.3.1). Determining the optimal value of the target took several optimisations and varied with each individual design.

The results presented in this section for the state-variables cost function (see equation 6.11) were found using the cost balancing parameters a and b set to 2 and 1 respectively. It has been found that these values produce reliable results for most design problems (see Section 6.3.2).

The figure 6.67 shows the designs, reconstructions and line intensities made using fixed-target and state-variables cost functions. The line intensities shown in figure 6.67 are taken from the optimisation algorithm and calculated at the image sample point coordinates. The line intensity for the fixed-target cost function design shows considerably more noise than the design made using state-variables cost functions. It also has several regions where the intensity drops considerably, these are not present in the design made using a state-variables cost function. These regions are very narrow, typically a single sample spacing across, and are probably caused by phase discontinuities along the line (see Section 6.3.3, page 97).

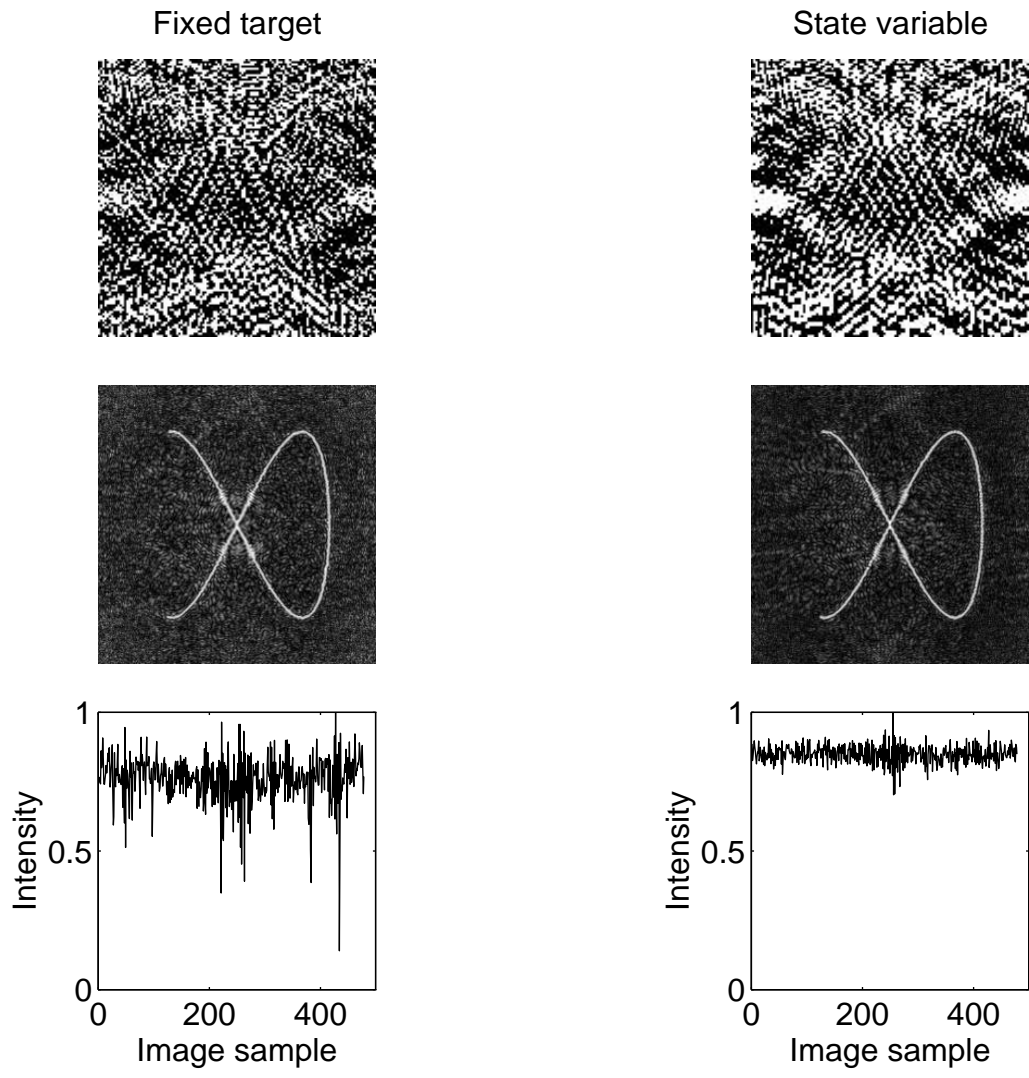


Figure 6.67: Simulations and designs for fixed-target and state-variables cost functions.

Chapter 7

Conclusions and future work

7.1 Conclusions

The direct-search method presented in this thesis is shown to be a highly reliable and high quality method for the computer design of holograms. It is capable of producing high quality solutions for computer generated hologram problems for a wide variety of applications including, optical interconnects, laser machining applications and laser generated ultrasound.

7.1.1 Existing work on computer generated hologram design

Chapter 2 explored published computer generated hologram techniques. It was clear from the existing work that the previously available design techniques were not capable of exploiting the full potential of computer generated holography. The existing techniques are not suitable for many of the most useful applications of computer generated holograms, for instance laser machining. Many of the techniques show unrealistic simulations of the designs which would not perform well as optical components. This is often due to the use of Fourier techniques and unrealistic models. The use of the fast Fourier transform may save computing time if the design specifications are restricted but, unless the technique is used with care, additional speckle noise is likely to occur when the designs are optically reconstructed. It is clear that the problem of image speckle has not been tackled adequately for methods such as projection on constraints.

This together with the inflexible design restraints imposed by the use of the fast Fourier transform technique means that these techniques are unsuitable for general computer generated hologram design.

Simulated annealing and genetic algorithms were also investigated but it is clear that these techniques consume an immense amount of computing time. It is not usually necessary to find the globally optimal solution to a CGH design problem, so the ability of these techniques to approach the globally optimal solution, at the expense of extensive computation, is not required.

Chapter 2 concluded with an introduction to direct-search methods.

7.1.2 Hologram fabrication and importance to hologram design

In Chapter 3 the important issue of hologram fabrication was discussed and some of the many hologram fabrication techniques available were examined. The hologram fabrication technique should be considered throughout the design process as it ultimately limits the hologram bandwidth, the size of the smallest hologram feature and the number of quantisation levels available. These are all extremely important parameters in the design of practical computer generated holograms.

7.1.3 Direct-search method and the underlying model

In Chapter 4, the direct-search method and the underlying model are presented. The model imposes several conditions upon the designs concerning the sampling and space-bandwidth product of the hologram and the image. These must be adhered to or else the direct-search method will produce unsatisfactory results. It is clear from this model that the direct-search method is extremely flexible with respect to the optical geometry of the design; there are no conditions prohibiting three-dimensional holograms or images.

Chapter 4 goes on to describe some of the most important aspects of the optimisation method itself, the cost function and the stopping function. These can have a significant effect upon the speed of optimisation, the quality of the final design and the reliability of the direct-search method. These aspects are explored comprehensively in Chapter 6.

7.1.4 Computational and analytical methods

Chapter 5 explored the practical and computation techniques involved with the direct-search method and the measurement of its performance. A technique was developed to provide good-quality, realistic information about the designs. This involved simulating the designs by taking an angular-spectrum representation of the hologram calculated using the fast Fourier transform algorithm and propagating this to a reconstruction plane where the simulated wavefront can be calculated again using the fast Fourier transform algorithm. This image is then divided up into three regions using mask functions derived from the original image data and information is extracted relating to the efficiency, mean image intensity, image noise and signal to noise ratio. Information relating to the background intensity can also be extracted.

These simulations provide an accurate representation of the optical performance of the hologram provided the correct sampling is used. Comparison of simulated and optical reconstruction was discussed and the usefulness of the simulations were demonstrated.

7.1.5 The direct-search method in practice

In Chapter 6 a comprehensive study of the direct-search method is presented. The basic design parameters, the design of algorithm and the cost function design were studied. The ability of the direct-search method to design holograms to produce three-dimensional images and arrays of points was demonstrated. The ability of the

direct-search method to tailor each design to the incident illumination was demonstrated (Section 6.6.2).

Basic design parameters

Section 6.1 showed that the critical sampling rates derived for the hologram sampling and image sampling in Chapter 4 do represent the minimum required sampling rate to produce a successful computer generated hologram.

The performance of the final design could be improved if the hologram sampling rate was increased above the critical rate. This was to be expected because as the sampling rate increases the hologram pixel size decreases and the proportional of the area of the hologram not set to an optimal phase level is reduced (Section 6.1.1). The critical hologram sampling rate is sufficient to encode a design capable of reconstructing the entire image. The performance of algorithm is unaffected by increasing the image sampling rate above the critical rate (Section 6.1.1).

The CPU time required to perform the optimisations varied roughly linearly with both the total number of hologram and image samples.

The variation in hologram performance with the number of phase levels was found to be consistent with previously published studies [31, 46, 68]. The CPU time required to perform the optimisations varies roughly linearly with the number of phase levels (Section 6.1.2).

Direct-search algorithm design

Section 6.2 investigated details of the design of the direct-search algorithm. The order in which the hologram pixels are chosen (Sections 6.2.3 and 6.2.3), was found not to affect the final quality of the solutions found, but can significantly change the amount of CPU time required perform the optimisation. Section 6.2.3 showed that the overwhelming majority of accepted hologram pixel changes occur on the edge of fringe boundaries.

When the number of phase levels was greater than two there were several different ways in which to choose the new phase levels for each trial (Section 6.2.3). Five of these were tested. Unrestricted access to all the phase levels resulted in faster optimisation¹ than restricted phase level access. Random phase selection was preferred over sequential phase selection because sequential phase choice affected the stopping function and reduced the direct-search method's reliability.

The initial starting conditions (Sections 6.2.1 and 6.2.1) were shown not to significantly affect the quality of the final solutions.

Section 6.2.1 showed that by starting with a “dark” solution, that is one where all the hologram pixels initially contribute nothing to the solution, the computational overhead of calculating the initial solution can be avoided.

Cost function design

The cost function controls the optimisation behaviour of the direct-search method and is therefore important to its success. Section 6.3 examined target based and state variables based cost functions (some additional cost functions were discussed

¹The method of *sequential phase, incremental number of phase levels* resulted in a shorter optimisation time because it seriously affected the stopping function.

in Appendix D). Cost functions designed to control the image phase distribution with the aim of improving the image intensity distribution were discussed in Section 6.3.3. Section 6.3 demonstrated the superiority of the state variables cost function approach. This type of cost function maintains a balance between the aims of high efficiency and low-noise of the image intensity throughout the optimisation. This helps to suppress image defects and increase the direct-search method’s reliability (Section 6.7). Furthermore this type of cost function does not require any adjustment for use with specific design problems. Additional phase terms are not required to improve image quality with state variable based cost functions (Section 6.3.3). Target based cost functions may benefit from such additional terms but balancing such a cost function is difficult.

Sections 6.4, 6.5 and 6.6 demonstrated the direct-search methods ability to produce three-dimensional images, array images and tailor the designs to the input illumination.

Section 6.7 compared two designs, one made using a fixed-target cost function and the other using a state variables cost function. The fixed-target design was the best, chosen out of many designs made with different targets. The state variables design was made using the standard values for the cost balancing parameters, which are suitable for all designs. The state variables design approach was found to be significantly better than the fixed-target design approach, resulting in much lower noise and higher efficiency.

7.1.6 Summary of optimal direct-search method

Table 7.1 shows a summary of important direct-search parameters and optimal settings investigated and found in this thesis. These form the basis for a robust and general technique for CGH design. This technique is not restricted to any particular optical setup and is capable of designing holograms that can produce continuous (as well as discrete) intensity distributions in two- and three-dimensions.

Parameter	Useful value
Hologram sampling rate	\geq critical rate
Image sampling rate	critical rate
Stopping Parameter (P_{stop})	$5\% > P_{\text{stop}} \times (p_1 - 1) > 1\%$ according to CPU time
Phase levels (P_1)	according to fabrication and CPU time
Pixel choice	random (exhaustive)
Phase choice	random
Pixel preselection	always
Start state	dark (all pixels off)
Cost function	state variables $C = -aI + b\sigma_I$, $a \approx b$
Phase cost	unnecessary

Table 7.1: Summary of optimal direct-search parameters.

7.1.7 Conclusion

The direct-search method developed in this thesis is a computational optimisation technique for the computer design of holograms. This method optimises a bandwidth-limited quantised phase distribution in order to produce a required image from a known wavefront and hologram aperture. The image may be three-dimensional, as may the surface on which the hologram phase distribution is defined. The smallest image feature that may be produced depends upon the numerical aperture of the hologram. The efficiency of the hologram designs is primarily determined by the number of phase levels available. The typical efficiency for a binary element is around 35%². This rises to around 70% when 8 or more levels are used. Whilst this method is computationally intensive and the designs presented in this thesis were made (at the time of the research) on a supercomputer, the computing power required to perform individual designs is not beyond the power of common desktop computers available at the time of the submission of this thesis.

7.2 Future work

7.2.1 Wavelength multiplexing

The direct-search method could be generalised so that holograms could be designed to image at multiple wavelengths. This is a similar problem to that of focal length multiplexing (Section 6.4.2). The calculations for the image at each wavelength would have to be made and the implementation of the hologram pixel phase value would have to be changed. The phase value stored at each pixel would have to be considered as an “etch depth” from which the phase shift at each wavelength could be calculated.

Characterisation of the best approach to choosing the best “etch depths” for a given set of materials and wavelengths would be required.

Possible applications for wavelength multiplexed holograms would include “broad-band” devices and devices to achieve spatial multiplexing and de-multiplexing of multiple wavelength signals.

7.2.2 Improving the speed of computation

The existing mathematical and computational techniques are very general. With a slight restriction to the flexibility of the algorithm the speed of computation could be increased by replacing the computation of the phase between the hologram pixel and image sample with an mathematical approximation and/or use of a look up table. Problems with vectorisation, storage and memory bandwidths may reduce the effectiveness of such work.

7.2.3 Non-scalar theory model for use with the direct-search method

The scalar theory model used in this thesis breaks down when the required diffraction angles become large or when the hologram contains very small features. The work

²Efficiency as defined in Chapter 5.

presented in this thesis is firmly within the scalar regime and is suitable for the applications presented. An investigation into the limits of this theory would be useful and the development of a non-scalar theory model would allow very high numerical aperture holograms and holograms carrying very small features to be designed using the direct-search method.

7.2.4 Investigation of the effect of hologram fabrication defects

Characterisation of hologram fabrication defects may allow adaption of the direct-search method so that hologram designs are tolerant of fabrication defects. It may also be possible to suppress hologram features that may lead to an increase in the number of defects, for instance it would be possible to discourage the presence of single pixels with phases different from all their neighbours which may be cause fabrication difficulties.

7.2.5 Gray scaling

The holograms presented in this thesis have been designed with only one desired image intensity level (and the background). It is possible to change the cost function so that gray level images can be produced. Investigation into the effect of noise when designing gray level images is required so that the effective number of image gray scales can be ascertained.

7.2.6 Development of a model for predicting the number of optimisation cycles

A method for estimating the number of optimisation cycles required to perform an optimisation and the CPU time required for each cycle would help predict the total CPU time required for each design. This would help users of the method to allocate their resources

7.2.7 Development of a specialised 1-D algorithm for pseudo 1-D devices

For some applications the use of a one dimensional direct-search algorithm may have advantages and could save a considerable amount of CPU time. Applications with linear or rotational symmetry may be suited to this technique. The current model is adequate for linear one-dimensional designs but additional work is required for the model to produce designs with rotational symmetry. The stopping function would require additional work as there may not be enough hologram samples to collect meaningful information about the probability of accepting a change. This however does not present a significant problem as such a small number of hologram samples presents a very small computing load.

Appendix A

Derivation of the Kirchhoff integral.

The Fresnel-Kirchhoff integral used as the basis for the direct-search model in this thesis is derived from the Helmholtz-Kirchhoff diffraction integral. The derivation is standard and found in many texts (see [67]). It is instructive to derive the Kirchhoff integral here to define the coordinates and notation. It is also possible to identify the most significant mathematical approximations which can be used to identify the limits of the model used.

Assuming the scalar limit and working from the Helmholtz-Kirchhoff diffraction integral the amplitude at an image point, \vec{r}_i , is given by

$$U_i(\vec{r}_i) = \frac{1}{4\pi} \oint_{\sigma} \left[\frac{e^{-ik|\vec{r}_i - \vec{r}_s|}}{|\vec{r}_i - \vec{r}_s|} \frac{\partial U_s(\vec{r}_s)}{\partial \hat{n}_{\sigma}} - U_s(\vec{r}_s) \frac{\partial}{\partial \hat{n}_{\sigma}} \frac{e^{-ik|\vec{r}_i - \vec{r}_s|}}{|\vec{r}_i - \vec{r}_s|} \right] d\sigma \quad (\text{A.1})$$

where $k = 2\pi/\lambda$, \vec{r}_s is the position on the closed surface σ , U_s is the amplitude at \vec{r}_s , \hat{n}_{σ} is the normal to surface σ at point \vec{r}_s (see figure A.1). The time dependence of the amplitude is neglected for clarity. This starting point means that the calculations are only valid in the scalar regime.

Rewriting the first derivative and taking a local plane wave approximation for $U_s(\vec{r}_s) \approx \hat{n}_s U_s e^{-ik|\vec{r}_s|}$ giving,

$$\frac{\partial U_s(\vec{r}_s)}{\partial \hat{n}_{\sigma}} = \hat{n}_{\sigma} \cdot \hat{n}_s \frac{\partial U_s(\vec{r}_s)}{\partial \hat{n}_s} = -ik U_s(\vec{r}_s) \hat{n}_{\sigma} \cdot \hat{n}_s \quad (\text{A.2})$$

where \hat{n}_s is the normal to the wavefront U_s at \vec{r}_s . This implies that the Kirchhoff integral loses validity if this approximation cannot be made. This will be the case if the range over which the a local plane wave approximation can be made is less than the wavelength of the source light.

The second term in equation A.1 can be rewritten thus

$$U_s(\vec{r}_s) \frac{\partial}{\partial \hat{n}_{\sigma}} \left[\frac{e^{-ik|\vec{r}_i - \vec{r}_s|}}{|\vec{r}_i - \vec{r}_s|} \right] = U_s(\vec{r}_s) \hat{n}_{\sigma} \cdot \hat{n}_i \frac{\partial}{\partial R} \left[\frac{e^{-ikR}}{R} \right] = -U_s(\vec{r}_s) \hat{n}_{\sigma} \cdot \hat{n}_i \left[ik + \frac{1}{R} \right] \frac{e^{-ikR}}{R} \quad (\text{A.3})$$

where \hat{n}_{σ} is the unit vector in the direction of $\vec{r}_i - \vec{r}_s$ and $R = |\vec{r}_i - \vec{r}_s|$.

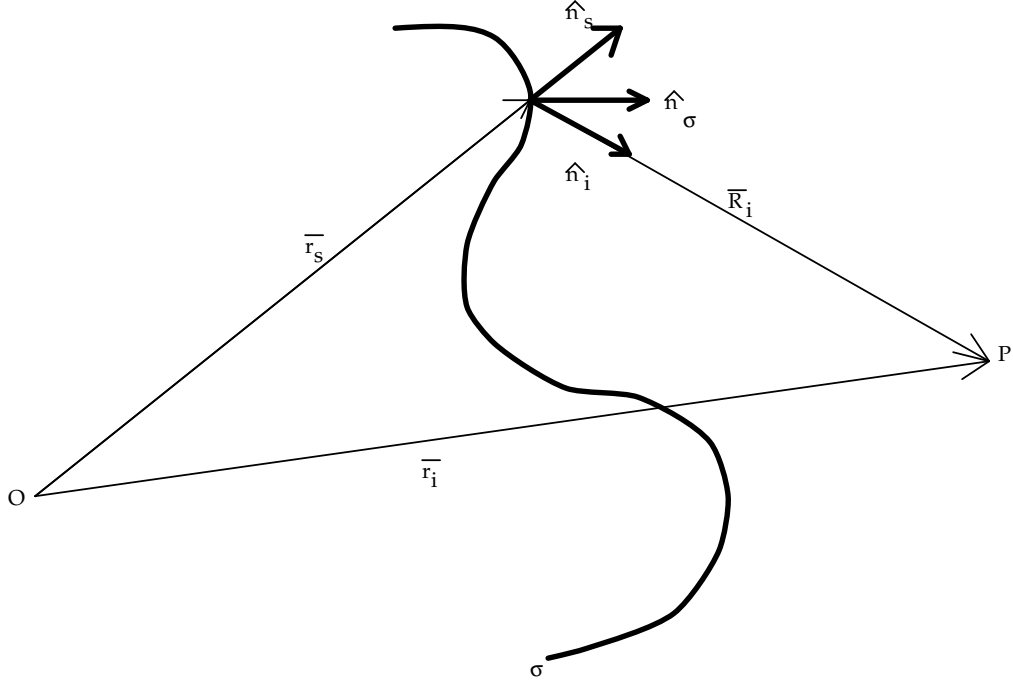


Figure A.1: Diagram showing Kirchhoff integral vectors.

If $R \gg k$ then the following approximation can be made

$$U_s(\vec{r}_s) \frac{\partial}{\partial \hat{n}_\sigma} \left[\frac{e^{-ik|\vec{r}_i - \vec{r}_s|}}{|\vec{r}_i - \vec{r}_s|} \right] \cong -ik U_s(\vec{r}_s) \hat{n}_\sigma \cdot \hat{n}_i \frac{e^{-ikR}}{R} \quad (\text{A.4})$$

This implies that the Kirchhoff integral is not valid close to the surface of integration. This is not usually a problem for most CGH design. This will be significant for devices with very short focal lengths ($F \approx \lambda$) [69] when a more rigorous model may be appropriate [70].

Substituting equations A.2 and A.4 into equation A.1 gives,

$$U_i(\vec{r}_i) \cong \frac{-ik}{4\pi} \oint_\sigma U_s(\vec{r}_s) \frac{e^{-ikR}}{R} [\hat{n}_\sigma \cdot \hat{n}_s - \hat{n}_\sigma \cdot \hat{n}_i] d\sigma \quad (\text{A.5})$$

rearranging gives

$$U_i(\vec{r}_i) \cong -i\lambda \oint_\sigma U_s(\vec{r}_s) \frac{e^{-ikR}}{R} \frac{[\cos \theta_s - \cos \theta_i]}{2} d\sigma \quad (\text{A.6})$$

which is the usual form of the Fresnel Kirchhoff integral used as the starting point in Chapter 4.

Appendix B

Effect of square hologram pixels

The direct-search method uses equation 4.4 to calculate the image complex amplitude from the hologram design. This equation implies that each square pixel of the hologram is approximated by a point source emitting the same energy as the finite pixel it replaces.

This has some small consequence when the hologram is reconstructed optically. Working from Goodman[4] the effect of the the use of finite square pixels rather than point sources at reconstruction can be discussed.

The complex amplitude at the hologram can be considered as a convolution between the collection of point sources emitting light with the design phase and a single pixel.

The actual hologram complex amplitude U_{actual} can be written as

$$U_{\text{actual}} = \left[U_s e^{-i(\phi_s + \phi_d)} \times \text{comb}\left(\frac{x}{X}\right) \text{comb}\left(\frac{y}{Y}\right) \right] \otimes \text{rect}\left(\frac{X}{2}, \frac{Y}{2}\right) \quad (\text{B.1})$$

where

$$\text{comb}(x) = \sum_{n=-\infty}^{\infty} \delta(x - n) \quad (\text{B.2})$$

$\delta(x)$ is the Dirac delta function, X and Y are the size of the hologram pixels $\text{rect}(x, y)$ is defined in as

$$\text{rect}(a, b) = \begin{cases} 1 & -a < x < a \text{ and } -b < y < b \\ 0 & \text{otherwise} \end{cases} \quad (\text{B.3})$$

Figure B.1 shows the comb and rect functions and how the calculated phase (an array of point sources) convolved with the rect function representing a hologram pixel can give the actual phase of the hologram.

The hologram can be thought of as performing two tasks, it adds the image information to the wavefront and it adds focal power to the wavefront like a lens. At the correct working distance the image is brought into focus and the two functions can be thought of as separate, one adding a sampled pixellated phase distribution¹ and the other acting as a lens² Thus the image formed at the correct working distance can be thought of as the Fourier transform of the part of the hologram phase

¹NB. this phase distribution is not the same as the hologram phase distribution, it is equivalent to the hologram phase distribution *without* the focal power.

²This argument only applies to light from a given diffracted order at the working distance for that order.

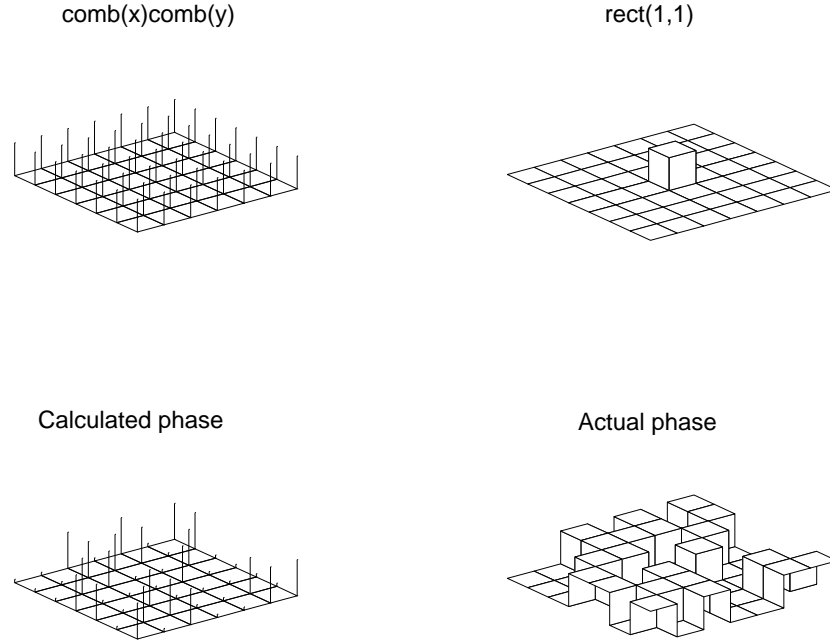


Figure B.1: Diagram showing how the actual hologram phase can be constructed as the ideal design phase convolved with the hologram pixel shape.

distribution that contains the imaging information. Thus the image formed at the correct working distance can be thought of as being the calculated image convolved with a comb function (as a result of the sampling of the hologram) and multiplied by a sinc^2 function (as a result of the pixellation of the hologram). The reconstructed image U_{image} is given as

$$U_{\text{image}} = \left[U_{\text{calculated}} \otimes Ft \left(\text{comb} \left(\frac{x}{X} \right) \text{comb} \left(\frac{y}{Y} \right) \right) \right] \times Ft(\text{rect}(X, Y)) \quad (\text{B.4})$$

where $U_{\text{calculated}}$ is the image complex amplitude calculated by the direct-search model and U_{image} is the actual image complex amplitude resulting from reconstruction with square pixels. The first term represents the calculated image, the second and third the effect of the pixellation and sampling respectively.

When the pixels become very small they are “rounded off” by diffraction effects not observed with scalar theory. The effect of the sampling (as a result of the second term) are convolutions of the image across the image plane. These will be diminished in intensity because of the third term (provided the sampling conditions are met they will be outside of the central lobe of the sinc^2 function) and by the neglected obliquity factor which diminishes the intensity of light diffracted by large angles.

The effect of the pixellation (as a result of the third term) results in the image intensity distribution being multiplied by a sinc^2 function. Provided the correct sampling conditions are met (Section 4.6) the image will be contained within the central peak of this sinc^2 function and be largely unaffected. Figure B.2 shows a simulated hologram images for an ideal hologram, a sampled hologram and a pixellated hologram, it also shows the distribution of the multiplying sinc^2 due to the square pixels.

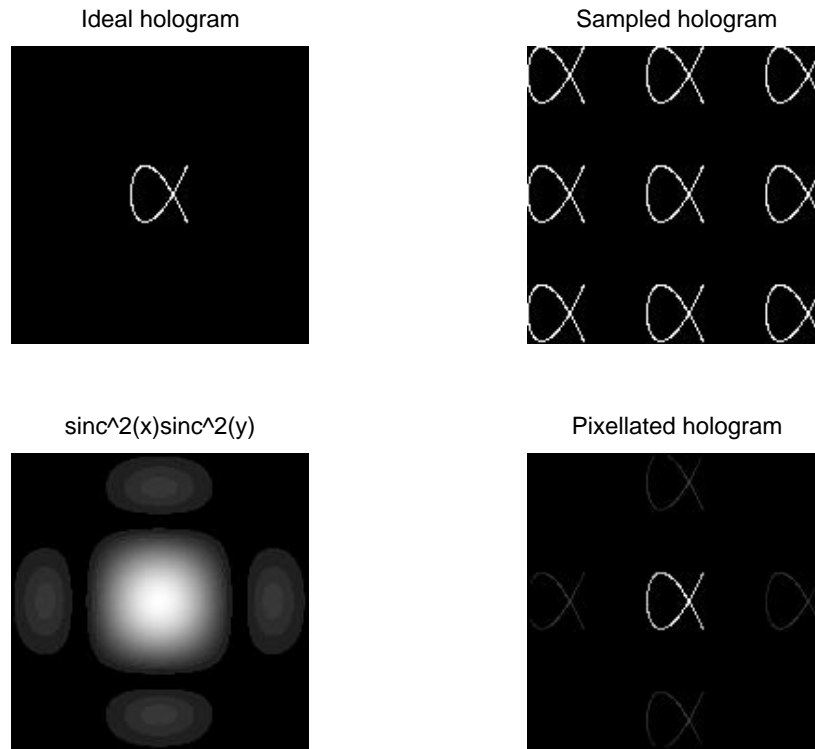


Figure B.2: Representation of the effect of sampling and pixellation of the hologram on the reconstructed image. The contrast of the lower two pictures has been reduced to make the side-lobes of the sinc^2 function more prominent.

Overall, provided the hologram is correctly sampled, the effect of the sampling is small as the higher order convolutions of the image will be very dim because of the obliquity factor. These can also be diminished by increasing the hologram resolution beyond the critical rate (see Sections 4.6 and 6.1.1) as this will result in more “rounding off” of the pixels and moving the higher order convolutions out to greater angles where they will be diminished more by the obliquity factor. The distortion of the image intensity distribution by the sinc^2 resulting from the pixellation is less significant than the obliquity factor and will be ignored.

Appendix C

Computational restrictions

Computers are developing quickly. When the research for this thesis was started super-computing resources were required. At the time of writing desktop computers with equivalent or faster CPUs than the super-computer originally used are available at fairly modest prices. These computers are capable of performing a CGH design typical of the ones presented in this thesis in around 5 seconds. Within the next five years common desktop PCs will out strip these computing speeds. This appendix will be redundant within that time. The aim of the research was not to develop an exotic design method that would always be restricted to super-computers, but to develop a technique that would be useful on generally available computers.

The majority of computational work carried out for this thesis¹ was performed on a Cray YMP8 computer. This computer was situated at the Atlas Centre, Rutherford Appleton Laboratory, was installed using UK Research Council funding and at the time of installation was the fastest academic computer available in the UK. Its peak performance was 2.7Gflops, its practical performance for most algorithms that can be vector processed was around 170Mflops. It has 8 processors each capable of 333Mflops peak performance sharing a central memory of 128Mwords (1Gbyte). Each processor can only achieve its peak speed if it can “chain” the result of one vector unit² into the other. This would result from prolonged add and multiply operations on a simple array³. For most algorithms it was not possible to constantly chain adds and multiplies so only one of the vector units can operate at a time resulting in a practical speed of around half the peak processor performance. The system at the time of use was set up to allow interactive jobs to be run if the total CPU time per job was less than 10 minutes. Longer jobs were run as batch jobs and submitted to batch queues where the job priority is dependent upon the system resources required (CPU time, memory high water mark, special i/o devices). Most of the direct-searches performed for this work required less than ten minutes CPU time and were run as interactive jobs to avoid waiting for batch queues. The ten minute CPU time limit was used as a practical limit on the length of optimisations and the extent to which an individual optimisation parameter could be varied where no other sensible limit applied. The other practical limit encountered using the YMP8 was the 8Mword memory on interactive jobs, this memory requirement was not exceeded except when performing simulated reconstructions with more than

¹Numerical computation.

²Each processor has separate vector add and multiply units.

³Simple arrays with equally spaced addresses can be “tuned” to be vector processed and to avoid memory conflicts which have a considerable performance impact.

1024 × 1024 samples⁴ and consequently the design parameters were chosen to avoid simulated reconstructions larger than 1024 × 1024 for most of the designs.

C.1 Vectorisation and calculation of inverse tangents

The calculation of inverse tangents is important for the calculation of the phase of a complex number, U , stored as real and imaginary parts, the phase, ϕ of the real and imaginary parts is,

$$\phi = \arg(U) = \arctan\left(\frac{\Re(U)}{\Im(U)}\right) \quad (\text{C.1})$$

the phase of amplitude was computed using a C function `atan2(real,imaginary)`, or a C++ Complex member function `arg(Complex)`.

At the start of work on the Cray YMP8 the function `atan2` did not vectorise. This can be easily understood as the function must go through different range reduction procedures to return accurate results depending on the relative signs of the real and imaginary parts. This involves branching operations which were not readily vectorised. During the work on the Cray YMP8 the system was up-graded and the vectorisation of the function `atan2` permitted. However the vectorisation of this function resulted in a less significant speed up than was usually experienced; presumably because vectorising around the branching operation used in the scalar version of `atan2` results in less efficient than usual vectorisation.

The (non) vectorisation of `atan2` has a knock on effect in that all code contained within the same loop can only be vectorised if the function `atan2` can. This can be avoided by separating the part of the loop containing `atan2` into a separate loop with intermediate temporary memory storage of the results before and after this loop. However this results in increased memory band-width and would be unlikely to produce significant enough time savings to be worth the effort.

The C++ function `arg(Complex)` was found not to vectorise after the system upgrade, and this can only be regarded as a peculiarity (one of many) of the Cray C++ compiler at this time⁵.

⁴Memory bank conflict problems and low i/o bandwidth meant that the reconstruction program required more memory to running efficiently than might initially be thought.

⁵The C++ compiler on the YMP8 at this time was in fact `cfront`, a C preprocessor that translated C++ code into C code so presumably the C++ function `arg(Complex)` was translated into `atan2(real,imaginary)` and should have vectorised when `atan2` did.

Appendix D

Not so sensible cost functions

D.1 Very simple cost functions

$$C = - \sum_{\text{image samples}} I_{\text{image sample}} \quad (\text{D.1})$$

Equation D.1 is particularly bad, despite its apparently simple behaviour (it gets more negative as more light gets into the image samples); it treats all the image samples with equal significance regardless of the intensity at that image sample. This contrasts with target based cost functions where the change in intensity of an image sample whose intensity is far from the target value is more significant than one whose intensity is near the target value.

The results of direct-search optimisations using cost functions like D.1 are interesting, there is a tendency for a small number of image samples get very bright whilst the majority of the image samples stay very dark, in particular the hologram phase distributions generated almost always resemble the addition of a simple grating and a zone plate. These solutions are usually very efficient in the sense that they get a comparatively high proportion of the available light into the design region, but the light tends not to be spread throughout the design region. The small areas that get very bright appear to be selected because they benefit from any symmetry that arises during the development of the solutions.

D.2 Other target based cost functions

With equation 6.7 as the basic definition for target based cost functions the following cost functions have been briefly tested to confirm that they did not behave significantly different from the target based cost function used in Section 6.3.1.

$$C = \sum_{\text{image samples}} |I - T| \quad (\text{D.2})$$

$$C = \sum_{\text{image samples}} (I - T)^4 \quad (\text{D.3})$$

Both equation D.2 and D.3 were found perform roughly similarly to equation 6.8 with the exception that they invariably produced noisier image intensity distributions. They were also at least as sensitive to the setting of the target value as equation 6.8.

Equation D.2 appeared to produced more image intensity “holes” than equation 6.8 and equation D.3 produced distinctly more uneven image intensity distributions than equation 6.8.

Bibliography

- [1] D. Gabor, "A new microscope principle," *Nature*, vol. 161, pp. 777–778, 1948.
- [2] E. N. Leith and J. Upatnieks, "Reconstructed wavefronts and communication theory," *Journal of the Optical Society of America*, vol. 52, no. 10, pp. 1123–1130, 1962.
- [3] E. N. Leith and J. Upatnieks, "Wavefront reconstruction with continuous tone objects," *Journal of the Optical Society of America*, vol. 53, no. 12, pp. 1377–1381, 1963.
- [4] J. W. Goodman, *Introduction to Fourier Optics*. McGraw-Hill, 1968.
- [5] H. Kogelnik and T. Li, "Laser beams and resonators," *Applied Optics*, vol. 5, no. 10, pp. 1550–1567, 1966.
- [6] B. R. Brown and A. W. Lohmann, "Complex spatial filtering with binary masks," *Applied Optics*, vol. 5, no. 6, pp. 967–969, 1966.
- [7] W.-H. Lee and M. O. Greer, "Matched filter optical processor," *Applied Optics*, vol. 13, no. 4, pp. 925–930, 1974.
- [8] M. S. Kim and C. C. Guest, "Simulated annealing algorithm for binary phase only filters in pattern classification," *Applied Optics*, vol. 29, no. 8, pp. 1203–1208, 1990.
- [9] F. Wyrowski, "Digital phase-encoded inverse filter for optical pattern recognition," *Applied Optics*, vol. 30, no. 32, pp. 4650–4667, 1991.
- [10] W. T. Cathey, *Optical Information Processing and Holography*. J. Wiley and Sons, 1978.
- [11] J. G. Walker, E. R. Pike, R. E. Davies, and M. R. Young, "Superresolving scanning optical microscope using holographic optical processing," *Journal of the Optical Society of America A*, vol. 10, no. 1, pp. 59–64, 1993.
- [12] W.-H. Lee, "Computer generated holograms: Techniques and applications," *Progress in Optics*, vol. XVI, pp. 121–229, 1978.
- [13] Riley and Birkett, "A reflection kinoform for use with a carbon dioxide laser," *OPTICA ACTA*, vol. 24, no. 10, pp. 999–1009, 1977.
- [14] J. F. Ready, *Industrial Applications of Lasers*. Academic Press, 1978.

- [15] M. R. Feldman and C. C. Guest, "Iterative encoding of high efficiency holograms for the generation of spot arrays," *JOSA*, reprinted form *optics letters*, vol. 14, no. 10, pp. 479–482, 1989.
- [16] R. G. Hoptroff, P. W. McOwan, T. J. Hall, W. J. Hossack, and R. E. Burge, "Two optimisation approaches to coho design," *Optics Communications*, vol. 73, no. 3, pp. 188–194, 1989.
- [17] J. Mait, "Design of damman gratings for two dimensional, non-seperable, non-centrosymmetric responses," *Optics Letters*, vol. 14, no. 4, pp. 196–198, 1990.
- [18] B. Kress and P. Meyrueis, "Tolerencing and packaging analysis for the optimization of a diffractive element for synchronous planar optical clock distribution," in *Diffractive and Holographic Optics Technology III*, pp. 95–100, SPIE, SPIE, 1996.
- [19] M. T. Eismann, A. M. Tai, and J. N. Cedrquist, "Holographic beamformer designed by an iterative technique," *SPIE*, vol. 1052, no. Holographic optics, pp. 10–18, 1989.
- [20] A. Vasara, J. Turunen, and A. T. Friberg, "Realization of general nondiffracting beams with computer generated holograms," *J. Opt. Soc Am.*, vol. 6, no. 11, pp. 1748–1754, 1989.
- [21] J. Ojeda-Castaneda and L. R. Berriel-Valdos, "Zone plate for arbitrary high focal depth," *Applied Optics*, vol. 29, no. 7, 1990.
- [22] H. Y. Chen and E. G. S. Paige, "Creation of 3-d radiation fields to specification and demonstration using an slm," *Electronics Letters*, vol. 30, no. 9, pp. 735–736, 1994.
- [23] C. B. Scruby and L. E. Drain, *Laser Ultrasonics, Techniques and Applications*. Bristol, UK: Adam Hilger, 1990.
- [24] N. Rykalin, A. Uglov, and A. Kokoru, *Laser Machining and Welding*. Pergamon Press, 1978.
- [25] J. F. Eloy, *Power Lasers*. J. Wiley and Sons, 1987.
- [26] N. C. Gallagher and B. Liu, "Method for computing kinoforms that reduces image reconstruction error," *Applied Optics*, vol. 12, no. 10, pp. 2328–2335, 1973.
- [27] A. F. Gmitrio, P. E. Keller, C. Coleman, and P. D. Maker, "Design and fabrication of multi-level phase holograms for on-axis optical interconnects," in *Diffractive Optics: Design, Fabrication and Applications*, pp. 239–242, OSA, 1994.
- [28] F. Linnane, D. Zhang, M. Clark, and M. G. Somekh, "Surface acoustic wave generation with customized optical beam distributions," in *1996 IEEE Ultrasonics Symposium*, (Piscataway, NJ, USA), IEEE and UFFC, IEEE, 1996.

- [29] M. G. Somekh, F. Linnane, M. Clark, and C. W. See, "Non-contact surface acoustic microscopy using laser ultrasound," *Measurement science and technology*, 1997.
- [30] B. R. Brown and A. W. Lohmann, "Computer generated holograms," in *The Engineering Uses of Holography*, pp. 77–97, Strathclyde University and The National Physical Laboratory, Cambridge University Press, 1968.
- [31] R. G. Canas, R. W. Smith, and A. A. West, "High efficiency, volume and surface relief computer generated diffractive optical elements," *SPIE*, vol. 1136, no. Holographic Optics II, pp. 208–214, 1989.
- [32] D. Kermisch, "Image reconstruction from phase information only," *JOSA*, vol. 60, no. 1, pp. 15–17, 1970.
- [33] P. Vermeulen, E. Barnard, and D. Casasent, "New fresnel cghs for lensless optical systems and thier applications," *SPIE*, vol. 1052, pp. 223–233, 1989.
- [34] S. Weissbach, F. Wyrowski, and O. Bryngdahl, "Coding and quantization of computer generated phase holograms with error diffusion," *SPIE*, vol. 1136, no. Holographic optics II, pp. 226–227, 1989.
- [35] C. Paterson, "Diffractive optical elements with spiral phase dislocations," *Journal of Modern Optics*, vol. 41, no. 4, pp. 757–765, 1994.
- [36] A. W. Lohmann and D. P. Paris, "Binary fraunhofer holograms, generated by computer," *Applied Optics*, vol. 6, no. 10, pp. 1739–1748, 1967.
- [37] C. Paterson and R. Smith, "Higher-order bessel waves produced by axicon-type computer-generated holograms," *Optics Communications*, no. 124, pp. 121–130, 1996.
- [38] R. W. Gerchberg and W. O. Saxton, "Practical algorithm for the determination of phase from image and diffraction plane pictures," *Optik*, vol. 35, no. 2, pp. 237–247, 1972.
- [39] P. M. Hirsch, J. A. Jordan, and L. B. Lesem, "Method of manufacturing wave shaping objects," *U. S. Patent No. 3606515*, 1971.
- [40] P. M. Hirsch, J. A. Jordan, and L. B. Lesem, "Method of making an object dependant diffuser," *U. S. Patent No. 3619022*, 1971.
- [41] H. Faroosh, Y. Fainman, and S. H. Lee, "Algorithm for computation of large size fast fourier transforms in computer-generated holograms by interlaced sampling," *Optical Engineering*, vol. 28, no. 6, pp. 622–628, 1989.
- [42] N. C. Gallagher and D. Sweeney, "Computer generated microwave kinoforms," *Optical Engineering*, vol. 28, no. 6, pp. 599–604, 1989.
- [43] F. Wyrowski and O. Bryngdahl, "Iterative fourier transform algorithm applied to computer holography," *JOSA*, vol. 5, no. 7, pp. 1058–1065, 1988.

- [44] M. Bernhardt, F. Wyrowski, and O. Bryngdahl, "Iterative techniques to integrate different optical functions in a diffractive phase element," *Applied Optics*, vol. 30, no. 32, pp. 4629–4635, 1991.
- [45] B. K. Jennison, D. W. Sweeney, and J. P. Allebach, "Iterative approaches to computer-generated holography," *Optical Engineering*, vol. 28, no. 6, pp. 629–637, 1989.
- [46] M. A. Seldowitz, J. P. Allebach, and D. W. Sweeney, "Synthesis of digital holograms by direct binary search," *Applied Optics*, vol. 26, no. 14, pp. 2788–2798, 1987.
- [47] N. Yoshikawa, M. Itoh, and T. Yatagai, "Quantized phase optimization of two dimensional fourier kinoforms by genetic algorithm," *Optics Letters*, vol. 20, no. 7, pp. 752–754, 1995.
- [48] A. D. Kathman and D. R. Brown, "New techniques for genetic algorithm optimization of diffractive optical elements," *Rochester 1994 Conference Proceedings*, pp. 137–138, 1994.
- [49] E. Aarts and Korst, *Simulated Annealing and Boltzmann Machines*. J. Wiley and Sons, 1989.
- [50] H. H. Szu and R. L. Hartley, "Nonconvex optimization by fast simulated annealing," *Proceedings of the IEEE*, vol. 75, no. 11, 1987.
- [51] S. Kirkpatrick, C. D. G. Jr, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, 1983.
- [52] Otten and V. Ginneken, *The Annealing Algorithm*. Kluwer Academic Publishers, 1989.
- [53] E. Aarts and Korst, *Simulated Annealing and Boltzmann Machines*, pp. 33–52. J. Wiley and Sons, 1989.
- [54] E. Aarts and Korst, *Simulated Annealing and Boltzmann Machines*, p. 58. J. Wiley and Sons, 1989.
- [55] M. P. Dames, R. J. Dowling, P. McKee, and D. Wood, "Design and fabrication of efficient optical elements to generate intensity weighted spot arrays," *Applied Optics*, vol. 30, pp. 2685–2691, 1991.
- [56] B. K. Jennison and J. P. Allebach, "Direct binary search computer generated hologram: an accelerated design technique and measurement of wavfront quality," *SPIE*, vol. 1052, pp. 2–9, 1989.
- [57] M. Clark, "A direct search method for the computer design of holograms," in *4th International Conference on Holographic Systems, Components and Applications*, pp. 96–99, IEE, 1993.
- [58] M. Clark, "A direct search method for the computer design of holograms for the production of arbitrary intensity distributions," in *Diffractive Optics: Design, Fabrication and Applications*, pp. 159–162, OSA, 1994.

- [59] M. Clark and R. Smith, "A direct-search method for the computer design of holograms," *Optics Communications*, vol. 124, no. 1-2, pp. 150–164, 1996.
- [60] J.-Y. Zhuang and O. K. Ersoy, "Fast decimation-in-frequency direct binary search algorithms for synthesis of computer-generated holograms," *Journal of the Optical Society of America A*, vol. 11, no. 1, 1994.
- [61] S. E. Broomfield, M. A. A. Neil, and E. G. S. Paige, "Programmable multiple-level phase modulation that uses ferroelectric liquid-crystal spatial light modulators," *Applied Optics*, vol. 34, no. 29, 1995.
- [62] M. Galem, M. Rossi, and H. Schutz, "Continuous-relief diffractive optical elements for two dimensional array generation," *Applied Optics*, vol. 32, no. 14, pp. 2526–2533, 1993.
- [63] E. Jager, J. Hobfeld, Q. Tang, and T. Tschudi, "Design of a laser scanner for kinoform fabrication," *SPIE*, vol. 1136, no. Holographic Optics II, 1989.
- [64] J. Amako and T. Sonehara, "Kinoform using an electrically controlled birefringent liquid crystal spatial light modulator," *Applied Optics*, vol. 30, no. 32, pp. 4622–4628, 1991.
- [65] E. G. S. Paige and R. H. Scarbrough, "Generation of binary, phase-only, holograms by on-line feedback of output plane intensity," *Rochester 1994 Conference Proceedings*, pp. 247–250, 1994.
- [66] E. Carcole, J. Campos, I. Juvells, and S. Borsch, "Diffraction efficiency of low-resolution fresnel encoded lenses," *Applied Optics*, vol. 33, no. 29, 1994.
- [67] Born and Wolf, *Principles of Optics*. Pergamon Press, 1964.
- [68] V. E. Levashov and A. V. Vinogradov, "Analytical theory of zone plate efficiency," *Physical Review E*, vol. 49, pp. 5797–5803, 1994.
- [69] D. W. Prather, M. S. Mirotznik, and J. Mait, "Design of subwavelength diffractive optical elements using a hybrid finite element-boundary element method," in *Diffractive and Holographic Optics III*, pp. 14–23, SPIE, SPIE, 1996.
- [70] M. S. Mirotznik, D. W. Prather, and J. N. Mait, "Hybrid finite element-boundary element method for vector modeling diffractive optical elements," in *Diffractive and Holographic Optics III*, pp. 2–13, SPIE, SPIE, 1996.