

IMPROVING THE RESOLUTION OF DIFFRACTION
PATTERNS FROM MANY LOW RESOLUTION
RECORDINGS

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND
ELECTRONICS ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCES
OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF
MASTER OF SCIENCE

By

Veysel Yücesoy

September 2010

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Levent Onural (Supervisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Gözde Bozdağı Akar

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Dr. Tarık Reyhan

Approved for the Institute of Engineering and Sciences:

Prof. Dr. Levent Onural
Director of Institute of Engineering and Sciences

ABSTRACT

IMPROVING THE RESOLUTION OF DIFFRACTION PATTERNS FROM MANY LOW RESOLUTION RECORDINGS

Veysel Yücesoy

M.S. in Electrical and Electronics Engineering

Supervisor: Prof. Dr. Levent Onural

September 2010

Holography attempts to record and reconstruct wave fields. The resolution limitation of the recording equipments causes some problems in the reconstruction process. An automatic method for the registration and stitching of low resolution diffraction patterns to form a higher resolution one is proposed. There is no prior knowledge about the 3D position of the object in the recordings and it is assumed that there is only one particle in the object field. The method uses Wigner transform, Canny edge detection and Hough transform to register the patterns, and some additional iterative methods depending on the local variance of the reconstructed patterns to stitch them. The performance of the overall system is evaluated against object radius, noise in the original pattern, recording noise and presence of multiple particles in the object field by computer simulations.

Keywords: Digital Holography, Registration, Stitching, Diffraction Patterns, Superresolution

ÖZET

BİRÇOK DÜŞÜK ÇÖZÜNÜRLÜKLÜ KAYIT KULLANARAK KIRINIM DESENLERİNİN ÇÖZÜNÜRLÜĞÜNÜN ARTTIRILMASI

Veysel Yücesoy

Elektrik ve Elektronik Mühendisliği Bölümü Yüksek Lisans

Tez Yöneticisi: Prof. Dr. Levent Onural

Eylül 2010

Holografi dalga alanlarını kaydetmeye ve geri çalmaya çalışır. Kaydedici aygıtların çözünürlük sınırları nedeniyle ışık dağılımını geri çalma aşamasında bazı problemler ortaya çıkmaktadır. Düşük çözünürlükteki kırınım desenlerinin otomatik olarak yerlerinin belirlenmesini ve birleştirilerek yüksek çözünürlüklü bir kırınım deseni elde edilmesini amaçlayan bir teknik geliştirilmiştir. Kayıtlardaki nesnelerin üç boyutlu yerleri hakkında hiçbir ön bilgi bulunmamasında ve nesnenin tek parçacıktan oluştuğu varsayılmaktadır. Sunulan teknik, cismin değişik kayıtlardaki yerini bulmak için Wigner dönüşümünü, Canny kenar bulma yöntemini ve Hough dönüşümünü kullanmakta ve dokuları birleştirmek için de yerel değişirliği baz alan bazı yinelemeli metotlar içermektedir. Önerilen tekniğin değişen nesne büyüklüğü, orjinal desende oluşan gürültü, kayıt ederken oluşan gürültü ve nesne alanında birden fazla parçacık bulunması durumlarına karşı başarıyı bilgisayar benzetimleri ile değerlendirilmiştir.

Anahtar Kelimeler: Sayısal Holografi, Kırınım Desenleri, Süper Çözünürlük

ACKNOWLEDGMENTS

First, I would like to thank to my supervisor Prof.Dr.Levent Onural for his precious guidance and important suggestions during my graduate study.

I also thank to other graduate students who work on the same subject due to their contributions as Dr.Gökhan Bora Esmer being the leading one.

I do not know how to thank to my family especially to my mother, father and uncles who had been encouraging and supporting me from my birth, I owe and love you all a lot, thanks. You are a gift from my fate.

And of course my dearest friends, whom I really love. I am grateful to Doruk Tunaoglu for always supporting me and making me stand when I was to fall, to Batın Şimşek for being with me in any case at any time, relieving and guiding me, to Hande Dağlı for her true endless fellowship, trusting me unconditionally and easing my life, to Çağlar Erkan for thinking me among his own worries and being honest in any case, and to Mert Giray İlhan, Yiğit Gençay, Erdem Kumkale, Ali Nehir Yücel, Çağan Yıldırım, Alp Oğuz and to all other friends for their friendship, being curious about me and helping me. You are all with me even if there are roads between us. I would be lacking in your absence. I also thank my colleagues at Aselsan for being understanding and helpful.

I want to acknowledge the support of TÜBİTAK (The Scientific and Technological Research Council of Turkey) BİDEP 2210 graduate student fellowship.

Contents

1	Introduction	1
2	Superresolution - Synthetic Aperture Techniques in Digital Holography	6
2.1	Single Camera Approaches	8
2.2	Multiple Camera Approaches	10
3	Preliminaries About Diffraction Patterns	11
3.1	Optics Basics	11
3.1.1	Plane Wave	12
3.1.2	Complex Wavefunction and Monochromatic Waves	13
3.1.3	Transfer Function of Free Space	14
3.1.4	Fresnel Approximation of the Transfer Function	16
3.2	Diffraction of Light	18
4	Wigner Transform	22

4.1	Properties of the Discrete Wigner Distribution	23
4.2	Wigner Analysis of Diffraction Patterns	25
4.2.1	1D Diffraction Patterns	25
4.2.2	2D Diffraction Patterns	27
4.2.3	Analysis of Wigner Transforms	28
5	Registration and Stitching	35
5.1	Registration of Digital Holograms	35
5.1.1	Wigner Analysis	38
5.1.2	Iterative Methods for Depth Estimation	44
5.2	Stitching of Digital Holograms	50
5.3	Overall System	57
6	Simulation Results	65
6.1	A Complete Registration and Stitching	65
6.2	Other Simulation Results	83
6.2.1	Radius of the Object	83
6.2.2	Noise Simulations	85
6.2.3	Presence of Multiple Particles	91
6.2.4	Radius of the Object and Noisy Recording	100
7	Conclusion	102

APPENDIX	107
A General Information About Patterns	107
B Discrete Fourier Transform	108
C Input File Generation	110
D Propagation of Diffraction Patterns	111
E Discrete Wigner Transform	113
F Canny Edge Detection	115
F.1 Smoothing and enhancement	115
F.2 Detection	118
F.3 Edge thinning	119
F.4 Thresholding	120
G Hough Transform	122

List of Figures

3.1	The diagram explains the linear system formed by the free space .	16
3.2	Aperture function (i.e. object function)	19
3.3	An example of a diffraction pattern	21
4.1	Wigner transform a chirp function	30
4.2	The resulting binary image when Canny edge detection is applied to the Figure 4.1	32
4.3	The detected lines by edge detection and Hough transform are shown on the initial Wigner transform pattern.	34
5.1	The way by which the low resolution diffraction patterns are recorded (or calculated for simulation purposes)	36
5.2	Some different diffraction patterns generated by computer	37
5.3	This is an example diffraction pattern.	39
5.4	The horizontal and vertical slices of the pattern which are used to form Wigner transforms in the following parts.	40
5.5	The plots of the slices of the pattern shown in Figure 5.4	41

5.6	Wigner transform of the column function	42
5.7	Wigner transform of the row function	43
5.8	The result of edge detection on column and row functions.	44
5.9	The lines detected by edge detection and Hough transform	44
5.10	The reconstructed pattern only with the estimated parameters	45
5.11	Result of the variance filter in the first step	50
5.12	Result of the variance filter in the second step	51
5.13	Result of the variance filter in the third step	51
5.14	The final reconstruction after the depth iteration process.	52
5.15	Parallel plane stitching back and forth	56
5.16	Zeroth order interpolator	57
5.17	The pattern generated when there is a mismatch in x,y positions of the diffraction patterns.	58
5.18	The variance filter result of the mismatch problem.	59
5.19	The final reconstruction of the iteration process.	60
5.20	The block diagram of the overall system.	61
6.1	Low resolution pattern no.1.	67
6.2	Edge detection and Hough transform results of the low resolution pattern no.1	68
6.3	Reconstruction of low resolution pattern no.1	69

6.4	Low resolution pattern no.2.	70
6.5	Edge detection and Hough transform results of the low resolution pattern no.2	71
6.6	Reconstruction of low resolution pattern no.2	72
6.7	Low resolution pattern no.3.	73
6.8	Edge detection and Hough transform results of the low resolution pattern no.3	74
6.9	Reconstruction of low resolution pattern no.3	75
6.10	Low resolution pattern no.4.	76
6.11	Edge detection and Hough transform results of the low resolution pattern no.4	77
6.12	Reconstruction of low resolution pattern no.4	78
6.13	Low resolution pattern no.5.	79
6.14	Edge detection and Hough transform results of the low resolution pattern no.5	80
6.15	Reconstruction of low resolution pattern no.5	81
6.16	The results from the stitching process	82
6.17	The effect of the object radius on Wigner transform	85
6.18	Examples of the intensity of initial noisy fields	87
6.19	Initial pattern of the simulation 27 from Table 6.6	93
6.20	Initial pattern of the simulation 13 from Table 6.6	94

6.21	Initial pattern of the simulation 9 from Table 6.7	96
6.22	Initial pattern of the simulation 32 from Table 6.7	97
6.23	Initial pattern of the simulation 35 from Table 6.8	99
A.1	Axis properties of the figures within thesis	107
G.1	This figure explains the relation between (x, y) coordinates and (θ, ρ) coordinates in Hough transform.	123

List of Tables

6.1	The results of the object radius simulations	84
6.2	The results of the Gaussian noise in the wave field	87
6.3	The results of the uniformly distributed noise in the wave field	88
6.4	The results of the Gaussian recording noise simulations	89
6.5	The results of the uniformly distributed recording noise simulations	90
6.6	The results of the multiple particle simulations with one extra particle between object and recording equipment	92
6.7	The results of the multiple particle simulations with one extra particle behind object with respect to recording equipment	95
6.8	The results of the multiple particle simulations with two extra particles, one is strictly behind object with respect to recording equipment and the other is randomly placed in depth	98
6.9	The results of the simulations with respect to both radius and noise changes	101

Dedicated to my grandmother

Chapter 1

Introduction

Image (video) capturing and displaying is a progressive research area and it is on the verge of making a transition from 2D to 3D. Nowadays, most of the movie theaters view 3D movies besides 2D movies and 3D televisions and cameras became commercially available in the last decade. As this transition progresses (1) consumers will be able to shop from internet while looking at the 3D image of an object from different views, (2) doctors will benefit from 3D views of magnetic resonance images (MRI) and tomography, (3) teleoperation, especially telesurgery, will become more efficient by supplying a 3D view of the scene, (4) artists will make 3D paintings which will open a vast area for modern art and its up to human imagination to extend this list.

One approach for capturing the 3D view of a scene is stereoscopic imaging which basically records two 2D images of a scene using two cameras parallel to each other and separated by the average distance between a human's eyes. Then these images are displayed on a screen and filtered by a special pair of glasses so that each eye of a human will only see the corresponding image. There are various types of viewing and filtering these images. Some of these systems are:

1. Traditional approaches view right and left images with a different color and filter the images accordingly. This approach annoys the human brain and is not efficient in terms of capturing the color.
2. IMAX Technology reflects the right and left images on the screen by using two different light sources with perpendicular polarization and filters this image with a pair of properly polarized glasses.
3. LED TV's display right and left images one after each other and filters this image by using a pair of active LCD glasses. Each side of this pair of glasses can be opaque or transparent according to the synchronization between the TV and the glasses.

Note that, stereoscopic imaging creates a 3D feeling in a human's mind by showing the appropriate 2D images to his/her eyes. As a result, it is user dependent (i.e. it can not be used on a bird without changing the display and filtering technology) and it does not really capture the 3D data (i.e. you can not see an object from a different angle if a stereoscopic image is not obtained from that angle).

An alternative to stereoscopic imaging for 3D capture and display problem is holography which aims to capture the 3D data of a scene by recording both the amplitude and the phase properties of the light distribution in the scene at the aperture. The diffraction pattern caused by the object under a coherent reference beam (i.e. a laser light source) is stored on a photographic film or on a digital sensor in order to capture this 3D data. Although the resulting data is stored in a 2D array, with proper signal processing and equipment the same light distribution of the scene can be reconstructed as if emanating from the aperture using which the hologram were recorded.

Holography is different than stereoscopic imaging in many aspects:

1. It requires special techniques and equipments for capturing a 3D image instead of two 2D cameras.
2. It reconstructs the real 3D view by reconstructing the physical light distribution instead of deceiving the brain with two 2D images.
3. It does not require any glasses for filtering.

As a result, holography provides a continuous change in the view as a human changes his/her point of view of the scene, a viewer can focus to different points, and its performance does not depend on the properties of the visual capturing system of a viewer. On the other hand, stereoscopic imaging can only provide discrete changes in the view as a human changes his/her point of view of the scene (motion parallax problem) if there is not eye tracking and continuous rendering for a single viewer in the system, a viewer is restricted to the focus point of the cameras (accommodation problem) and it is highly dependent on the visual capturing system of the viewer (binocular disparity problem).

An important aspect in the capturing process of holography is the resolution of the recording equipment. If the resolution of the equipment is low, then image taken from a single position will not capture all frequency components of the light distribution. If low frequencies are omitted then most of the information about the scene is lost and if high frequencies are omitted then sharpness of the reconstructed 3D image and the range for the angle of effective view is degraded. If this range is too small then reconstruction of the 3D image is impractical since it becomes difficult for a viewer to place both of his eyes in the effective viewing region where small head movements will take him outside this region. Techniques and equipments which can capture the 3D data in high resolution should be developed in order to overcome aforementioned problems. Note that this is a capturing problem which degrades the performance of the display process. Reader is directed to [1], [2], [3], [4], [5] and [6] for a detailed review of display techniques and problems in holography.

The resolution problem is present in photography, as well, where it is solved by applying super-resolution techniques. These techniques enhance the resolution of a photograph by processing and combining totally or partially overlapped photographs of the same scene with sub-pixel accuracy. A similar method is available in holography [7], [8], [9], [10], [11] where the constraint of overlapping the images is not necessary if the light distribution does not change with respect to time while different 3D images are captured.

In this thesis, our aim is to develop a method for enhancing the resolution of a hologram by processing and combining different holograms of the same scene. In order to ease our problem, we assume that the hologram patterns are captured from the same diffraction field, they are parallel to each other (i.e. recording equipment is not tilted between recordings) and there is only one object in the scene. We do not make any assumptions nor have any prior knowledge about the 3D positions of the recording equipment. Wigner transform, Canny edge detection and Hough transform are used in registration process. Local variance is used as the sharpness parameter during the stitching process. Stitching is the part in which the preregistered recordings are combined to form a higher resolution pattern. The aim of the stitching is to involve all possible information coming from different low resolution recordings into a higher resolution grid. Sampling and interpolation problems are solved within the requirements of this thesis and these are explained in the relevant sections. The proposed method is evaluated by simulations under different conditions such as noisy recordings, different sized objects and with scenes involving multiple objects (although we assumed that there is a single object).

Rest of the thesis is organized as follows: Chapter 2 is about superresolution techniques in general and how it is applicable to digital holography. Chapter 3 reveals the preliminaries about the optical properties of the diffraction patterns and their mathematical representations. Chapter 4 restates the basic definitions

and properties of the Wigner transform and explains how it is applicable to diffraction patterns. This chapter also gives basic information about edge detection and Hough transform. In Chapter 5, all the steps of the registration and stitching processes are explained and all the relevant information about their mechanism is stated. Chapter 6 contains the simulation results of the proposed method under various scenarios. Finally, Chapter 7 concludes the thesis and gives some brief ideas about possible future work.

Chapter 2

Superresolution - Synthetic Aperture Techniques in Digital Holography

Synthetic aperture refers to a collection of techniques which aim to combine a number of signals in an appropriate way to form a high resolution signal. It was first proposed by Sir Martin Ryle and co-workers from the Radio Astronomy Group at Cambridge University [12] in order to use in radio astronomy. Sir Martin Ryle and Tony Hewish jointly received a Nobel Prize for their contributions to radio interferometry [13]. This technique formed a basis to other fields such as synthetic aperture radar (SAR) [14], synthetic aperture sonar [15] and synthetic aperture magnetometry [16].

The general idea behind the superresolution is to smartly combine the information from a number of low resolution recordings to have a final high resolution recording. It may also be possible to make this upsampling only with one low resolution recording. This is called interpolation. Interpolation is the smart way

of estimating the mid values from the recorded points. It produces artificial values depending on the known values of the function. However, superresolution tries to make this upsampling by making use of the real data coming from other low resolution recordings. Superresolution is used to enhance the resolution of the imaging sensor when it does not have the desired resolution.

Superresolution is applicable to both videos and still images. It is generally used with a motion detection algorithm to register the low resolution recordings. The registration process is important in superresolution because the sharpness of the final high resolution recording basically depends on the position estimation accuracy of the features in the low resolution recordings. In the stitching process, the registered low resolution recordings are combined with a suitable merging algorithm. The merging algorithm can both be in frequency or time(space) domain depending on the purpose of the whole algorithm and the features that are extracted from low resolution recordings. Also some iterative techniques may be used in stitching process in order to correct any possible errors caused by registration. The sharpness of the final high resolution image is the basic property of interest which is desired to be maximised at the end of the entire process.

In daily life photography or video recording, overlapping recordings are required for superresolution. The low resolution recordings can be nearly totally overlapped if the aim of superresolution is to enhance only the resolution of the low resolution recording. In this case, subpixel shifts of the same data give necessary information to enhance the resolution. The low resolution recordings can be partially overlapped if the aim is also to enlarge the recorded object field in the final high resolution image. In this case, motion detection is crucial to find the same features in different recordings in order to be able to register the low resolution recordings.

It is also possible to use this technique in digital holography in order to improve the resolution and the viewing angle in reconstruction process. The idea is a bit different than it is in daily life photography. There is no need for overlapped recordings in order to register them. Since in digital holography, the wave field generated by the diffraction of the light from the object field is recorded, it has to be consistent with all of the recordings unless the wavefield is changed during recording period. So it is possible to register and stitch non-overlapping recordings in digital holography in order to enhance the final resolution (In this paragraph overlapping is used to express two different recordings of the same scene having some parts in common. This definition is used from a photography point of view. From holography point of view, any recorded patterns of the same light distribution can be assumed to overlap since they carry consistent information about the object field). This enhancement also yields an enhancement in the viewing angle in the reconstruction process. There has been some major contributions to this field in digital holography. The following subsections will give basic information about these techniques from two basic categories: (i) the capture is done by a single camera where the object wave field is moved across the camera, (ii) the capture is done by a number of cameras simultaneously where the object wave field is stable.

2.1 Single Camera Approaches

Massig has proposed a simple technique in 2002 [7]. In his method, a camera is moved along a plane and several captures are taken. The plane in which the camera is moved is a perpendicular plane to the optical axis of the system formed by object and the camera in his technique. Massig has enlarged the camera to have a larger angle of the object beam. This will yield an increased perspective in the reconstruction. His method is for off-axis lensless Fourier digital holography with a spherical reference beam [7]. The advantage of this setup is that object

does not have to be necessarily moved while the camera is moved to capture different patterns.

A successful way for synthetic aperture was proposed by Binet et al. in 2002 [8]. He used a static camera placed at the far field [8]. The object is rotated by small angles while the camera remains still and captures the patterns. The angles should be small in order not to lose the correlation between speckle patterns. If the correlation is lost then the registration and stitching may become impossible between successive captures. The major disadvantage of this method is this correlation issue since a large number of patterns with a small angle difference is required to increase the aperture.

A modification to Binet's method was proposed in 2007 by Hennelly [9]. Hennelly placed a mirror in the path of the object beam in order to direct different parts of the object beam to the camera to get rid of the decorrelation of the speckle patterns. Since in Hennelly's method the mirror rotates instead of the object itself, it is possible to overcome the speckle by decorrelation. Due to additional mirror the stitching is required to be done both in the spatial frequency domain and in the space domain.

Di et al. proposed a different method to build a 2D hologram from a linear CCD [10]. They used a precision translational stage to move the linear CCD in y direction. They did not use any type of a post analysis to register the captured patterns. They used a computer to control the precision translational stage and captured 5000 different patterns with the linear CCD. At the end of capture process which lasts for 51s, they orderly take all the captured patterns together to form a 5000×5000 pixel digital hologram [10]. They used a lensless Fourier transform recording method. The disadvantage of this method is the total capture time which only allows the capture of static fields.

2.2 Multiple Camera Approaches

A new method related to multiple camera approaches was proposed by Kreis and Schlüter [11] in 2007. They use a spherical reference beam and Fourier holography. They have showed that the PSF of two camera system is equivalent to the PSF of a single camera system multiplied by a cosine whose frequency is dependent on the separation of the cameras [11]. Authors have proposed two different methods to reconstruct the object field [11]. In both methods it is very important to know relative positions of the CCDs. The position is determined by recording a circular fringe pattern on both CCDs and making a least squares fit on these patterns to find the relative positions as explained in [11].

Chapter 3

Preliminaries About Diffraction Patterns

3.1 Optics Basics

Fourier optics is an approach in which the light is assumed to be a composition of harmonic functions. Such an approach is especially useful when the optical system is linear and shift invariant. This approach is simply the same as it is in classical signal processing. In classical signal processing, to be able to calculate the response of a linear and shift invariant system to an arbitrary input $f(t)$, the input function is characterized as a weighted sum (integral) of harmonic functions (called as Fourier transform of the input signal $F(\omega)$) and the linear and shift invariant system is characterized by its transfer function $H(\omega)$. The simple relation between the input signal $f(t)$ and the output signal $g(t)$ after passing the linear system $H(\omega)$ is governed by

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \exp(-j\omega t) dt = F\{f(t)\} \quad (3.1)$$

$$H(\omega) = \int_{-\infty}^{\infty} h(t) \exp(-j\omega t) dt = F\{h(t)\} \quad (3.2)$$

$$G(\omega) = \int_{-\infty}^{\infty} g(t) \exp(-j\omega t) dt = F\{g(t)\} \quad (3.3)$$

$$g(t) = f(t) * h(t) \quad (3.4)$$

$$G(\omega) = F(\omega)H(\omega) \quad (3.5)$$

As it is seen from Equations 3.4 and 3.5, the convolution relation between the input and output functions of time becomes a simple relation of multiplication in the Fourier domain. This simple relationship holds for any higher dimensional case, as well.

In wave optics, propagating light waves are four dimensional signals which have to be a solution to the wave equation

$$\nabla^2 u - \frac{1}{c^2} \frac{\partial^2 u}{\partial t^2} = 0 \quad (3.6)$$

where u is the propagating wave, c is the speed of light in vacuum and ∇^2 is the Laplacian operator given as

$$\nabla^2 = (\partial^2/\partial x^2) + (\partial^2/\partial y^2) + (\partial^2/\partial z^2). \quad (3.7)$$

.

3.1.1 Plane Wave

The plane wave is one of the elementary waves and has the complex amplitude

$$U(\mathbf{r}) = A \exp(j\mathbf{k}^T \mathbf{r}) \quad (3.8)$$

where A is the complex constant called the complex envelope and $\mathbf{k} = (k_x, k_y, k_z)$ is called the wavevector where the units of k_x, k_y, k_z is radians/meter. The wavefronts (i.e. equiphase lines) of this wave obeys an equation describing parallel

planes perpendicular to the wavevector \mathbf{k} . These planes are separated by a distance $\lambda = 2\pi/k$ hence

$$\lambda = \frac{c}{\nu} \quad (3.9)$$

is called the wavelength. The plane wave has a constant intensity everywhere in space so it carries infinite power. This wave is an idealization since it exists everywhere and for every time instant. However, as it is in 1D case, this ideal harmonic wave family forms a basis for all other waves which means it is possible to write any wave function as a weighted sum of different plane waves (Fourier transform).

3.1.2 Complex Wavefunction and Monochromatic Waves

A valid complex monochromatic field is represented by Equation 3.10 which consist of propagating waves.

$$U(\mathbf{r}) = \int_{k_x^2 + k_y^2 \leq k} A(\mathbf{k}) \exp(j\mathbf{k}^T \mathbf{r}) d\mathbf{k} \quad (3.10)$$

where

$$\begin{aligned} \mathbf{k} &= [k_x, k_y, k_z]^T \\ k_z &= \sqrt{k^2 - k_x^2 - k_y^2} \end{aligned} \quad (3.11)$$

Hence, it is also possible to write complex monochromatic wave function as described in Equation 3.12 since it is convenient to show real functions in terms of complex functions.

$$U(\mathbf{r}, t) = \int A(\mathbf{k}) \exp[j(\mathbf{k}^T \mathbf{r} + 2\pi\nu t)] d\mathbf{k} \quad (3.12)$$

By using Equation 3.10 it is possible to rewrite Equation 3.12 as

$$U(\mathbf{r}, t) = U(\mathbf{r}) \exp(j2\pi\nu t) \quad (3.13)$$

Since $U(\mathbf{r})$ is a position dependent complex field it is convenient to write it in the following form

$$U(\mathbf{r}) = a(\mathbf{r}) \exp[j\phi(\mathbf{r})] \quad (3.14)$$

Hence, complex monochromatic wavefunction is $U(\mathbf{r}, t) = a(\mathbf{r}) \exp[j\phi(\mathbf{r})] \exp[j2\pi\nu t]$.

So it is also possible to write a valid monochromatic wave function as

$$u(\mathbf{r}, t) = \text{Re}\{U(\mathbf{r}, t)\} = a(\mathbf{r}) \cos[\phi(\mathbf{r}) + j2\pi\nu t] \quad (3.15)$$

where

$$\begin{aligned} a(\mathbf{r}) &= \textit{amplitude} \\ \phi(\mathbf{r}) &= \textit{phase} \\ \nu &= \textit{frequency (cycles/s or Hz)} \\ \omega &= 2\pi\nu = \textit{angular frequency (radians/s)} \end{aligned} \quad (3.16)$$

and $U(\mathbf{r})$ satisfies the Equation 3.10.

The amplitude and the phase may be position dependent, however, the wavefunction is a harmonic function of time with frequency ν everywhere.

3.1.3 Transfer Function of Free Space

In order to be able to calculate the diffraction pattern at a distance d from the object plane, we need to have an equation governing the propagation of light in free space. Let us assume two hypothetical parallel planes with d distance between them. Furthermore, let us assume that one of the planes is at $z = 0$ position and the other is at $z = d$ position. We need to examine the propagation of a monochromatic light (optical wave) of wavelength λ and complex amplitude $U(x, y, z)$ in free space. Let us call the plane at $z = 0$ the input plane and the plane at $z = d$ the output plane of the total system. From classical signal processing point of view, there should be a relation between the output and input of this system which is called as the **transfer function of the free space**. To be able to calculate it, we first need to define the input and output of the system.

Let us call $f(x, y) = U(x, y, 0)$ the input and $g(x, y) = U(x, y, d)$ the output of the system.

Again referring to classical signal processing, if the system is linear and shift invariant, then its input-output relationship can be expressed by its impulse response. Since the complex amplitude has to obey the **linear** wave equation given by Equation 3.6 at every position and time, the system is then immediately a linear system. Shift invariency is also satisfied by the system because since it is freespace it is by nature invariance to the displacement of coordinate system. Hence, it is enough to find the impulse response of the system $h(x, y)$ to characterize the input-output relation of free space or equivalently its Fourier transform which is also called as the transfer function of the system, $H(\nu_x, \nu_y)$.

As explained for one dimensional case, in Equations 3.4 and 3.5 the transfer function of the system is the factor by which a harmonic input function should be multiplied to yield the output harmonic function of the same frequency. So it is possible to write this relation also in 2D case as

$$G(\nu_x, \nu_y) = H(\nu_x, \nu_y)F(\nu_x, \nu_y) \quad (3.17)$$

So, the transfer function $H(\nu_x, \nu_y)$ is simply the ratio of the input and output functions in the frequency domain.

Let us consider a harmonic function input of $f(x, y) = A \exp[j2\pi(\nu_x x + \nu_y y)]$. This input function corresponds to a plane wave of complex amplitude

$$U(x, y, z) = A \exp[j(k_x x + k_y y + k_z z)] \quad (3.18)$$

everywhere in space where $k_x = 2\pi\nu_x$ and $k_y = 2\pi\nu_y$. Hence we can write input and output functions as

$$f(x, y) = U(x, y, 0) = A \exp[j(k_x x + k_y y)] \quad (3.19)$$

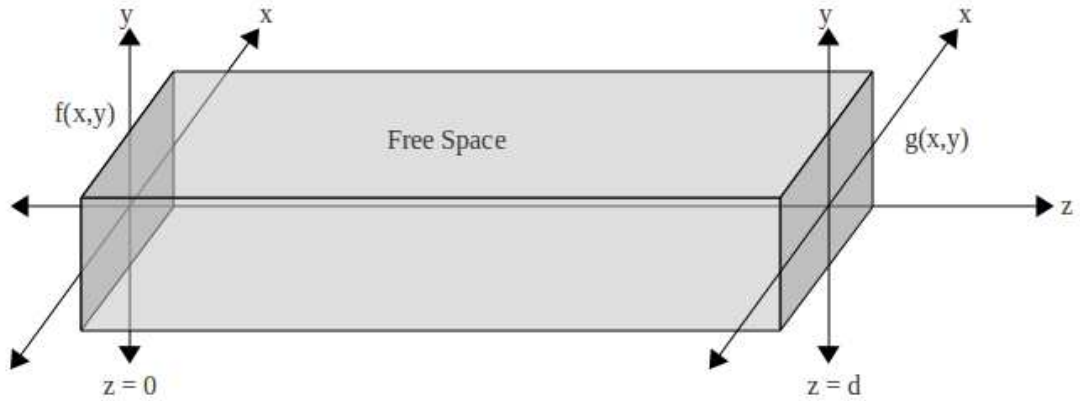


Figure 3.1: The diagram explains the linear system formed by the free space

$$g(x, y) = U(x, y, d) = A \exp[j(k_x x + k_y y + k_z d)] \quad (3.20)$$

and the ratio of input-output as

$$\frac{g(x, y)}{f(x, y)} = \frac{U(x, y, d)}{U(x, y, 0)} = A \exp[jk_z d] \quad (3.21)$$

From equation 3.18

$$k_z = (k^2 - k_x^2 - k_y^2)^{1/2} = 2\pi \left(\frac{1}{\lambda^2} - \nu_x^2 - \nu_y^2 \right)^{1/2} \quad (3.22)$$

Now, we can rewrite the ratio given in equation 3.21 as

$$H(\nu_x, \nu_y) = \exp \left[j2\pi \left(\frac{1}{\lambda^2} - \nu_x^2 - \nu_y^2 \right)^{1/2} d \right] \quad (3.23)$$

The Equation 3.23 is called the transfer function of the free space. This transfer function governs all the propagation of monochromatic light waves in free space.

3.1.4 Fresnel Approximation of the Transfer Function

If we have a bandlimited input function $f(x, y)$ as it only contains spatial frequencies that are smaller than the cutoff frequency $1/\lambda$ so that $\nu_x^2 + \nu_y^2 \ll 1/\lambda^2$,

it may be possible to simplify the equation 3.23 further. Then the propagating wave is a **paraxial wave** which makes small angles with the optical axis. These angles are defined as $\sin(\theta_x) = \lambda\nu_x$ and $\sin(\theta_y) = \lambda\nu_y$. Since the wave is a paraxial wave and it makes small angles with the optical axis, the assumption $\sin(x) = x$ is valid for all these angles. So it is possible to rewrite the equations above with the given assumption.

$$\begin{aligned}\sin(\theta_x) &\rightarrow \theta_x = \lambda\nu_x \\ \sin(\theta_y) &\rightarrow \theta_y = \lambda\nu_y\end{aligned}\tag{3.24}$$

Then the phase factor in Equation 3.23 can be rewritten by a Taylor series expansion as

$$\begin{aligned}2\pi \left(\frac{1}{\lambda^2} - \nu_x^2 - \nu_y^2\right)^{1/2} d &= 2\pi \frac{d}{\lambda} (1 - \theta^2)^{1/2} \\ &= 2\pi \frac{d}{\lambda} \left(1 - \frac{\theta^2}{2} + \text{Higher Order Terms}\right)\end{aligned}\tag{3.25}$$

where $\theta^2 = \lambda^2(\nu_x^2 + \nu_y^2)$.

Neglecting the third and all other higher order terms, the equation 3.23 simplifies to

$$H(\nu_x, \nu_y) = H_0 \exp[-j\pi\lambda d(\nu_x^2 + \nu_y^2)]\tag{3.26}$$

where $H_0 = \exp\left(j\frac{2\pi d}{\lambda}\right)$.

Equation 3.26 shows the Fresnel approximated transfer function of the free space. A generally accepted validity condition for this approximation is that the third term in Equation 3.25 should stay small.

At this point, it is defined that the output wave of the free space propagation is calculated as the inverse Fourier transform of the multiplication of Fourier transform of the input wave and the Fresnel transfer function of the free space given by Equation 3.23 under the validity condition.

3.2 Diffraction of Light

When an optical wave is transmitted through a mask, the intensity of the wave after passing this mask is called the diffraction pattern. This mask can be simply an aperture function (i.e. a little slit on an opaque screen) or any kind of complex function (i.e. a mask which alters both the amplitude and the phase of the passing wave). For the sake of simplicity, we will concentrate on the aperture functions. If the ray optics postulates were enough to characterize the behaviour of the light, then the diffraction pattern should be a perfect shadow of the aperture. However, due to wave nature of the light, the diffraction pattern is a function of aperture function, wavelength of the light and distance to the aperture. The simplest idea to calculate the diffraction pattern after an aperture states that the light distribution just after the aperture is completely masked by the aperture shape. More mathematically if we have an aperture function of the type

$$p(x, y) = \begin{cases} 1 & , \text{ inside the aperture} \\ 0 & , \text{ outside the aperture} \end{cases} \quad (3.27)$$

and if we assume that the $U(x, y)$ and $f(x, y)$ are the complex amplitude of the wave just to the left and right of the opaque screen then according to this simple idea

$$f(x, y) = U(x, y)p(x, y) \quad (3.28)$$

as shown in the Figure 3.2. If we call $g(x, y)$ as the output complex amplitude of the system, it is possible to calculate it as described by Equation 3.26 under the validity condition. Hence, $g(x, y)$ is known as the Fresnel diffraction pattern, as free space propagation is held by Fresnel approximation.

If we make use of the inverse Fourier relation between input and output functions and the relation between the aperture function and input function given by the Equation 3.28 we can write $g(x, y)$ at a distance d when the incident wave

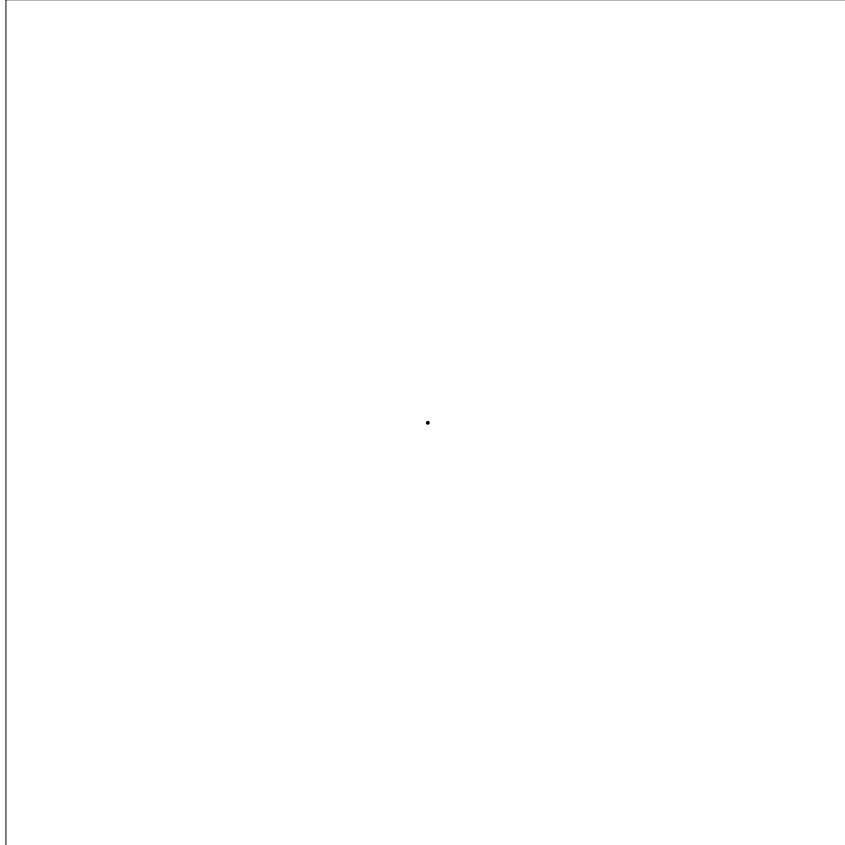


Figure 3.2: Aperture function (i.e. object function). The generation details of this figure is given in Appendix C. The vertical axis is y -axis going from 1 to 1024 from top to down and horizontal axis is x -axis going from 1 to 1024 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0\ 255]$, 255 being the brightest pixel in the figure. Black lines are used to differentiate the figure from the rest of the white sheet, they are not a part of original figure.

is a plane wave, as

$$g(x, y) = \frac{1}{j\lambda d} \int \int_{-\infty}^{\infty} p(x', y') \exp \left[-j\pi \frac{(x - x')^2 + (y - y')^2}{\lambda d} \right] dx' dy' \quad (3.29)$$

It makes the interpretation of the equation easier to normalise all distances with $(\lambda d)^{1/2}$ as a unit of distance so applying $X = x/(\lambda d)^{1/2}$ and $X' = x'/(\lambda d)^{1/2}$ the above equation simplifies to

$$g(X, Y) = -j \int \int_{-\infty}^{\infty} p(X', Y') \exp\{-j\pi[(X - X')^2 + (Y - Y')^2]\} dX' dY' \quad (3.30)$$

This integral is simply a convolution of $p(X, Y)$ and $\exp[-j\pi(X^2 + Y^2)]$ and this equation governs all the relations of the diffraction patterns and the object functions throughout this thesis statement. Figure 3.3 shows a diffraction pattern calculated by the discrete methods explained in Appendix D. The object function (aperture function) is generated by the methods explained in Appendix C with the parameters $N = M = 1024$, $X_c = Y_c = 512$ and $r = 4.5$. Then the methods in Appendix D are used with parameters $\lambda = 600nm$, $X = 10\lambda$, $N = 1024$ and $Z = NX^2/\lambda$. The real part of the resulting matrix is shown in Figure 3.3.

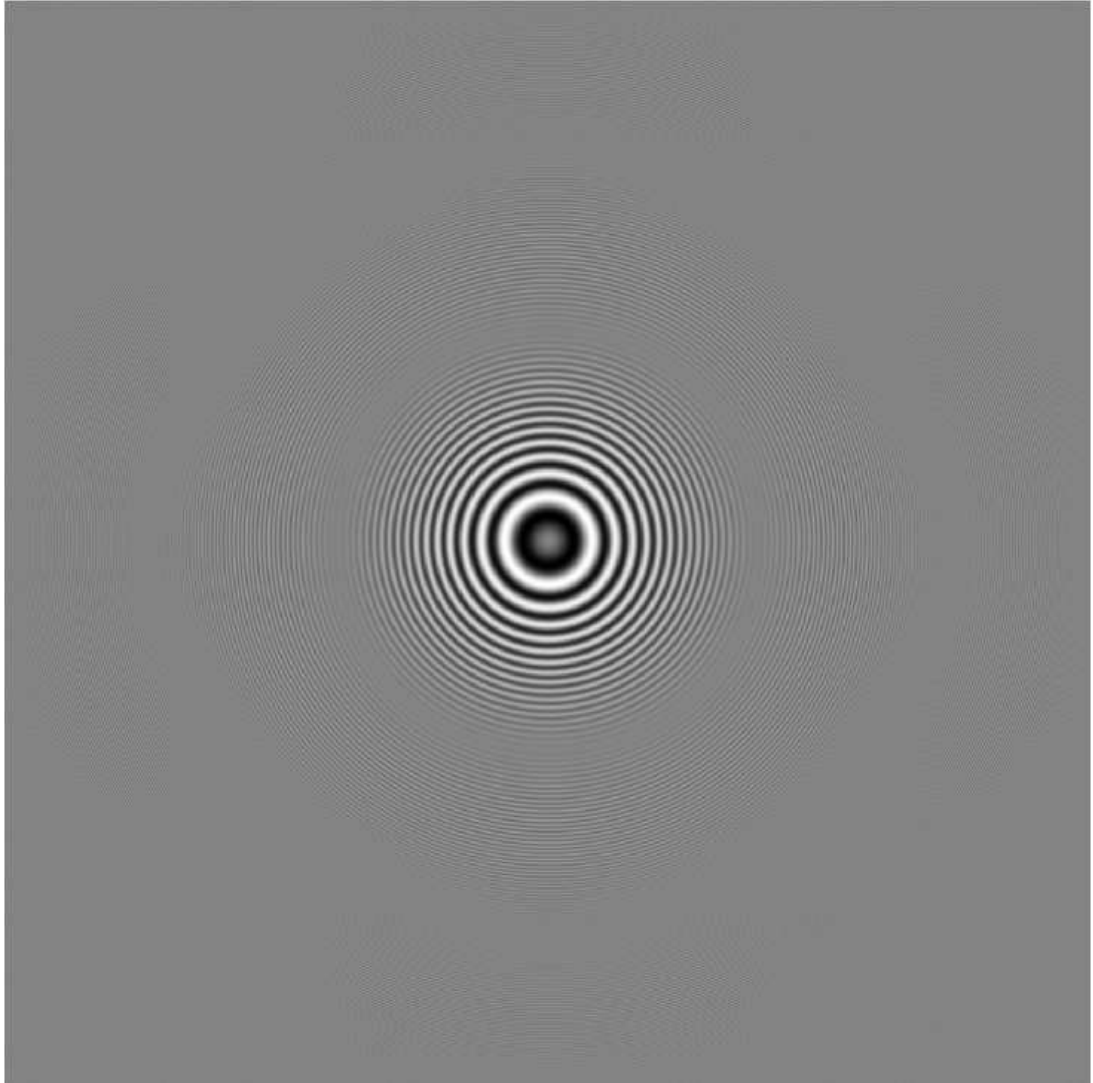


Figure 3.3: An example of a diffraction pattern; real part is plotted. This figure is (1024×1024) in size. The vertical axis is n -axis going from 1 to 1024 from top to down and horizontal axis is m -axis going from 1 to 1024 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0, 255]$, 255 being the brightest pixel in the figure.

Chapter 4

Wigner Transform

Wigner transform is a time-frequency method in signal processing. It gives information about the time variation of the frequency components of a signal. It was first proposed by Eugene Wigner as quantum corrections to classical statistical mechanics in 1932 by the name **Wigner quasi-probability distribution** (also called **the Wigner function** or **the Wigner-Ville distribution**) [17]. It is then understood that it is also a good analysis tool in time-frequency problems. When compared to short time Fourier transform, Wigner transform can yield better sharpness in some cases and it has the most of the properties that the other time-frequency techniques have.

Since discrete Wigner transform is used throughout this thesis, it may be better to concentrate on the basic properties of the discrete transform.

A diffraction pattern is a 2D space signal within which frequency information is stored. Hence a joint space-frequency analysis of such a signal is more likely to have more information than a single global frequency analysis. This relation is the same in time varying signals which carry frequency content because by the help of a joint time-frequency analysis it is possible to gain information about the time varying frequency content. Hence, a Wigner transform of a diffraction

pattern is a good candidate to gain information about the space varying frequency content of the signal which is closely related with the position parameters of the pattern itself as it will be explained in following subsections. [18]

4.1 Properties of the Discrete Wigner Distribution

Detailed information about the Wigner transform can be found in the literature [18], [19], [20], [21] and [22]. However, the basic definitions and properties of the discrete transform is restated here to ease the understanding of the following sections.

Basic Definitions:

1. The general continuous formulation of the Wigner transform of two time signals $f(t)$ and $g(t)$

$$W_{f,g}(t, \omega) = \int_{-\infty}^{\infty} f\left(t + \frac{\tau}{2}\right) g^*\left(t - \frac{\tau}{2}\right) \exp(-j\omega\tau) d\tau \quad (4.1)$$

where t, ω and τ are continuous. Auto Wigner transform of a continuous signal $f(t)$ is the same when $g^*(t)$ is replaced by $f^*(t)$.

2. The general discrete formulation of the cross Wigner transform of two time signals $f(n)$ and $g(n)$

$$W_{f,g}(n, \theta) = 2 \sum_{k=-\infty}^{\infty} \exp(-j2k\theta) f(n+k) g^*(n-k) \quad (4.2)$$

where n and k are discrete and θ is continuous. Auto Wigner transform of a discrete signal $f(n)$ is the same when $g^*(n)$ is replaced by $f^*(n)$.

Basic Properties (of the discrete auto wigner distribution) :

1. The function is periodic with respect to θ with a period of π .

$$W_f(n, \theta) = W_f(n, \theta + \pi), \quad \forall \theta \quad (4.3)$$

2. The auto Wigner transform $W_f(n, \theta)$ is real for any signal $f(n)$.

3. If the signal $f(n)$ is shifted in space by n_0 ,

i.e.

$$g(n) = f(n - n_0) \quad (4.4)$$

then the corresponding discrete-space Wigner distribution is also shifted in space by n_0 .

$$W_g(n, \theta) = W_f(n - n_0, \theta) \quad (4.5)$$

4. The following sum formula holds:

$$W_{f+g}(n, \theta) = W_f(n, \theta) + W_g(n, \theta) + 2Re\{W_{f,g}(n, \theta)\} \quad (4.6)$$

5. If $f(n)$ is a finite-duration signal, i.e.,

$$f(n) = 0, \quad n < n_a \text{ or } n > n_b \quad (4.7)$$

then

$$W_f(n, \theta) = 0, \quad n < n_a \text{ or } n > n_b \quad (4.8)$$

6. Since the discrete space Wigner transform is periodic with respect to frequency variable with a period π , frequency components beyond π interval causes aliasing when this space-frequency transform is applied. If the signal has zero energy outside π interval in frequency domain then aliasing is avoided. [18]

4.2 Wigner Analysis of Diffraction Patterns

4.2.1 1D Diffraction Patterns

For a 2D diffraction pattern the relation between the diffraction pattern and the object plane is governed by Equation 3.30. To start with a simpler case, the relation between the diffraction pattern and the object plane for 1D case is governed by Equation 4.9.

$$\psi_z(x) = [1 - a(x)] * h_z(x) \quad (4.9)$$

where

$$h_z(x) = \frac{1}{(j\lambda z)} \exp \left[j \left(\frac{\pi}{\lambda z} x^2 - \frac{\pi}{4} \right) \right] \quad (4.10)$$

If we assume that the object function $a(x)$ is real valued (true for all cases throughout this thesis) and take the intensity by eliminating negligible cross term the equation simplifies to

$$I_z(x) = 1 - a(x) * g(x) \quad (4.11)$$

where

$$g(x) = 2\text{Re}\{h_z(x)\} = \frac{2}{(\lambda z)^{1/2}} \cos\left(\frac{\pi}{\lambda z} x^2 - \frac{\pi}{4}\right) \quad (4.12)$$

[18]

The Wigner transform of a function gives information about the instantaneous frequency curve on a space-frequency domain with some possible artifacts due to nonlinear nature of Wigner transform. As a simple example let us consider a chirp function

$$f(t) = \exp(j\alpha t^2/2) \quad (4.13)$$

and its Wigner transform is given by

$$W_f(t, w) = 2\pi\delta(w - \alpha t) \quad (4.14)$$

This shows that the Wigner transform of a chirp function is a line impulse. Since the above kernel $g(x)$ is also a chirp function (furthermore it is a real valued chirp), it is possible to write this kernel in form of complex exponentials and then apply the summation property of the Wigner transform to write the Wigner transform of the kernel $g(x)$ as follows:

$$W_g(x, w) = \frac{2\pi}{\lambda z} \delta\left(w - \frac{2\pi}{\lambda z}x\right) + \frac{2\pi}{\lambda z} \delta\left(w + \frac{2\pi}{\lambda z}x\right) + \frac{2\sqrt{2}}{(\lambda z)^{1/2}} \cos\left(\frac{2\pi}{\lambda z}x^2 - \frac{\lambda z}{2\pi}w^2 - \frac{\pi}{4}\right) \quad (4.15)$$

It is important to notice that the first two terms in Equation 4.15 are the desired line impulses over the lines

$$w = [(2\pi)/(\lambda z)]x \quad (4.16)$$

$$w = -[(2\pi)/(\lambda z)]x \quad (4.17)$$

and the third term is called as the cross term which occurs due to the nonlinear nature of the Wigner transform. This term is negligible due to its low amplitude when compared to first two terms.

Now let us assume the simplest possible object plane configuration as $a(x) = \delta(x - x_0)$ and also use a simpler version of the intensity function shown in Equation 4.11 as

$$J(x) = 1 - I_z(x) \quad (4.18)$$

Hence it is now easy to calculate the Wigner transform of this diffraction function by simply making use of convolution property of the Wigner transform,

$$W_j(x, w) = \int_{-\infty}^{\infty} W_a(\alpha, w) W_g(x - \alpha, w) d\alpha \quad (4.19)$$

Since $W_a(x, w) = 2\delta(x - x_0)$ due to simple assumption of the object plane function and $W_g(x, w)$ is given by Equation 4.15 then

$$W_j(x, w) = 2W_g(x - x_0, w) \quad (4.20)$$

Equation 4.20 shows that the Wigner transform of a simple point object is a shifted copy of the Wigner transform of the chirp kernel function given by Equation 4.15. As explained before, if the cross term (third term in Equation 4.15 which has smaller amplitude than other two terms) is neglected, the graphical shape of the remaining terms given by Equations 4.16, 4.17 is two intersecting lines which intersect each other at the original x location of the object particle. Furthermore, the slopes of these lines gives a brief information about the distance between the object plane and the hologram plane. If the wavelength λ is known then the distance can be calculated as

$$z = \frac{2\pi}{\lambda m} \quad (4.21)$$

where m is the slope of the intersecting lines in Wigner transform domain.

4.2.2 2D Diffraction Patterns

It is important to notice that Wigner transform of a 1D function yields a 2D function in (x, ω) domain. Hence, it is easy to conclude that Wigner transform of a 2D function yields a 4D function which is hard to visualize and interpret. In order to handle these difficulties it is possible to apply Wigner transform on two different 1D slices of a 2D function (i.e. a vertical and a horizontal line segment from 2D diffraction pattern). The Wigner transform of the column (i.e. vertical line) function will give information about both depth and y -position of the object function whereas the Wigner transform of the row (i.e. horizontal line) function will give information about both depth and x -position of the object function. Let us assume that object function is $a(x, y)$ and Wigner transform of $a(x_0, y)$ is $W_{ax}(y, \omega)$ and Wigner transform of $a(x, y_0)$ is $W_{ay}(x, \omega)$.

Now, if we assume an impulsive object function i.e. $a(x, y) = \delta(x - x_a, y - y_a)$ for any x_a and y_a then the 2D intensity function can be written as

$$I_z(x, y) = 1 - a(x, y) * g(x, y) \quad (4.22)$$

as it is written in Equation 4.11 where $g(x, y)$ is the 2D kernel function as explained in Equation 4.12. It was explained that the kernel function is a real valued chirp function and the Wigner transform of a 1D chirp is two intersecting lines on (x, w) domain where x is the space variable and w is the frequency variable.

Let us call $J(x, y)$ the constant term adjusted intensity function as it is written in Equation 4.18.

So, the Wigner transform of $J(x_a, y)$ is

$$W_{J_x}(y, w) = W_g(y - y_a, w) \quad (4.23)$$

and the Wigner transform of $J(x, y_a)$ is

$$W_{J_y}(x, w) = W_g(x - x_a, w) \quad (4.24)$$

where $W_g(x, w)$ is given by Equation 4.15.

Hence the analysis of these Wigner transforms will give enough information about the (x, y, z) locations of the impulsive object function in each diffraction pattern.

4.2.3 Analysis of Wigner Transforms

Up to this point, a theoretical understanding of the Wigner transform is explained by the help of works done in literature. It is clear from the literature that Wigner transform is a good candidate to extract 3D position of a 2D diffraction

pattern by analyzing 1D slices of the diffraction pattern. After understanding this feature of the Wigner transform it is also important to develop some image processing tools to be able to extract this 3D position information from any diffraction pattern. The following sections explain an image processing method to analyze Wigner transforms of the diffraction patterns. Figure 4.1 shows a Wigner transform, as explained in Appendix E, of a 1D slice of a 2D diffraction pattern generated from an impulsive object function defined in Appendix C with the parameters $N = M = 1024$, $X_c = Y_c = 512$ and $r = 4.5$ as it is explained in Appendix D with the parameters $\lambda = 600nm$, $X = 10\lambda$, $N = 1024$ and $Z = NX^2/\lambda$. The middle row (i.e. $n = 512$ and $\forall m, m \in [1, 1024]$ from Figure 3.3) of the 2D diffraction pattern is used to generate Wigner transform.

The intersecting lines are clearly visible in Figure 4.1. The intersection point of the lines gives the estimate x or y position of the impulsive object function and the slope of the lines are related with the estimate z position of the diffraction pattern as indicated by Equations 4.16 or 4.17.

In order to get this information from Wigner transforms, it is required to computationally analyze these Wigner transform patterns to extract intersection point location and slopes of the lines seen in Figure 4.1. To emphasize the lines in the Wigner transform and to get rid of the cross term problem in the transform domain, edge detection as explained in details in Appendix F is implemented on the Wigner transform patterns. After that the resulting binary pattern is analyzed by the Hough transform given in Appendix G to calculate the slopes of the lines and the intersection point.

Edge Detection for Line Detection

To be able to automatically register the diffraction patterns, an automated Wigner transform analyzer is required. The first problem with the Wigner

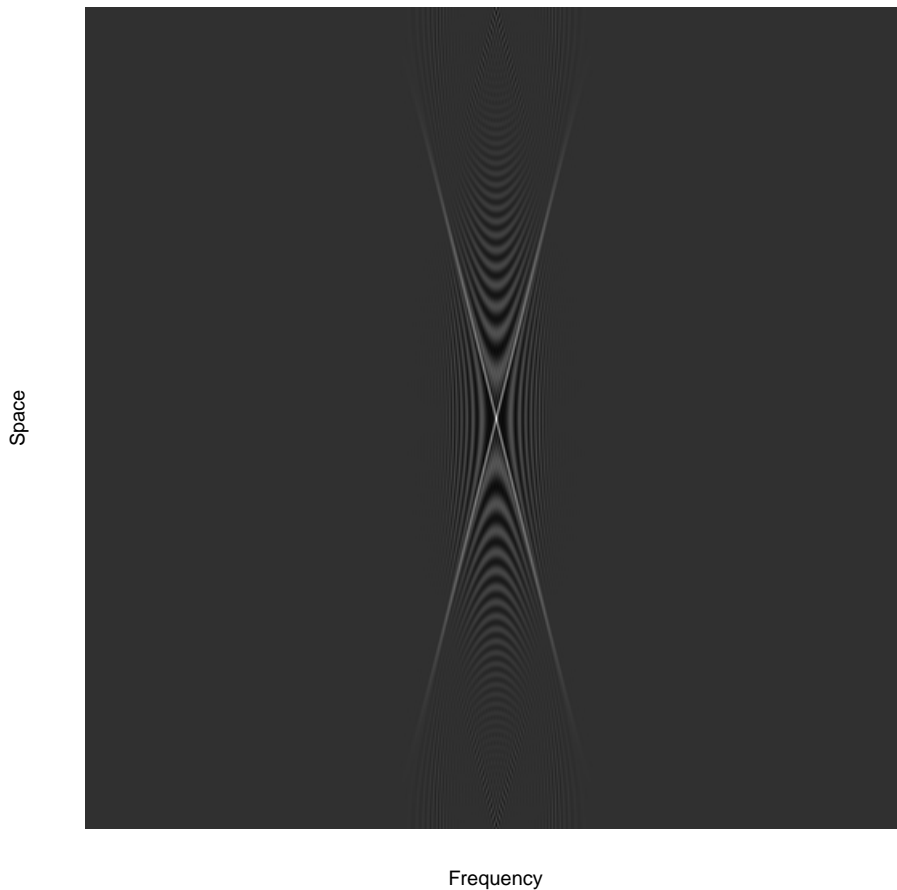


Figure 4.1: Wigner transform a chirp function. The horizontal axis is the frequency variable $\theta = \pi p/N$ where integer $p \in [-511, 512]$. The vertical axis is m -axis going from 1 to 1024 from top to down and horizontal axis is p -axis going from -511 to 512 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0, 255]$, 255 being the brightest pixel in the figure.

transform is the cross term problem. For some object function and under noisy environments it is possible to have strong cross terms which can become indistinguishable with the desired Wigner lines when compared simply according to their gray-level pixel values. So a method which can perform in a more robust manner under strong cross term interferences should be implemented. When the nature of the white lines seen in Figure 4.1 is compared to the cross term effect, it is clear that the white lines form edges with the remaining pattern due to high contrast ratio. Also we know that Hough transform is a good candidate to analyze these lines in a given pattern to find their slopes. Hough transform is also more suitable to binary patterns. Hence, it is better to use a method to distinguish these lines from the rest of the pattern (especially from the cross term effect); this will yield a binary output in which the lines are emphasized and are analyzed by Hough transform in an easier way due to binary nature.

An edge detection algorithm is used to extract these lines from the rest of the pattern. The binary output of this edge detector is shown in Figure 4.2. Some different edge detection methods are tried and finally Canny edge detection algorithm as explained in Appendix F is used to extract the line information from Wigner transform functions. The input of this detector is the pattern shown in Figure 4.1. The reason to select Canny edge detector is the ability of this detector to detect weak edges which are connected to strong edges. The thresholds for Canny edge detection was selected as $[0.1 \ 0.5]$ and details about these thresholds are given in Appendix F. By the help of this ability of this detector, it became possible to detect these Wigner lines under some stronger cross term conditions and noisy environments. Hence, as it was aimed, a binary pattern with the lines having the highest contrast according to the rest of the pattern is generated as seen in Figure 4.2.

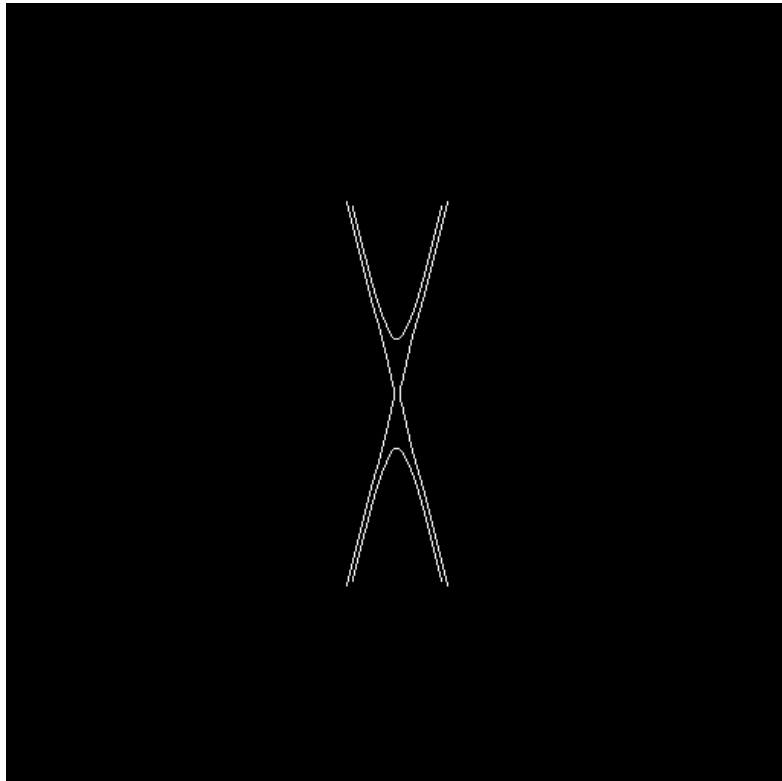


Figure 4.2: The resulting binary image when Canny edge detection is applied to the Figure 4.1

Hough Transform to Detect Lines

Hough transform is a 2D-to-2D mapping in which the lines in the beginning pattern having the same slope are mapped to the same point in the resulting pattern to make that point brighter than the neighboring points in a grayscale transform environment (further details are given in Appendix G). Since it is clear that in a Wigner transform of the diffraction pattern generated from an impulsive object function as explained in Appendices C and D and Subsection 4.2.3 there will be two strong lines having negative slopes as given by Equations 4.16 and 4.17 as shown in Figure 4.2, it is legitimate to search for two peaks of the Hough transform of the Wigner transform function. By the help of such information, it is possible to detect some parts of the Wigner lines in the original Wigner transform patterns. This result is shown in Figure 4.3.

As seen in Figure 4.3, the strong edges which are formed between the Wigner lines and the rest of the pattern and which are detected by Canny edge detector as shown in Figure 4.2 are easily recognized by the Hough transform. The slope information given in Equations 4.16 and 4.17 is the same as the slope of the lines detected by the Hough transform because the lines detected by Hough transform are the edges of the real Wigner lines. So depth information which is closely related to slope of these Wigner lines as given by Equation 4.21 remains almost the same. However, intersection point of the Wigner lines which is closely related to the (x, y) position of the object function, may change within a small neighborhood. To get rid of this error, an iterative method is used as explained in registering process in Chapter 5.

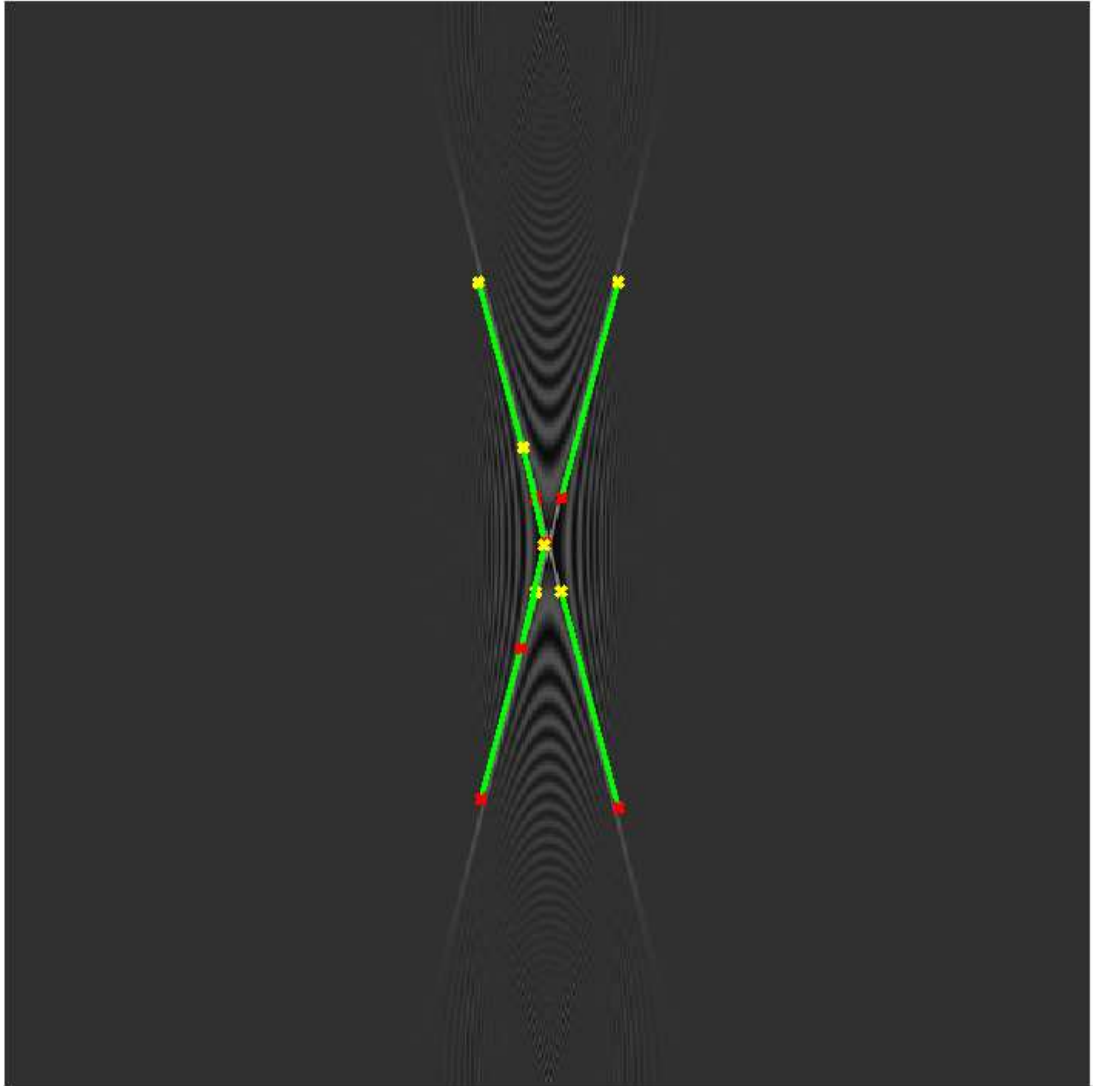


Figure 4.3: The detected lines by edge detection and Hough transform are shown on the initial Wigner transform pattern.

Chapter 5

Registration and Stitching

We have already explained the basics of optical diffraction in Chapter 3 and some basic properties of the time-frequency analysis in Chapter 4. This chapter will be about how to make use of these facts to be able to extract the relevant position information from a diffraction pattern by the help of Wigner transform, how to make these estimates more accurate by some iterative methods and finally how to merge a number of these diffraction patterns with a suitable interpolation method in order to have one higher resolution diffraction pattern.

5.1 Registration of Digital Holograms

The aim of this study is to merge some number of low resolution diffraction patterns in order to form a higher resolution diffraction pattern. The classical method in holography is to record a single hologram and then to try to reconstruct it. There is a number of new methods which states that if it is possible to form a high resolution diffraction pattern from some low resolution patterns which are recorded from different 3D positions with respect to object field, it

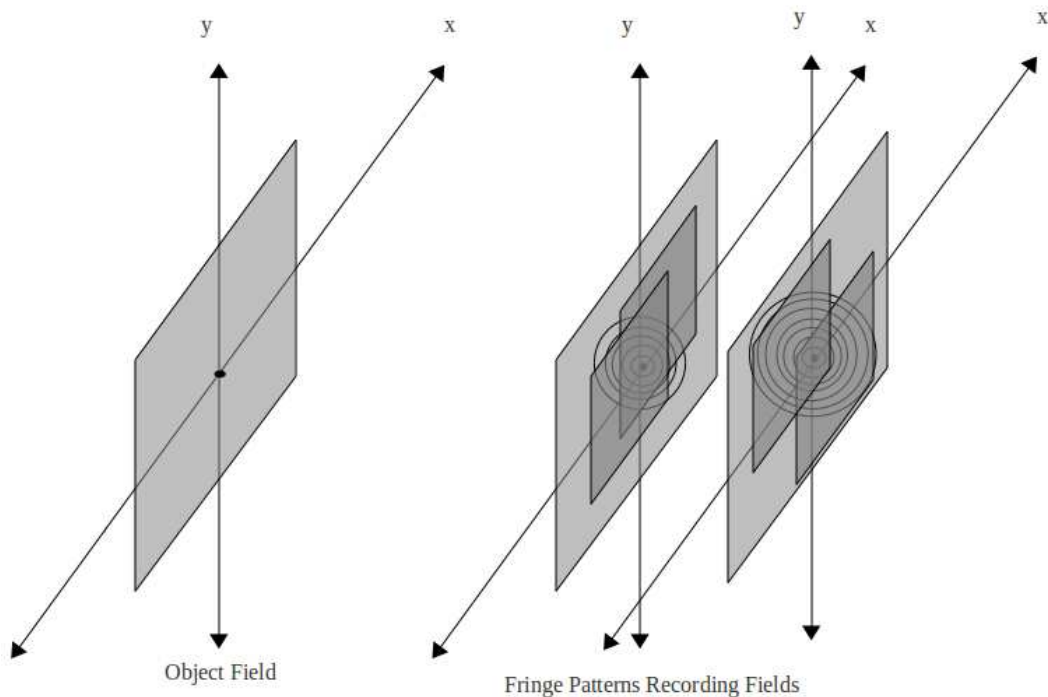


Figure 5.1: The way by which the low resolution diffraction patterns are recorded (or calculated for simulation purposes)

may be possible to have larger viewing angles in the reconstruction process. The idea of low resolution pattern recording is shown in Figure 5.1.

To simplify the problem of registration, we simply dealt with the diffraction patterns which are recorded on planes which are parallel to the object field plane as shown in Figure 5.1. Both (x, y) locations and depth of each recording is different and unknown. The patterns shown in Figure 5.2 are some examples of low resolution diffraction patterns recorded (or calculated for simulation purposes) on a parallel plane to the object field plane when the aperture function was simply a circular slit such that

$$p(x, y) = \left\{ \begin{array}{ll} 0 & , \quad (x - x_c)^2 + (y - y_c)^2 \leq r^2 \\ 1 & , \quad otherwise \end{array} \right\} \quad (5.1)$$

where (x_c, y_c) is the center of the circular aperture on the object field plane, r is the radius of the aperture and x, y are integers satisfying $x \in [1, N]$, $y \in [1, M]$

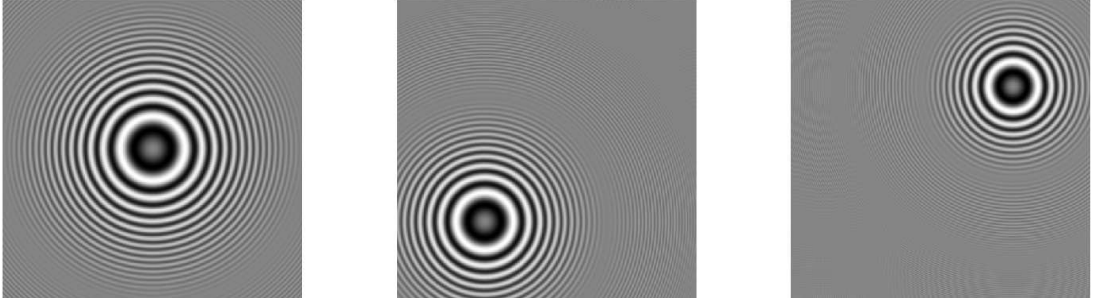


Figure 5.2: These are some different diffraction patterns generated by computer for different 3D positions with respect to the same object field shown in Figure 3.2. The vertical axis for all patterns is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0, 255]$, 255 being the brightest pixel in the figure. The generation details of these patterns are given in Appendix D. The parameters are $\lambda = 600nm$, $X = 10\lambda$ and $Z = kNX^2/\lambda$ where $k = [1.0, 0.9, 0.8]$ respectively. The input pattern is generated by using Appendix C with size 2048×2048 object being in the center with radius $r = 4.5$ and a different 512×512 section of each diffraction pattern is grabbed and shown in this figure. The details of this low resolution pattern grabbing from a higher resolution one are explained in text where necessary. These figures are only for example purpose, no further operation is conducted on these.

for a $(N \times M)$ initial object matrix. An example of the aperture function is shown in Figure 3.2.

How to calculate the wave distribution just after the aperture plane if the aperture is illuminated with a plane wave propagating in the z -direction is already explained in Chapter 3. Patterns in Figure 5.2 are computed using details in Appendix D with an object pattern given by Equation 5.1 and discrete methods described in Appendix C. The wavelength of the propagating plane wave, λ , is 600nm and the distance, d , between the recording plane and the object plane is $d = NX^2/\lambda$ where $N = 2048$ is the size of the object pattern and $X = 10\lambda$ is the sampling period of the recording sensor. The center of the object is at $(Y_c, X_c) = (1024, 1024)$ and the radius of the object is $r = 4.5$ pixels. The registration process is composed of the estimation of the position parameters $(x, y, z)_{est}$ from a given diffraction pattern without any prior knowledge about the position of the sensor which have recorded the pattern (or without any prior

knowledge about the real parameter which are used to calculate it in a computer environment). The registration process may also include some iterative methods to be able to make $(x, y, z)_{est}$ more accurate and call them $(x, y, z)_{finer}$ to make errors as small as possible during the stitching process.

5.1.1 Wigner Analysis

In Chapter 4, the meaning of the Wigner transform of a 1D diffraction pattern is explained in detail. It is also stated that it is possible to get the same kind of information from the Wigner transform of 1D slices from a 2D diffraction pattern. If these 1D slices are chosen in a legitimate way, it is possible to get $(x, y, z)_{est}$ just by applying the Wigner transform to two perpendicular slices of the 2D pattern, i.e. a row and a column slice. To begin with, there is only a 2D diffraction pattern in hand as shown in Figure 5.3. So the first step to start analysis is to apply a Wigner transform to 1D slices of this pattern. To suppress the effect of the cross term, it is useful to get slices which pass through the center of the aperture if possible. So for this specific example, it is legitimate to get the slices shown by the black lines in Figure 5.4.

In Figures 5.7 and 5.6, the result of the Wigner transform of the slices given in Figure 5.4 are shown. Arms of the Wigner transform as explained in Chapter 4 is clearly visible in this figure. One point to note is that the slope of the arms should be the same in both figures since the slope is the depth related parameter and the depth remains the same within the same diffraction pattern since tilts are avoided. However, intersection point of the white Wigner arms varies since the Wigner transform of the horizontal slice carries information about the x location of the pattern whereas the Wigner transform of the vertical slice carries information about the y location of the pattern.

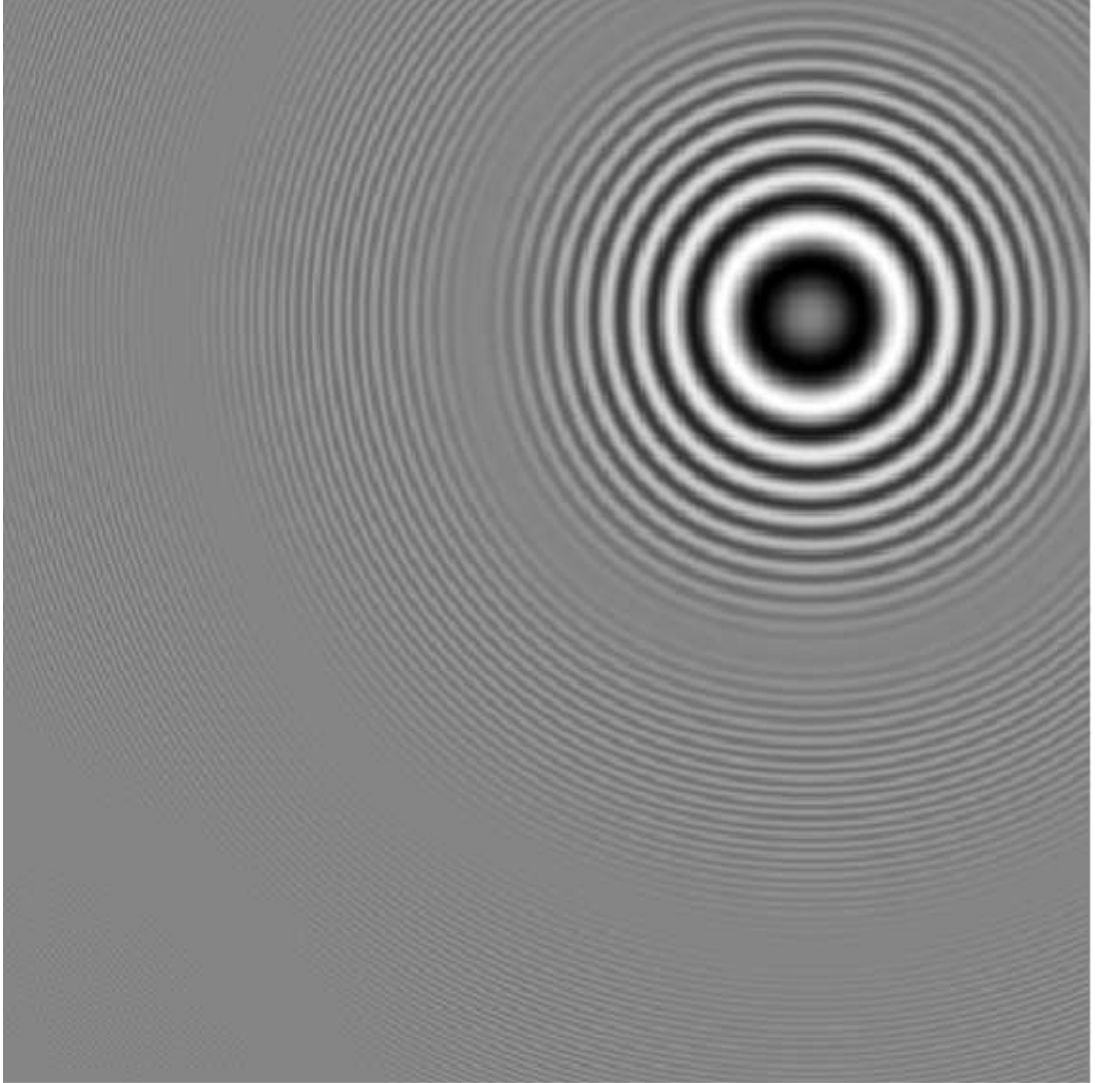


Figure 5.3: This is an example diffraction pattern. This figure obeys the details explained in Appendix A. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0\ 255]$, 255 being the brightest pixel in the figure.

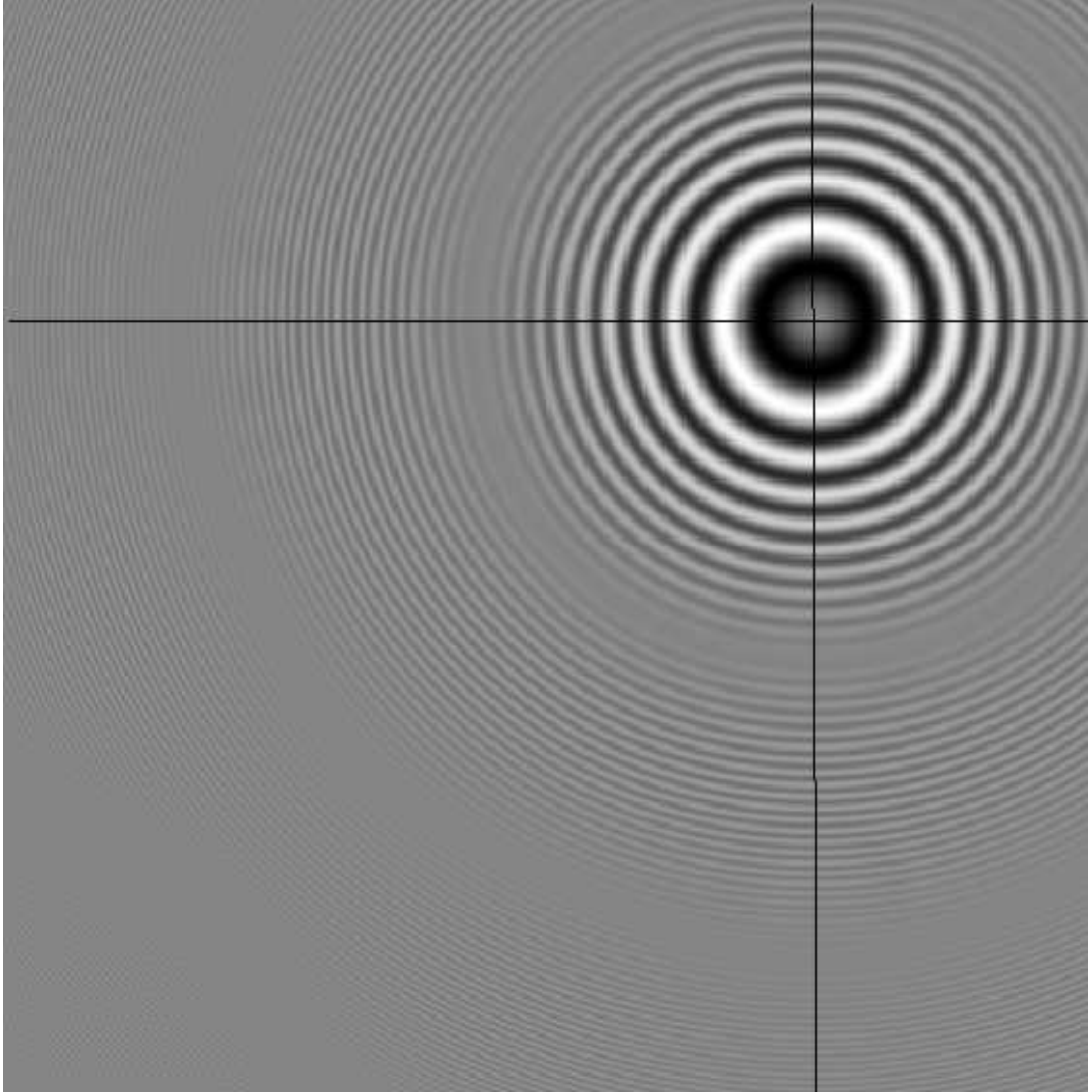


Figure 5.4: The horizontal and vertical slices of the pattern which are used to form Wigner transforms in the following parts. This figure obeys the details explained in Appendix A. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

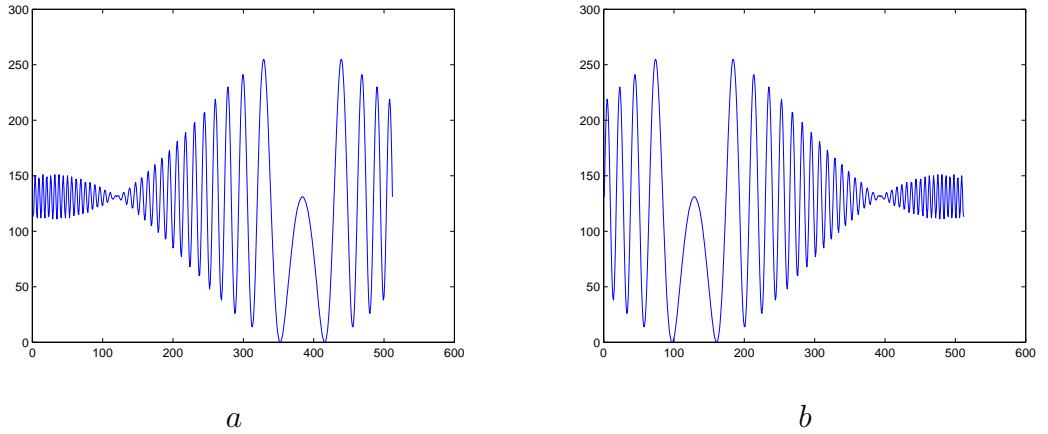


Figure 5.5: The plots of the slices of the pattern shown in Figure 5.4. (a) is the plot of row slice and (b) is the plot of column slice. Horizontal axis is the space variable in the interval $[1\ 512]$ and the vertical variable is the pixel value between $[0\ 255]$, 255 being the brightest pixel in the pattern.

After this point, these Wigner transforms are analysed by the methods given in Chapter 4, i.e. edge detection and Hough transform, to get the initial estimate $(x, y, z)_{est}$. The Figure 5.8 shows the output of the edge detector of the two patterns. Canny edge detector and Hough transform are well defined in literature [23], [24]. The Figure 5.9 shows the results of the detection over the original Wigner transform. As it is clear from the figure, detection is successful especially in terms of slope (i.e. in terms of depth parameter z) but not as expected in terms of intersection point (i.e. in terms of x, y location) because the edges that are formed between the Wigner lines and the rest of the pattern are detected instead of the centers of the lines. For all the parameters some iterative methods are required to be applied. For the z parameter, it is easier to solve the problem in registration process since it is related to sharpness of the single reconstructed pattern (i.e. it is the focusing parameter of the pattern). However, x, y position of the pattern is the relative information within different diffraction patterns so it is logical to solve this problem in stitching process. Hence, x, y location problem is left for the stitching part and z parameter problem is solved in this process. As a last remark on Wigner transform, the reconstructed pattern is shown in Figure

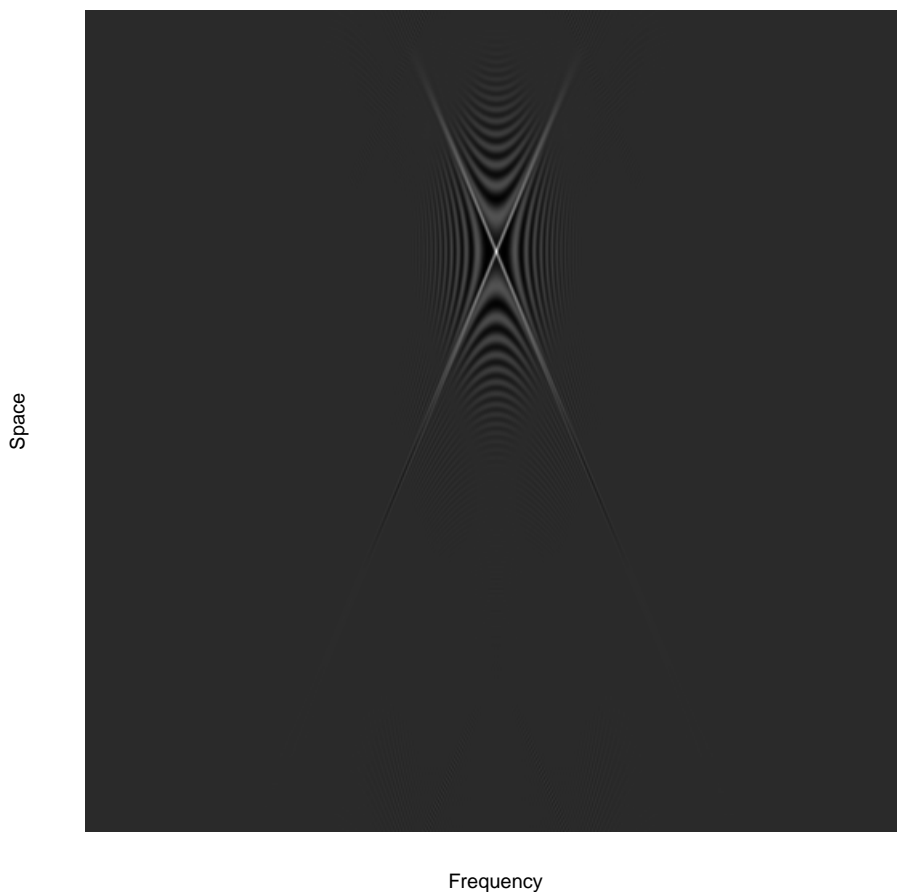


Figure 5.6: Wigner transform of the column function shown in Figure 5.5 a, calculated as explained in Appendix E. The horizontal axis is the frequency variable $\theta = \pi p/N$ where integer $p \in [-255, 256]$. The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is p -axis going from -255 to 256 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

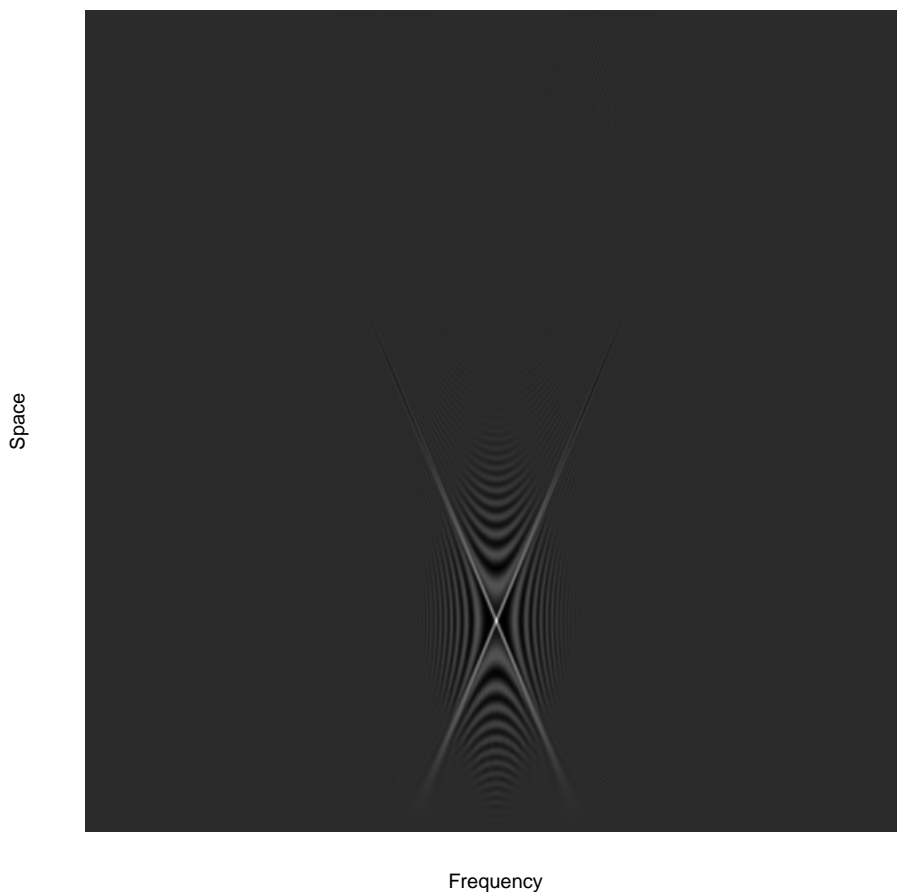


Figure 5.7: Wigner transform of the row function shown in Figure 5.5 b, calculated as explained in Appendix E. The horizontal axis is the frequency variable $\theta = \pi p/N$ where integer $p \in [-255, 256]$. The vertical axis is m -axis going from 1 to 512 from top to down and horizontal axis is p -axis going from -255 to 256 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.



Figure 5.8: The result of edge detection on column and row functions. All the figure properties are the same as Figures 5.6 and 5.7, respectively.



Figure 5.9: The lines detected by edge detection and Hough transform shown on Wigner transform output for column and row functions. All the figure properties are the same as Figures 5.6 and 5.7, respectively.

5.10 with the parameters $(x, y, z)_{est}$ estimated only by analysis of the Wigner transform.

5.1.2 Iterative Methods for Depth Estimation

Having estimated all the position parameters by Wigner transform as explained in Section 5.1.1, we now need to correct the depth parameter by reducing the error. As seen in Figure 5.10, when we reconstruct the pattern with the depth

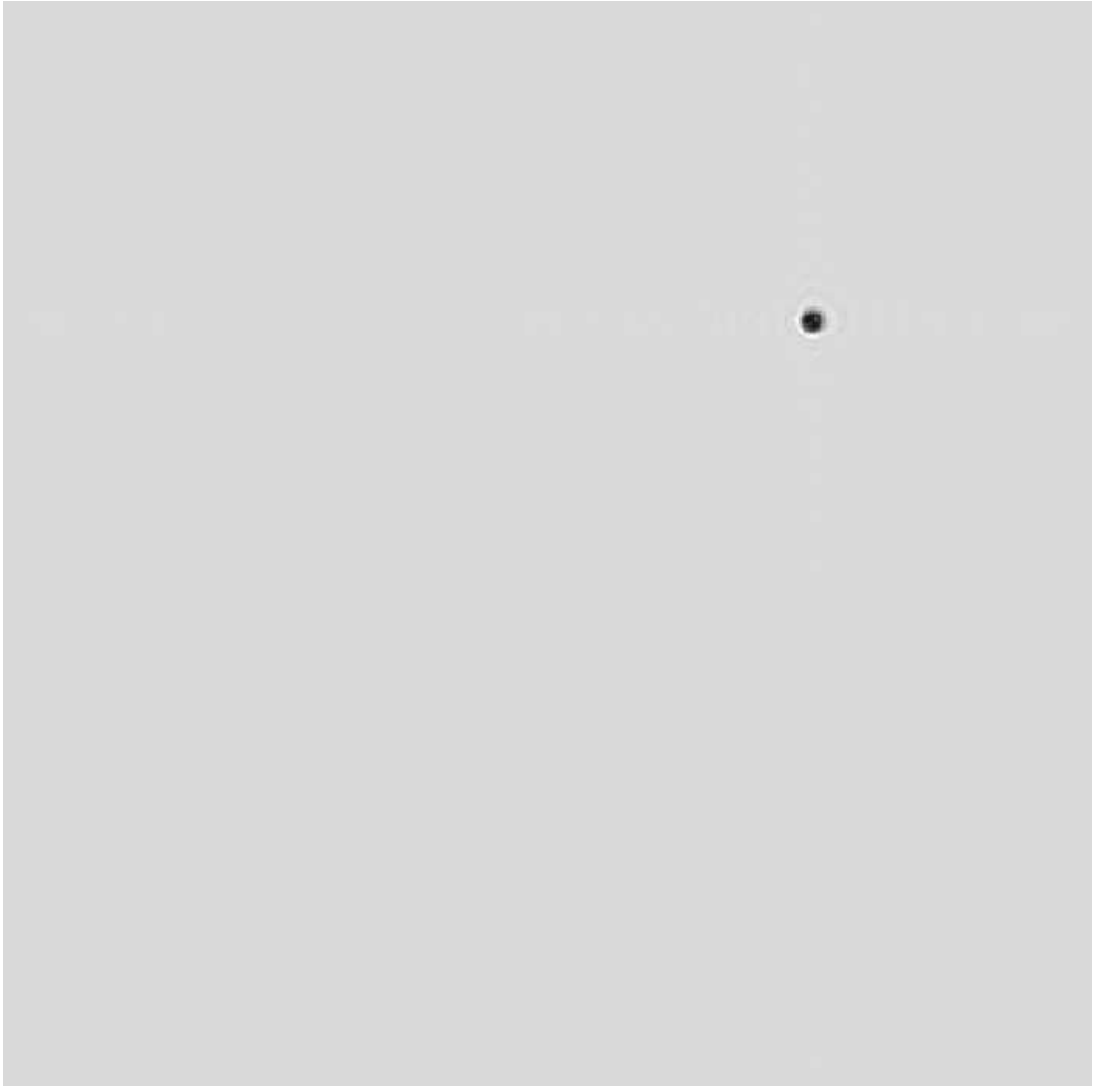


Figure 5.10: This is the reconstructed pattern only with the parameters gained by the Wigner transform. Due to incorrect depth estimation (approximately 5% error in depth) the fringe effect is still visible in this reconstruction. The size of the figure is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0, 255]$, 255 being the brightest pixel in the figure.

parameter estimated by the Wigner transform, some fringes remain. This is due to the difference between the actual depth parameter and the estimated one. Since the reconstruction is done with the estimated parameter, and it is different than the actual depth, a focused reconstruction is not possible. In other words, since the diffraction pattern is not propagated up until the real depth on which the object plane should lie due to the error on estimated depth, the resultant reconstruction is also a diffraction pattern of the original object field at a close distance. Hence, the fringe pattern is still visible in this reconstruction as a result of defocusing.

It is commonly known that in daily life photography, if the recorded scene is not in focus, then the recorded image will not be sharp, i.e. there will be a blur in the recorded image. The blurring effect is most disturbing at the edges. Nowadays, many digital cameras with suitable mechanical lenses offer some autofocus feature. In theory, this autofocus issue is achieved by examining the scene with different focal lengths with a moving local variance filter. The maximum of the variance is recorded for different focal lengths and at the end the focal length having the maximum local variance value is selected as the autofocus focal length. Since the effect of defocusing is a blur, the local variance filter selects the focal length in which the sharpest edges are formed in order to get rid of the blurring effect.

Diffraction pattern recording is different from daily life photography because diffraction patterns are generated by monochromatic coherent light sources. Hence, the effect of defocusing is observed as fringe patterns, not as a simple blur. Indeed, any diffraction pattern is a defocused photograph of the original object field under coherent imaging.

In coherent imaging all the equations governing the propagation of light are well known, so reconstruction from the out-of-focus recording is possible when

some of the parameters (like wavelength of the light source, physical dimensions of the recording CCD, etc.) are known.

To sum up, a diffraction pattern yields the sharpest pattern if it is reconstructed with correct depth value provided that the other parameters are known. The result z_{est} from the Wigner transform is not the correct depth as seen in Figure 5.10 because there are still some fringes in the reconstructed pattern. However, it is clear that it is quite close to the correct value. Hence, iteration is logical in this case to search for the correct parameter. It is also known that, because of the equation governing the propagation of light in free space, the fringes are visible before and after the real depth, so it is not possible to discriminate in which direction to iterate at the beginning of the iteration. To be able to make the iterations faster, a stepped iteration method is used. At the first stage, a large step is used to understand the behaviour of the fringe pattern, i.e. try to find which direction to go in order to be able to get rid of fringe pattern. The second and possibly a third stage use finer steps to locate the correct depth parameter. The constraint function is a moving local variance filter over the reconstructed pattern which is a parameter to tell the sharpness of the pattern.

The local variance filter is a moving window on the reconstructed pattern and has the formula given by Equation 5.3.

$$mean = \frac{1}{N \times M} \sum_{a=1}^N \sum_{b=1}^M P(a, b) \quad (5.2)$$

$$v_l(i, j) = \frac{1}{SOF^2} \sum_{a=i-hs}^{i+hs} \sum_{b=j-hs}^{j+hs} [P(a, b) - mean]^2 \quad (5.3)$$

where

$$\begin{aligned}
P &= \text{reconstructed pattern} \\
N &= \text{vertical resolution of } P \\
M &= \text{horizontal resolution of } P \\
v_l &= \text{local variance} \\
m &= \text{local mean} \\
SOF &= \text{size of the filter (i.e. } SOF - by - SOF \text{ moving window)} \\
hs &= \text{half size of the filter (i.e. } (SOF - 1)/2) \\
i, j &= \text{position on } P
\end{aligned} \tag{5.4}$$

It is important to note that i, j should be selected accordingly not to exceed matrix dimensions.

In the iteration process, for each reconstructed pattern, this moving local variance filter is applied and the maximum v_l for each pattern is recorded. Then the pattern which has the biggest v_l is selected as the nearest depth to the actual depth since it is the most focused pattern due to variance filter analysis. The next step of the iteration (if it exists) starts from this updated depth estimation. The algorithm of this iteration process is given below.

Step 1. A reconstruction of the diffraction pattern is generated using the depth estimated by the Wigner analysis (i.e. z_{est}), as explained in Appendix D with pre-known wavelength of the coherent light source ($\lambda = 600nm$) used in recording phase and with preknown sampling period of the recording equipment ($X = 10\lambda$).

Step 2. 21 different reconstructions are calculated with the parameters given above (N, X and λ) and for depth range of $d = z_{est} + k0.01z_{est}$ for $k \in [-10, -9, \dots, -1, 0, 1, \dots, 9, 10]$.

- Step 3.** The moving local variance filter given by Equation 5.3 is applied to each pattern and the maximum variance is recorded for each pattern (i.e. $v(d) \rightarrow \max_{(i,j)} [v_l(i,j)]_{RP_d}$).
- Step 4.** The pattern which has the maximum variance (i.e. $\max_d [v(d)]$) is selected as the best focused image. In other words the depth variable d which was used to reconstruct that pattern is the closest depth to the actual depth within above iteration interval; so z_{est} is updated to this value. (i.e. z_{est} is set to $\arg(\max_d [v(d)])$.)
- Step 5.** 21 different reconstructions are calculated with the parameters given above (N, X and λ) and for depth range of $d = z_{est} + k0.001z_{est}$ for $k \in [-10, -9, \dots, -1, 0, 1, \dots, 9, 10]$.
- Step 6.** The moving local variance filter given by Equation 5.3 is applied to each pattern and the maximum variance is recorded for each pattern (i.e. $v(d) \rightarrow \max_{(i,j)} [v_l(i,j)]_{RP_d}$).
- Step 7.** The pattern which has the maximum variance (i.e. $\max_d [v(d)]$) is selected as the best focused image. In other words the depth variable d which was used to reconstruct that pattern is the closest depth to the actual depth within above iteration interval; so z_{est} is updated to this value. (i.e. z_{est} is set to $\arg(\max_d [v(d)])$.)
- Step 8.** 21 different reconstructions are calculated with the parameters given above (N, X and λ) and for depth range of $d = z_{est} + k0.0001z_{est}$ for $k \in [-10, -9, \dots, -1, 0, 1, \dots, 9, 10]$.
- Step 9.** The moving local variance filter given by Equation 5.3 is applied to each pattern and the maximum variance is recorded for each pattern (i.e. $v(d) \rightarrow \max_{(i,j)} [v_l(i,j)]_{RP_d}$).
- Step 10.** The pattern which has the maximum variance (i.e. $\max_d [v(d)]$) is selected as the best focused image. In other words the depth variable d which was used

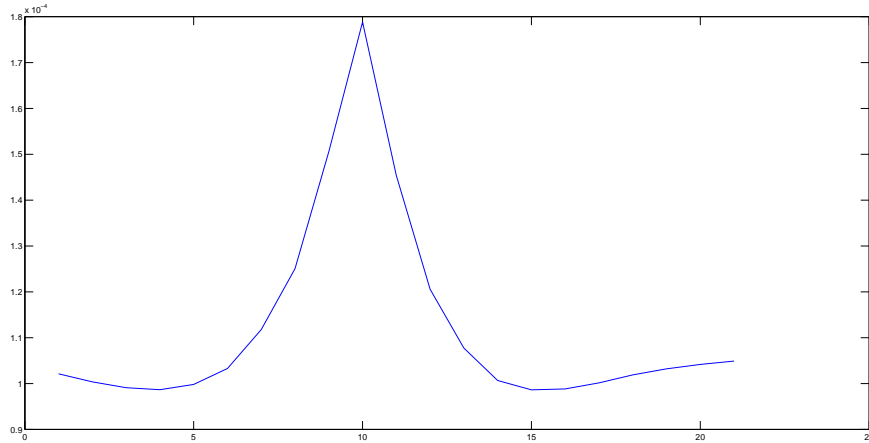


Figure 5.11: This is the result of the variance filter in the first part of the iteration process. The best estimate corresponds to maximum variance. The vertical axis represents the maximum local variance in each iteration. Horizontal axis is the iteration number called s which satisfies $k = s - 11$ for iteration step 2.

to reconstruct that pattern is the closest depth to the actual depth within above iteration interval; so since this is final iteration z_{finer} is selected as this value. (i.e. z_{finer} is set to $\arg(\max_d[v(d)])$.)

The starting pattern is shown in Figure 5.10. The Figures 5.11, 5.12, 5.13 show the intermediate steps of the iterative method with the constraint function plots. Figure 5.14 shows the reconstructed pattern with the z_{finer} parameter after iterations. It is clear that the fringe pattern has disappeared in this resultant reconstruction.

5.2 Stitching of Digital Holograms

After a successful registration process, we have $(x, y)_{est}$ and z_{finer} for the diffraction pattern given in Figure 5.3. Let us call these parameters $(x, y)_{est_1}$ and z_{finer_1} by assuming that this pattern was the first pattern to be registered. Hence, if we have n diffraction patterns, after the registration process we

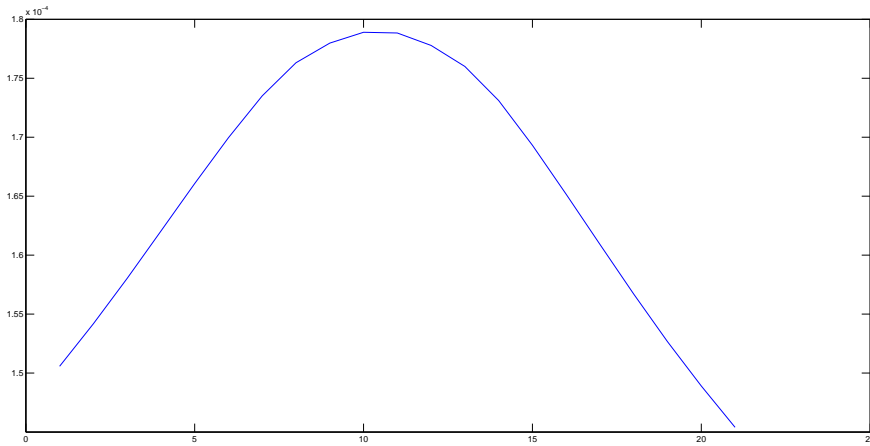


Figure 5.12: This is the result of the variance filter in the second part of the iteration process. The best estimate corresponds to maximum variance. The vertical axis represents the maximum local variance in each iteration. Horizontal axis is the iteration number called s which satisfies $k = s - 11$ for iteration step 5.

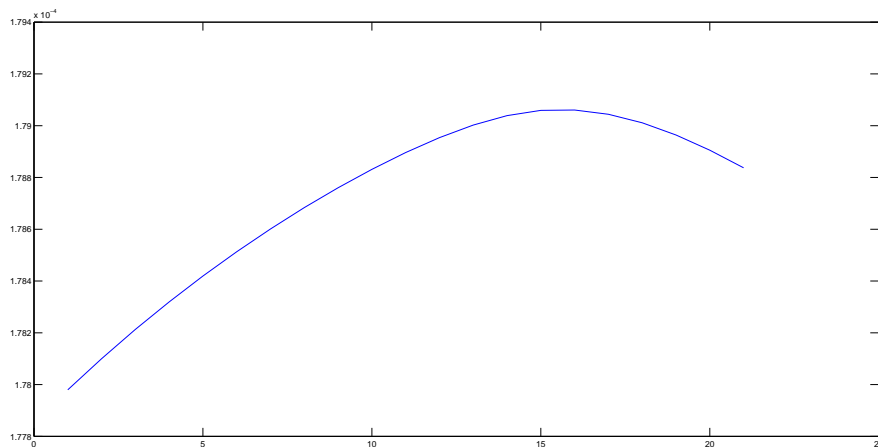


Figure 5.13: This is the result of the variance filter in the third part of the iteration process. The best estimate corresponds to maximum variance. The vertical axis represents the maximum local variance in each iteration. Horizontal axis is the iteration number called s which satisfies $k = s - 11$ for iteration step 8.

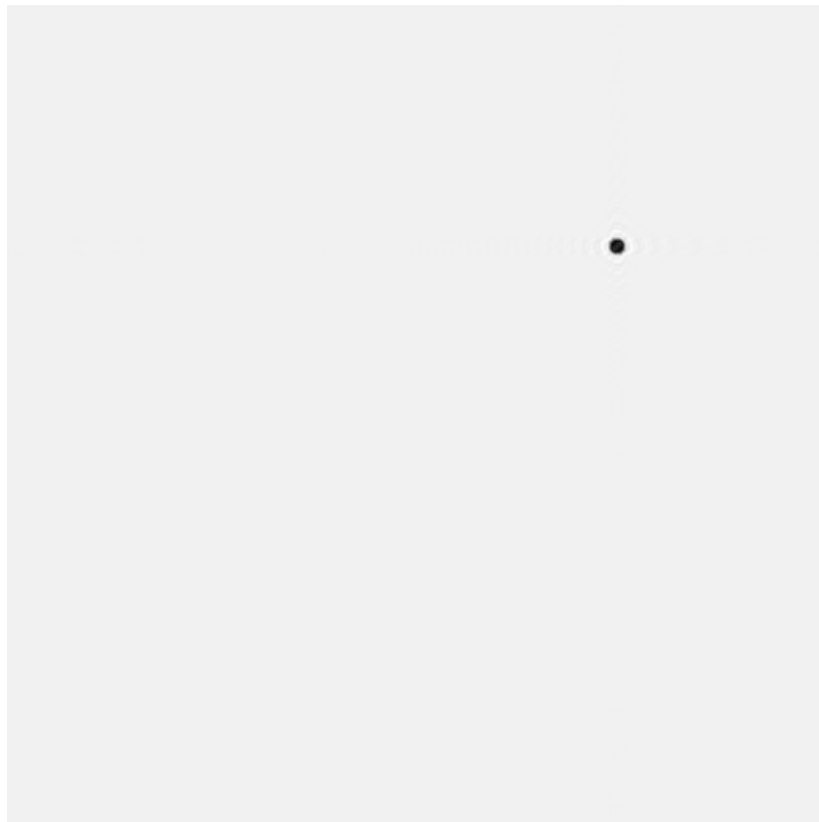


Figure 5.14: The final reconstruction after the depth iteration process. The size of the pattern is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

will have a vector for each parameter such that $X_{initial} = [x_{est_1} x_{est_2} \dots x_{est_n}]$, $Y_{initial} = [y_{est_1} y_{est_2} \dots y_{est_n}]$ and $Z_{final} = [z_{finer_1} z_{finer_2} \dots z_{finer_n}]$.

In the stitching process, we need to merge the low resolution diffraction patterns in order to be able to get the final high resolution diffraction pattern. As it is explained in the previous paragraph, after the registration process, we have some good estimates of the depth parameter (i.e. z position of the recording sensors) and some rough estimates of the x, y positions of the sensors as depicted in Figure 5.15. Another important point about these diffraction patterns is that any of these patterns is a snapshot of the same light distribution at an unknown position in space, as shown in Figure 5.15 with the estimate parameters. Hence, it is clear that if we know the actual object function (at $z = 0$ in Figure 5.15) and we propagate that function to the depths given in figure, we will end up having the same low resolution diffraction patterns at the same z locations due to consistency. So, it is legitimate to see the stitching problem as an inverse problem of the above situation. We do not have the object function, indeed we are trying to recover the object function from a number of low resolution snapshots of it. We have a number of low resolution patterns as depicted by LRP_i in Figure 5.15 and some estimates about their positions in space given by $x_{est_i}, y_{est_i}, z_{finer_i}$ for $i = 1, 2, 3$ as generated in the registration process. As explained before, any of these LRP 's contains some information about the object function but none of these contains the entire information. They may contain the same information content (i.e. there may be redundant parts) but we know that they are consistent since all of them were recorded from the same object function.

As a consequence, the stitching process can be summarized as a constraint implementation process. We start from one of the given low resolution diffraction patterns, for example, $LRP1$ in Figure 5.15. Since we know the depth parameters of all the patterns as fine as possible from the registration process, by making use of the methods in Appendix D we propagate the starting diffraction pattern

$LRP1$ to the depth of another diffraction pattern in hand ($LRP2$ for example) and apply the second pattern $LRP2$ as a constraint to the field generated by the starting pattern $LRP1$. This constraint application process is nothing but the insertion of the newly added diffraction pattern ($LRP2$) onto the pattern generated by the propagation of first pattern ($LRP1$) to the depth of the second pattern. Suitable interpolation (i.e. zeroth order interpolator) is used during this insertion. All the data in the second pattern is copied to the corresponding location in the propagated pattern. The details of this method is given below:

- Let us assume that $LRP1$ is size $N \times N$. Generate a blank pattern of size $4N \times 4N$ and call it HRP , the high resolution pattern. All the entries of HRP are initially equal to the mean graylevel of the $LRP1$ i.e. the mean value of all the pixels in $LRP1$. It may be necessary to adjust this mean graylevel when an additional low resolution pattern is included in HRP in order to get rid of edge effects in propagation, however, the mean graylevels of the low resolution patterns used in this thesis are equal, so after this step no adjustment for mean graylevel is required. With (Y_{est1}, X_{est1}) known, place $LRP1$ in the corresponding location on HRP as given in the equation below:

$$HRP[2N - \text{round}(Y_{est1}) - 1 + n, 2N - \text{round}(X_{est1}) - 1 + m] = LRP1[n, m] \quad (5.5)$$

where $n, m \in [1, 2, 3, \dots, N]$.

- Propagate HRP to the depth of $LRP2$ by using $Z = -Z_{finer1} + Z_{finer2}$, predetermined wavelength and sampling period of the recording equipment ($\lambda = 600nm$ and $X = 10\lambda$ for this thesis) and the methods given in Appendix D size being $4N \times 4N$.

- Copy the information in $LRP2$ to the correct position in HRP as explained below:

$$HRP[2N - \text{round}(Y_{est2}) - 1 + n, 2N - \text{round}(X_{est2}) - 1 + m] = LRP2[n, m] \quad (5.6)$$

where $n, m \in [1, 2, 3, \dots, N]$. Be careful not to change the entries of HRP that do not coincide with the location of $LRP2$ on HRP .

- Propagate HRP to the depth of $LRP3$ by using $Z = -Z_{finer2} + Z_{finer3}$, predetermined wavelength and sampling period of the recording equipment ($\lambda = 600nm$ and $X = 10\lambda$ for this thesis) and the methods given in Appendix D size being $4N \times 4N$.

- Copy the information in $LRP3$ to the correct position in HRP as explained below:

$$HRP[2N - \text{round}(Y_{est3}) - 1 + n, 2N - \text{round}(X_{est3}) - 1 + m] = LRP3[n, m] \quad (5.7)$$

where $n, m \in [1, 2, 3, \dots, N]$. Be careful not to change the entries of HRP that do not coincide with the location of $LRP3$ on HRP .

- Propagate HRP to the depth of object field by using $Z = -Z_{finer3}$, predetermined wavelength and sampling period of the recording equipment ($\lambda = 600nm$ and $X = 10\lambda$ for this thesis) and the methods given in Appendix D size being $4N \times 4N$. This is the reconstruction of the object pattern resulting from the stitching of three low resolution diffraction patterns.

When we apply this approach to all the patterns $LRP1$, $LRP2$, $LRP3$ consecutively, we will have a diffraction pattern which will carry all the information in the low resolution patterns, hence it is the resultant high resolution diffraction pattern. There are two basic problems to handle in this approach: First one is

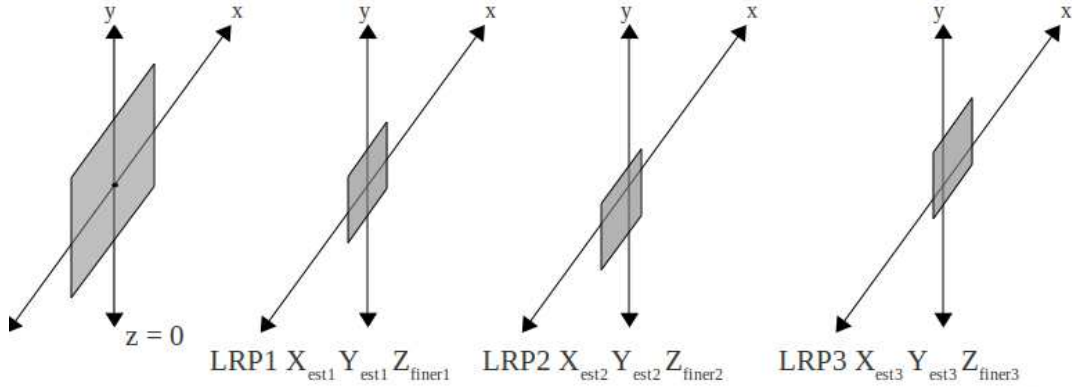


Figure 5.15: Parallel plane stitching back and forth

the interpolation issue. All the recorded patterns are discretizations of a continuous wave at a fixed location. Interpolation may be required while merging these low resolution recordings due to subpixel inaccuracy of these discretizations. The second problem is the relative x, y position problem between different diffraction patterns. Since it is not easy to get a fine estimate of x, y locations in registration process, these iterations are left to the stitching process. The relative x, y position is a critical parameter for stitching process because relative positions of different recordings play an important role while merging the patterns.

Interpolation problem occurs due to the discretization of the problem in the sensors. The recording cameras have some fixed resolution and fixed pixel positions. Due to this discretization, it is natural to have subpixel inaccuracy between low resolution recordings. While applying one of the patterns as a constraint to the field generated by another pattern, an interpolation may be required. However, due to the nature of the diffraction patterns, any rough interpolator may be good enough to address this problem. In this study a zeroth order interpolator is used when it is required in stitching process. A zeroth order interpolator places the value of the nearest pixel to the desired location in the final grid. An example of a zeroth order interpolator is shown in Figure 5.16.

Second problem is the error in the x, y estimates in registration process. The result of a large error is shown in Figure 5.17. As it is clear from the figure,

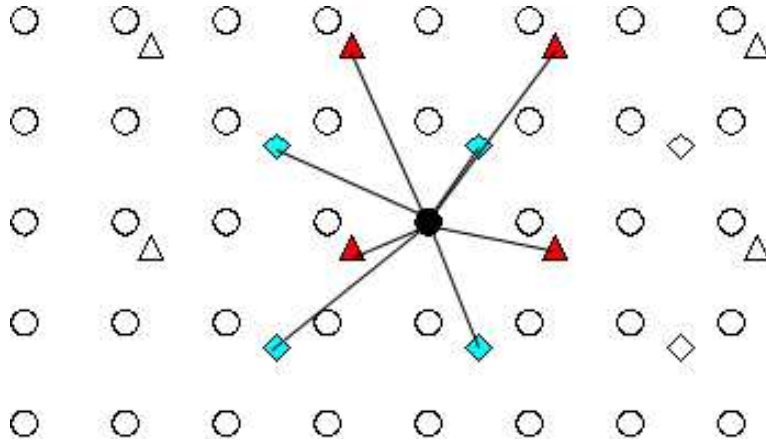


Figure 5.16: Zeroth order interpolator. All the relevant distances are calculated in a low resolution pattern and the nearest neighbor is directly used in high resolution pattern. In the figure, circular pixels are from the high resolution resultant pattern and the triangular and diamond shaped ones are from different low resolution patterns.

a mismatch in relative x, y locations results in a virtual object near the real object. As it is clear in the Figure 5.17, this virtual object degrades the sharpness of the real object as it was in the wrong depth estimation case explained in Section 5.1.2. So, moving local variance filter is a good candidate to solve this problem since it has already helped in inaccurate depth estimation problem. We will iterate the position of the object from the second pattern (i.e. $LRP2$ in Figure 5.15) in a small neighborhood and try to maximize the sharpness of the reconstructed pattern (i.e. the entire object function generated by both $LRP1$ and $LRP2$ in the Figure 5.15) with the moving local variance filter given by Equation 5.3. The result of the variance filter is shown in Figure 5.18 and the resultant reconstruction when the variance is maximum is shown in Figure 5.19.

5.3 Overall System

This section explains the overall system formed by registration and stitching processes. Figure 5.20 shows a block diagram representation of the overall system.

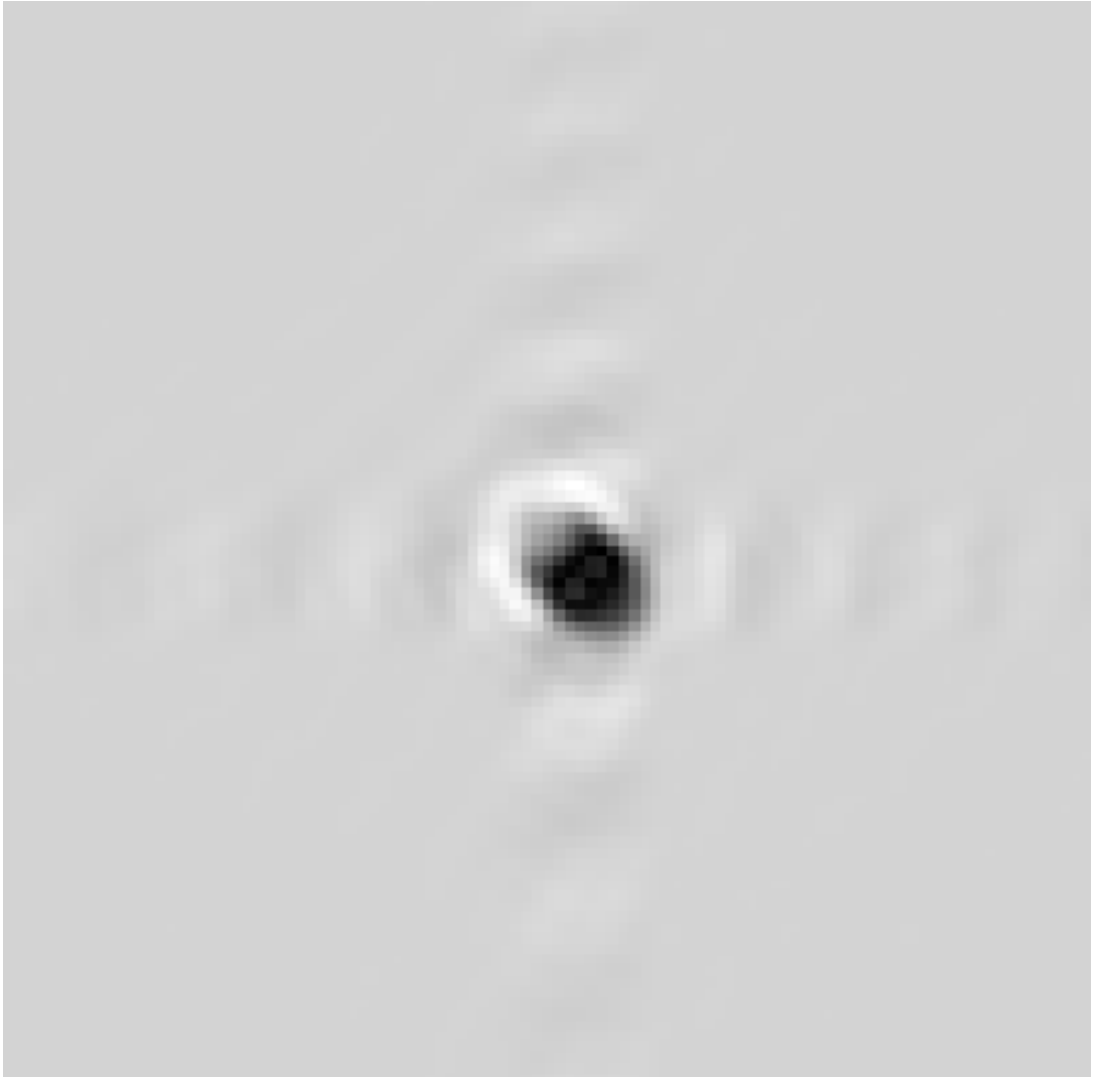


Figure 5.17: The pattern generated when there is a mismatch in x,y positions of the diffraction patterns. The size of the original pattern is 512×512 . This is a zoomed version of the original pattern. The vertical axis is n -axis going from 206 to 306 from top to down and horizontal axis is m -axis going from 206 to 306 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel.

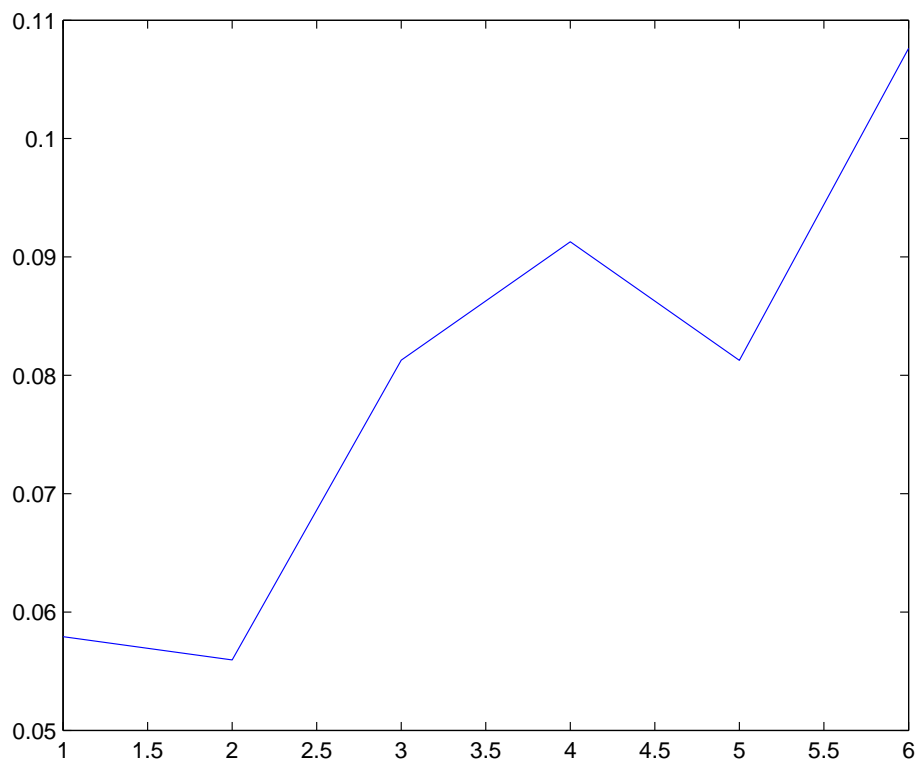


Figure 5.18: The variance filter result of the mismatch problem. The vertical axis represents the maximum local variance in each iteration. Horizontal axis is the iteration number called s which satisfies $k = 6 - s$ where k is the error in the position of the object in pixel units.

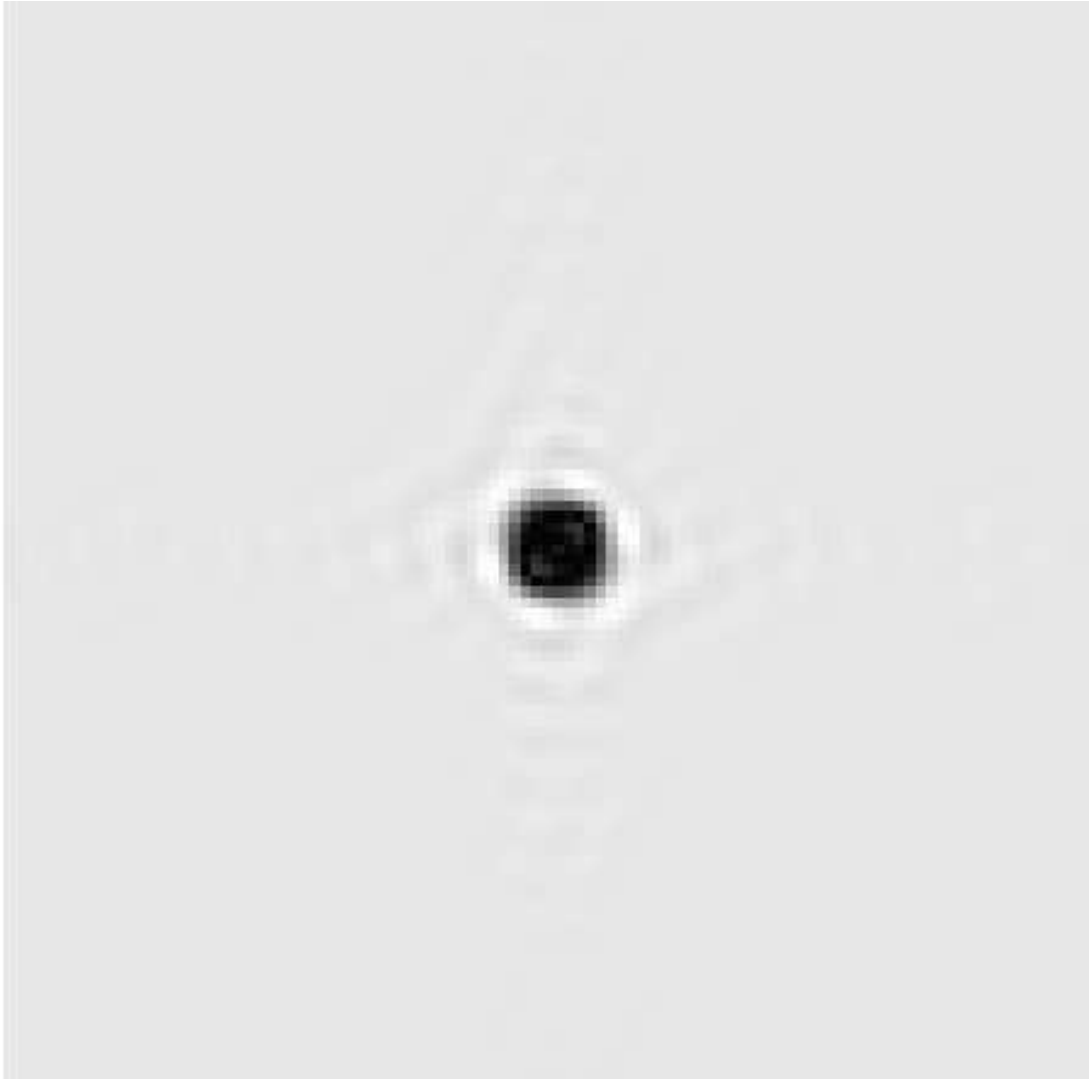


Figure 5.19: The final reconstruction of the iteration process. The size of the original pattern is 512×512 . This is a zoomed version of the original pattern. The vertical axis is n -axis going from 206 to 306 from top to down and horizontal axis is m -axis going from 206 to 306 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel.

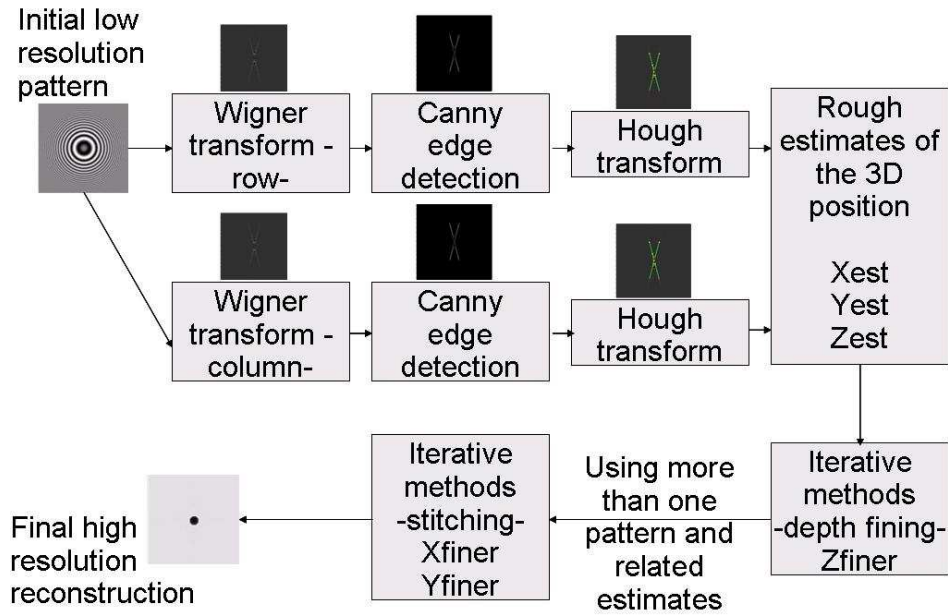


Figure 5.20: The block diagram of the overall system.

The input of the overall system is a low resolution diffraction pattern (depicted as "Initial low resolution pattern" in the figure). Since Wigner transform of a 2D function yields a 4D result, we prefer to apply Wigner transform on 1D slices of the original 2D function to get 2D results (easier to visualise and interpret). This property of Wigner transform is also explained in Chapter 4. For a Wigner transform of a 1D function, the slope of the Wigner lines is related to the distance between the object and the recording equipment and the intersection point of the Wigner lines gives the 1D position of the object in the recording. Hence, an easy choice for possible 1D slices from the initial low resolution diffraction pattern is to select a row and a column of the pattern (easy since there is no need for interpolation, all the samples are equidistance). In this case, intersection point in the Wigner transform of the row function will give information about the x-position of the object in the overall recording and intersection point in the Wigner transform of the column function will give information about the y-position of the object in the overall recording. The slopes of all the lines in both Wigner transform domains (both for row and column functions) will carry

information about the distance between the object and the recording equipment. To sum up, we need to compute two Wigner transforms (i.e. one for a row function and one for a column function) in parallel in order to be able to extract 3D object position in a recording. Indeed, we need to compute two parallel sets of Wigner transform, Canny edge detection and Hough transform for 1D functions grabbed from the initial recording to determine first rough estimates for object position as shown in the upper part of the Figure 5.20. A sample result for each step is shown above the related block for illustrative purposes. The output of the Wigner transform is a real valued matrix. Throughout this thesis, these patterns are scaled to be between $[0 \ 255]$ for printing purposes. The line impulses of the Wigner transform are depicted as white and the contrast between these lines and the rest of the pattern is used to extract these lines by edge detection as explained in Chapter 4. The input of the Canny edge detector is the direct output of the Wigner transform and the output of this detector is a binary pattern in which the Wigner lines are emphasized as white on a black background. Since the input of the Hough transform should be a binary pattern, the output of the Canny edge detector is directly fed to Hough transform. The output of the Hough transform is the accumulator matrix and the Wigner lines are extracted by the analysis of the peaks in this accumulator matrix as explained in Appendix G. The detected lines superposed onto the original Wigner transform patterns are shown as the output of the Hough transform during this thesis. At the end of the application of these methods onto a row and a column function, we have the rough estimates for 3D position of the object in the low resolution recording as shown by $(X_{est}, Y_{est}, Z_{est})$ in the Figure 5.20. It is possible to call the blocks from the beginning of the algorithm up to this point as the registration part of the algorithm. In this part, rough estimates of the object position is gained with no prior knowledge about the position of the recording equipment and using only

the methods explained above. After registration, an extra step takes place before stitching in which it is possible to make distance estimate finer with iterative methods.

All the blocks after registration use iterative methods (i.e. variance filter explained by Equation 5.3). These iterative methods are guaranteed to success if a limited amount of error is not exceeded in registration process. This amount is determined as if 10 pixels in x, y locations and no more error than 5% in z locations for this thesis. All the iterative methods after registration are designed to handle errors up to these maximum levels in this thesis.

For depth fining, three different depth ranges are defined as explained in Subsection 5.1.2 according to this maximum error level of 5%. A reconstruction for each distance is calculated and the distance which yields the sharpest reconstruction (i.e. reconstruction with the maximum local variance) is selected as the finest distance estimate for that range. The details of this algorithm are also explained in Subsection 5.1.2.

After this block, the stitching process starts. Indeed, it is more convenient to name this process as the stitching and x, y registration fining process. As explained before in this thesis, it is not possible to make x, y estimates finer by making use of only one low resolution pattern. We do not have enough information to set up an iterative method to make x, y estimates finer for a single recording. The solution to this problem is the relative fining of the 2D object position is two or more low resolution recordings. This process is merged into the stitching process since the difference between the result of a finely registered pair and a bad registered pair is easily spottable after stitching. The result of a bad registered pair is a virtual object as explained before and this kind of a result degrades the sharpness of the overall reconstruction. To handle this problem, we assume to have a correct x, y estimate for one of the low resolution recordings.

Then we move the second recording within a neighbourhood defined by the predetermined maximum error to find the x, y location which yields the maximum local variance after stitching. This x, y position is recorded as the finest x, y estimates for the second low resolution recording. Applying this process to all the low resolution recordings concludes the stitching and x, y registration processes together with the overall system. It is possible to form a higher resolution reconstruction after this point. Since the stitching performs successfully with the given error limits, simulations are conducted up to this point and the estimates which stay within these limits are accepted to be successful.

Chapter 6

Simulation Results

6.1 A Complete Registration and Stitching

In this chapter, simulation results of the algorithms described in previous chapters are presented. The simulation starts with five different diffraction patterns with no prior information about their 3D positions in space. We only know that all the diffraction patterns were recorded from the same light distribution. These five low resolution diffraction patterns are shown in Figures 6.1, 6.4, 6.7, 6.10 and 6.13. These patterns are obtained as follows: A high resolution pattern is calculated with the given techniques in Chapter 3 in Section 3.2. The object pattern is generated as it is explained in Appendix C with radius $r = 4.5$ pixels and the object being at the center of the pattern (i.e. $(Y_c, X_c) = (1024, 1024)$). The size of this high resolution pattern is 2048×2048 . In our simulations, the low resolution initial patterns which are shown in the figures were actually extracted from a high resolution diffraction pattern. This is done to have a ground truth reference to check the simulation results. Their sizes are all 512×512 . High resolution input pattern is propagated to five different depth positions as it is explained in Appendix D and different 512×512 parts of this pattern are

recorded as low resolution diffraction patterns. The wavelength of the beam, λ , is taken as 600 nanometers. The pixel size of the recording sensor array is assumed to be ten times the wavelength of the beam ($X = 10\lambda$). The intermediate results of the registration process which is held by applying Wigner transform as explained by Appendix E, edge detection as explained in Appendix F and Hough transform as explained in Appendix G, as it is explained in Chapters 4 and 5 to each low resolution pattern, are shown in Figures 6.2, 6.5, 6.8, 6.11 and 6.14. The input of the Wigner transform is any row or column of the low resolution diffraction pattern. The output of Wigner transform is fed into edge detection algorithm to emphasize the edges. Canny edge detection algorithm is used to detect edges in the Wigner transform pattern as the details of the algorithm are given in Appendix F. The low threshold of the Canny edge detector is selected as 0.1 where the high threshold is selected to be 0.5. The output binary pattern generated by Canny edge detection as mentioned in Appendix F is the input pattern for Hough transform. The aim of Hough transform is to find lines in the binary pattern generated by the edge detector. Hough transform is applied as given in Appendix G with a ρ resolution of 0.5 and θ resolution of 0.1. In the Hough transform domain, maximum 8 peaks are tried to be detected which are greater than the 0.8 times the global maximum of the accumulator matrix. The (5×5) neighbourhood of any possible peak is cleared not to select the same peak by numerical errors. The space between two line segments is filled if they are on the same line. The reconstructions generated by the parameters estimated in registration process are shown in Figures 6.3, 6.6, 6.9, 6.12 and 6.15. As a final step, all the results gained from the stitching process as it is explained in Chapter 5 in Section 5.2 are shown in Figure 6.16.

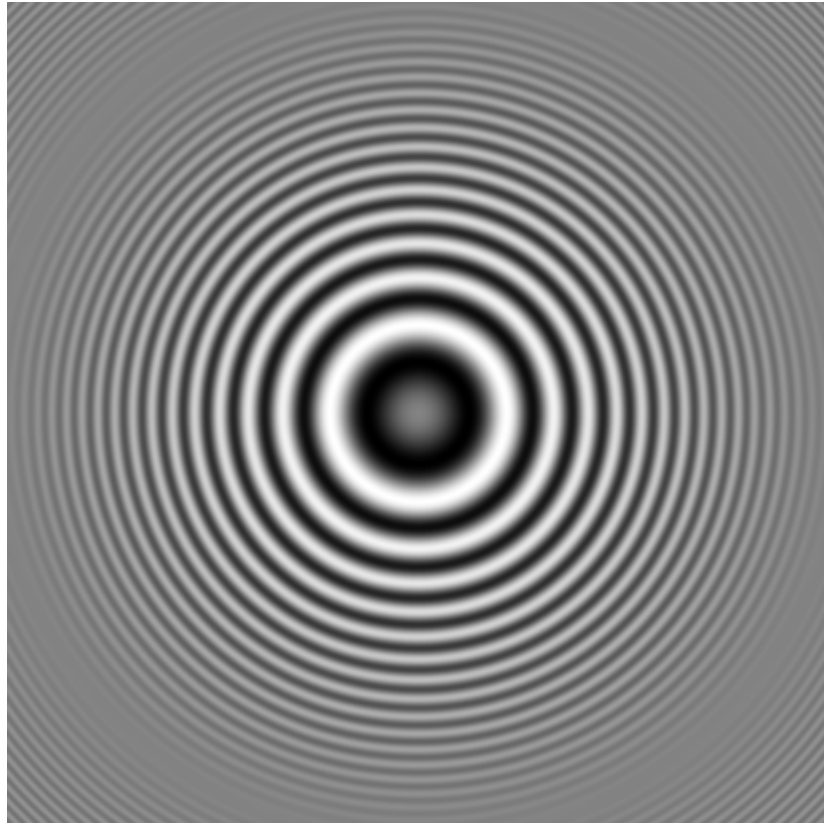


Figure 6.1: Low resolution pattern no.1. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

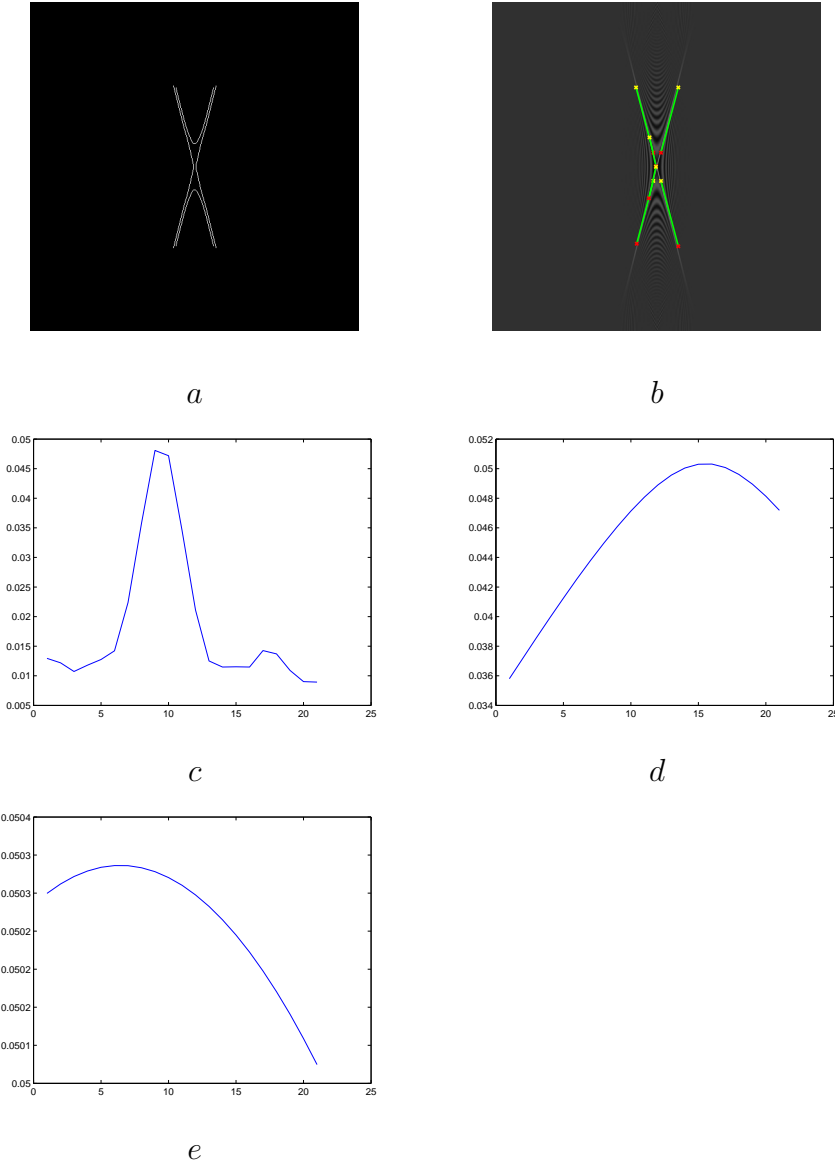


Figure 6.2: Edge detection and Hough transform results of the low resolution pattern no.1. Only the results from the Wigner transform of the column function are given. a) The binary output pattern of the Canny edge detector applied onto the Wigner transform of a 1D slice of the given diffraction pattern no.1. b) Lines detected by the Hough transform from the pattern shown in a superposed onto Wigner transform pattern. Both figures are given as an example, this is done for both row and column slices. c) The result for the first step of the variance filter d) The result for the second step of the variance filter e) The result for the third step of the variance filter. (The details of this filter is explained in Section 5.1.2) Axis properties: For (a) and (b); the horizontal axis is the frequency variable $\theta = \pi p/N$ where integer $p \in [-255, 256]$. The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is p -axis going from -255 to 256 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0, 255]$, 255 being the brightest pixel in the figure. For (c),(d),(e); the vertical axis represents the maximum local variance in each iteration. Horizontal axis is the iteration number called s which satisfies $d = z_{est} + (s - 11)\Delta_z$ as explained in Subsection 5.1.2.

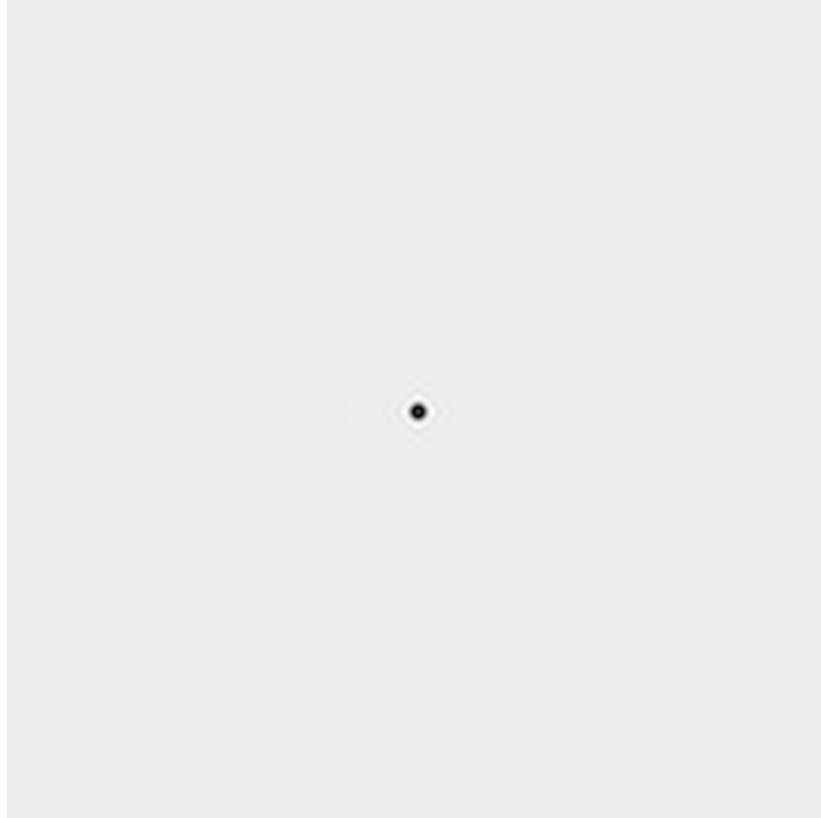


Figure 6.3: Reconstruction of low resolution pattern no.1 with the parameters estimated after the iterative process. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

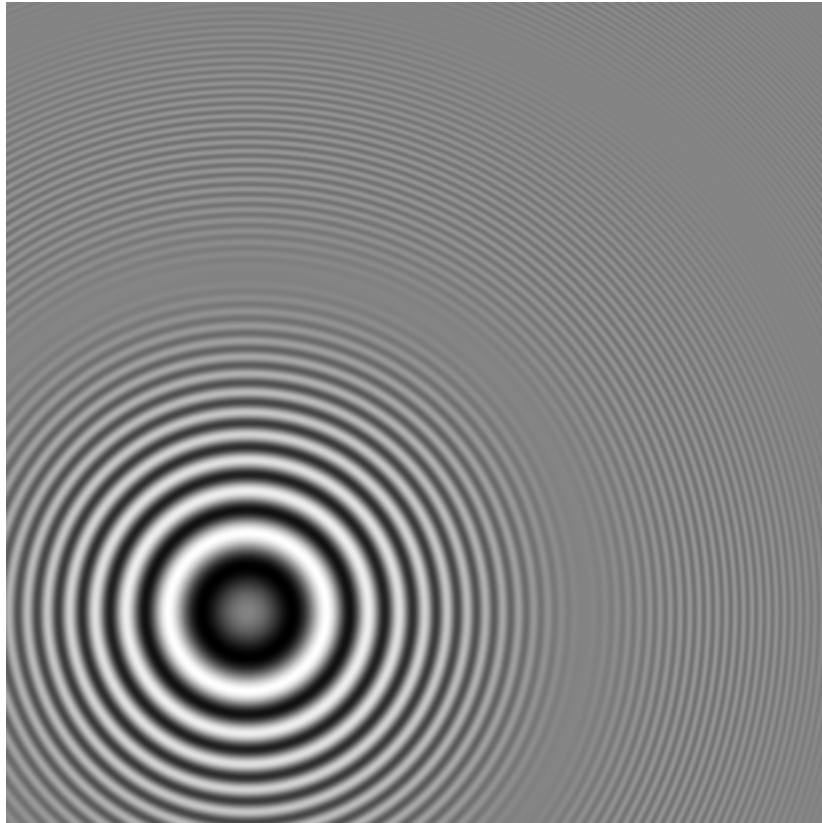


Figure 6.4: Low resolution pattern no.2. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

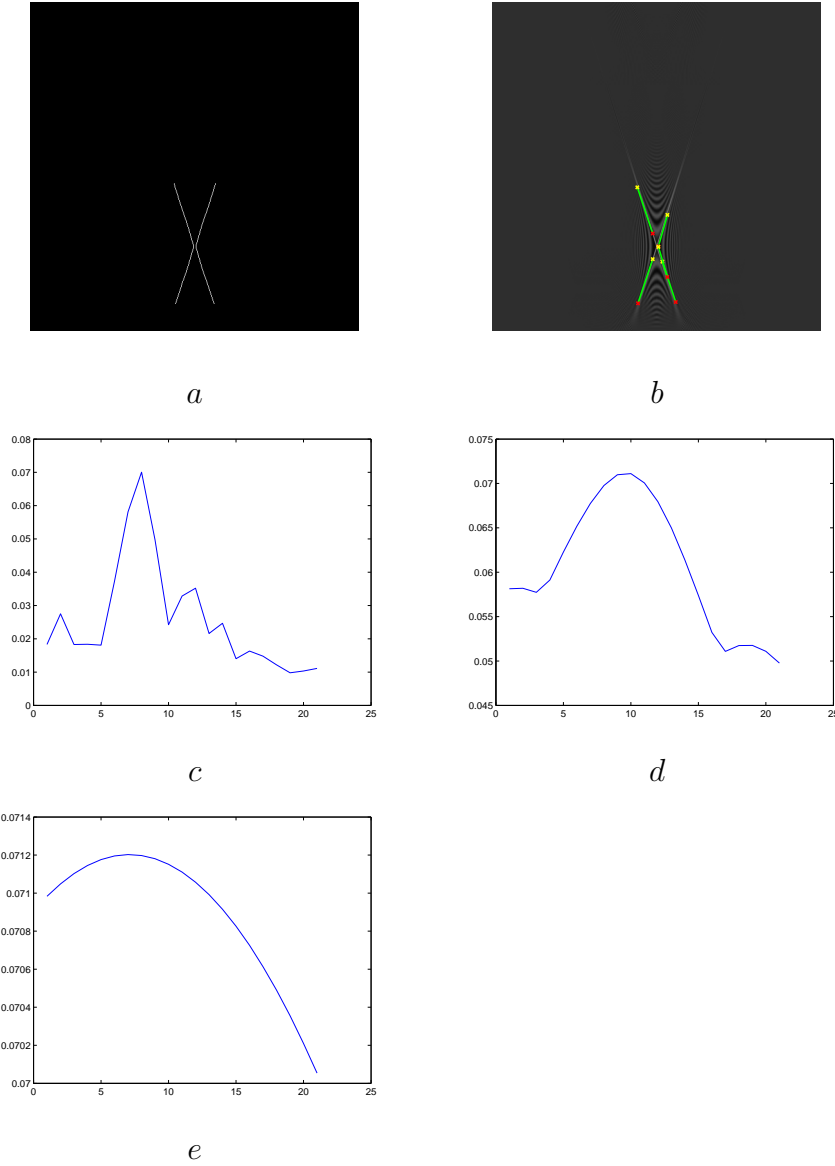


Figure 6.5: Edge detection and Hough transform results of the low resolution pattern no.2. Only the results from the Wigner transform of the column function are given. a) The binary output pattern of the Canny edge detector applied onto the Wigner transform of a 1D slice of the given diffraction pattern no.2. b) Lines detected by the Hough transform from the pattern shown in a superposed onto Wigner transform pattern. Both figures are given as an example, this is done for both row and column slices. c) The result for the first step of the variance filter d) The result for the second step of the variance filter e) The result for the third step of the variance filter. (The details of this filter is explained in Section 5.1.2) Axis properties: For (a) and (b); the horizontal axis is the frequency variable $\theta = \pi p/N$ where integer $p \in [-255, 256]$. The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is p -axis going from -255 to 256 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0, 255]$, 255 being the brightest pixel in the figure. For (c),(d),(e); the vertical axis represents the maximum local variance in each iteration. Horizontal axis is the iteration number called s which satisfies $d = z_{est} + (s - 11)\Delta_z$ as explained in Subsection 5.1.2.



Figure 6.6: Reconstruction of low resolution pattern no.2 with the parameters estimated after the iterative process. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

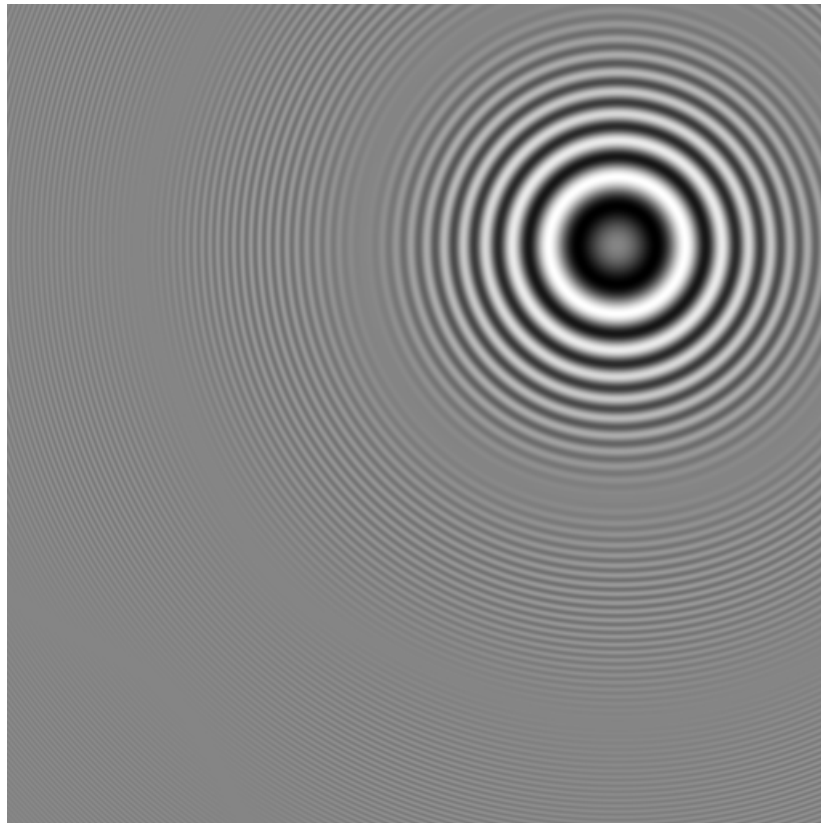


Figure 6.7: Low resolution pattern no.3. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

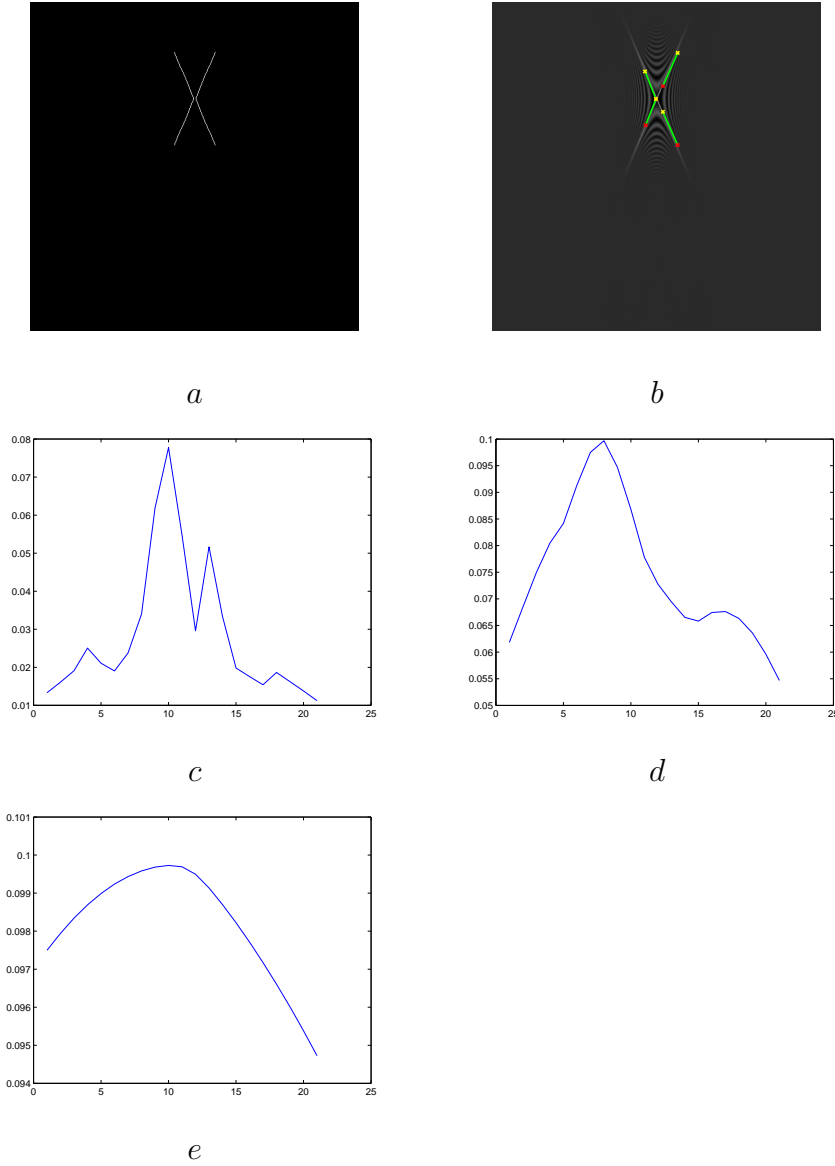


Figure 6.8: Edge detection and Hough transform results of the low resolution pattern no.3. Only the results from the Wigner transform of the column function are given. a) The binary output pattern of the Canny edge detector applied onto the Wigner transform of a 1D slice of the given diffraction pattern no.3. b) Lines detected by the Hough transform from the pattern shown in a superposed onto Wigner transform pattern. Both figures are given as an example, this is done for both row and column slices. c) The result for the first step of the variance filter d) The result for the second step of the variance filter e) The result for the third step of the variance filter. (The details of this filter is explained in Section 5.1.2) Axis properties: For (a) and (b); the horizontal axis is the frequency variable $\theta = \pi p/N$ where integer $p \in [-255, 256]$. The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is p -axis going from -255 to 256 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0, 255]$, 255 being the brightest pixel in the figure. For (c),(d),(e); the vertical axis represents the maximum local variance in each iteration. Horizontal axis is the iteration number called s which satisfies $d = z_{est} + (s - 11)\Delta_z$ as explained in Subsection 5.1.2.

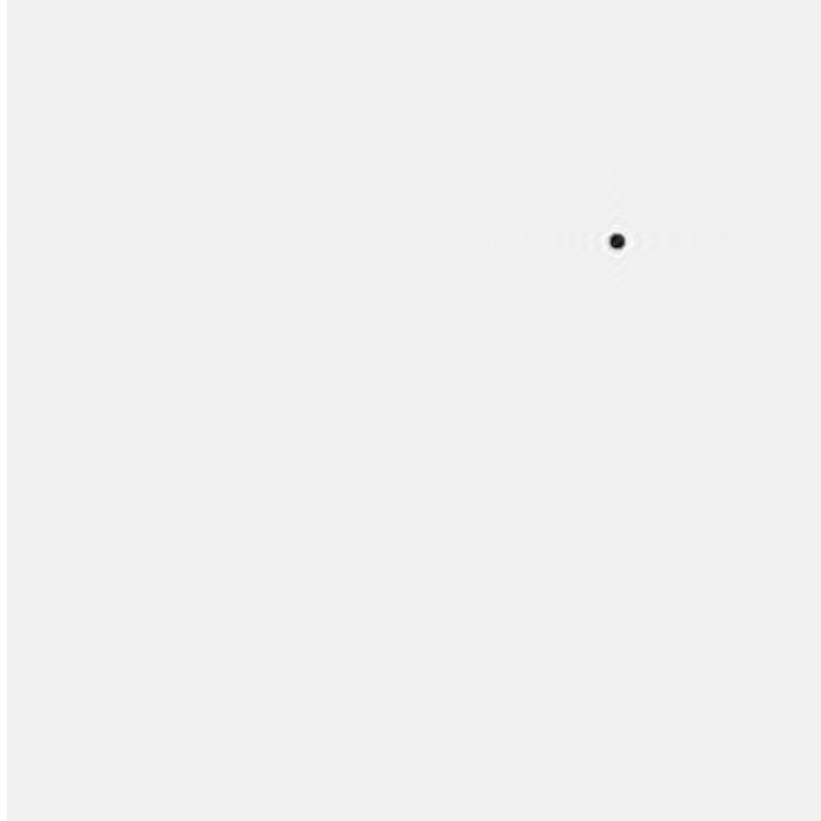


Figure 6.9: Reconstruction of low resolution pattern no.3 with the parameters estimated after the iterative process. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

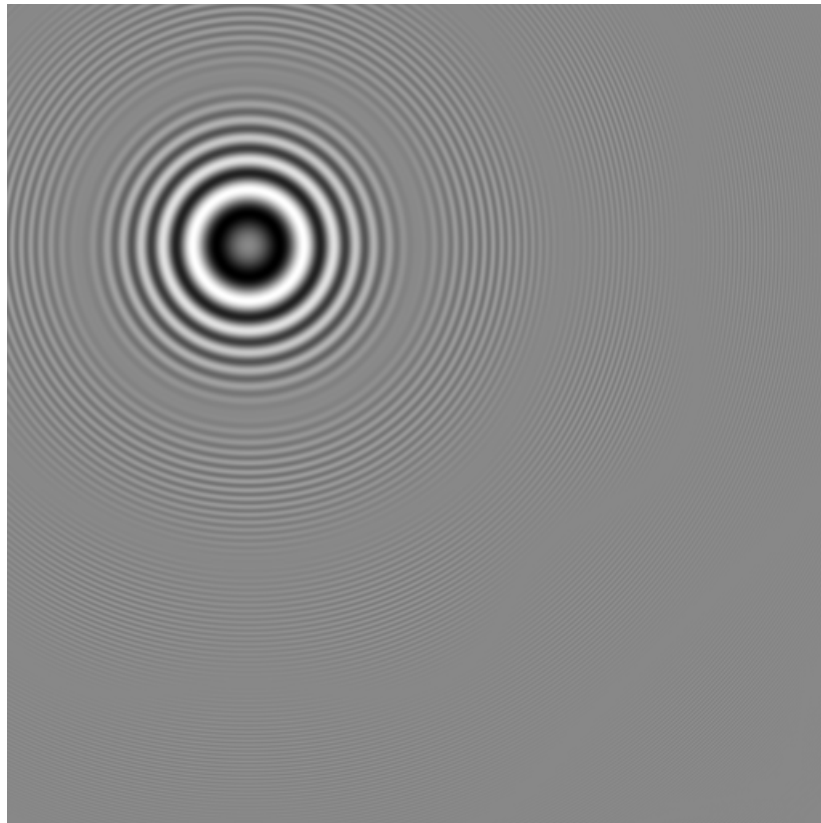


Figure 6.10: Low resolution pattern no.4. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

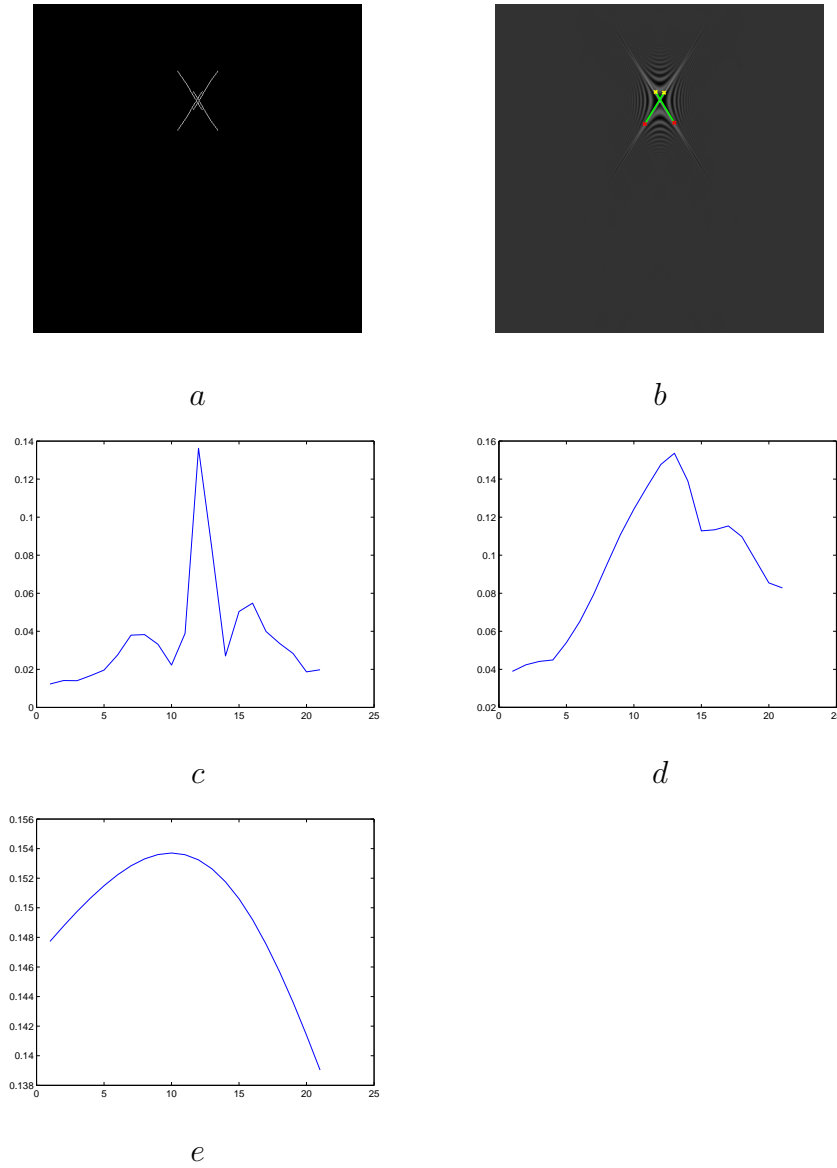


Figure 6.11: Edge detection and Hough transform results of the low resolution pattern no.4. Only the results from the Wigner transform of the column function are given. a) The binary output pattern of the Canny edge detector applied onto the Wigner transform of a 1D slice of the given diffraction pattern no.4. b) Lines detected by the Hough transform from the pattern shown in a superposed onto Wigner transform pattern. Both figures are given as an example, this is done for both row and column slices. c) The result for the first step of the variance filter d) The result for the second step of the variance filter e) The result for the third step of the variance filter. (The details of this filter is explained in Section 5.1.2) Axis properties: For (a) and (b); the horizontal axis is the frequency variable $\theta = \pi p/N$ where integer $p \in [-255, 256]$. The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is p -axis going from -255 to 256 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0, 255]$, 255 being the brightest pixel in the figure. For (c),(d),(e); the vertical axis represents the maximum local variance in each iteration. Horizontal axis is the iteration number called s which satisfies $d = z_{est} + (s - 11)\Delta_z$ as explained in Subsection 5.1.2.

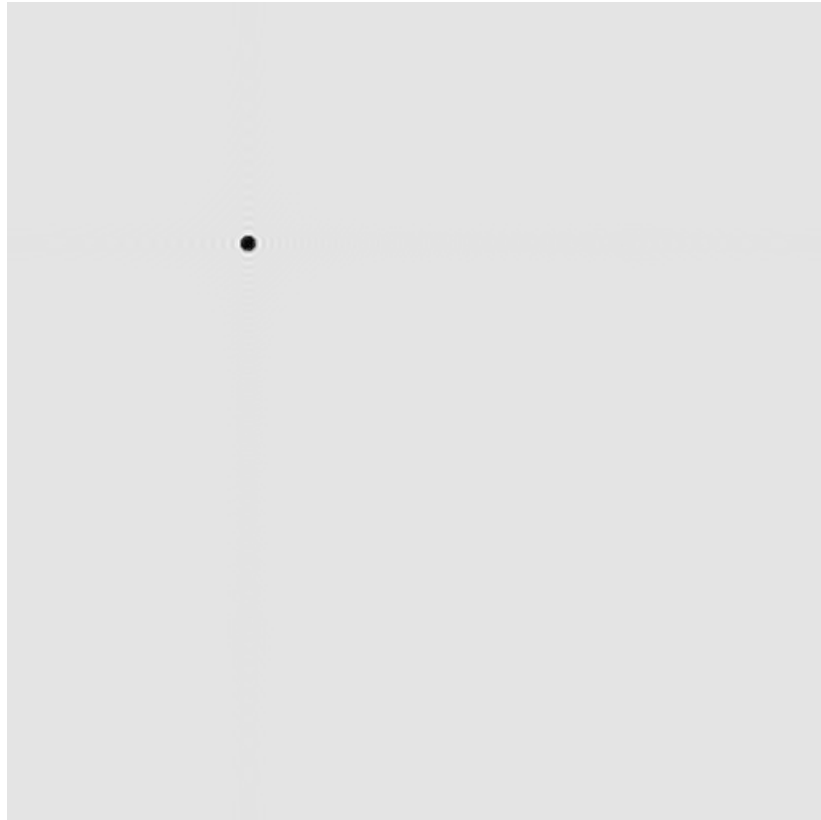


Figure 6.12: Reconstruction of low resolution pattern no.4 with the parameters estimated after the iterative process. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

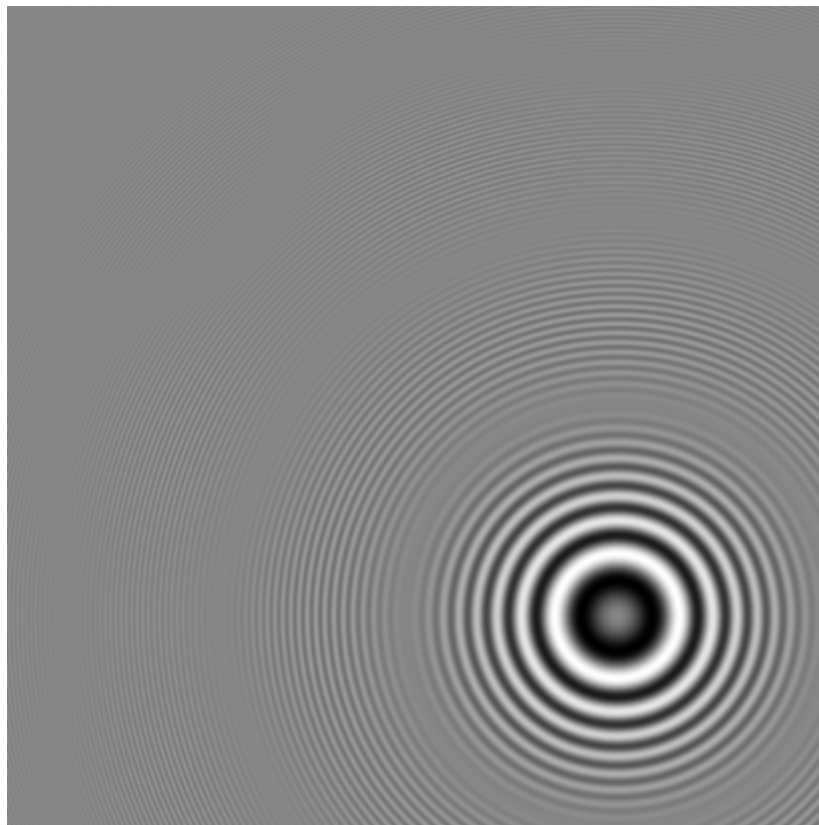


Figure 6.13: Low resolution pattern no.5. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

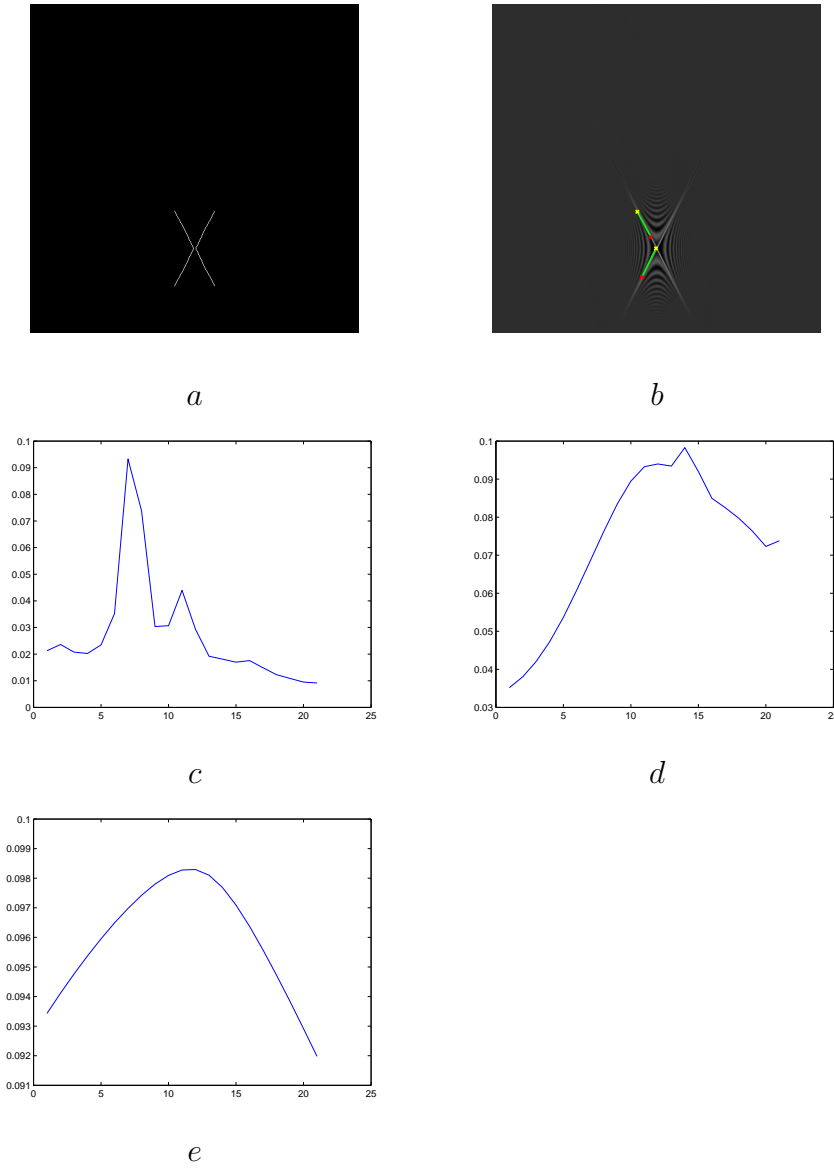


Figure 6.14: Edge detection and Hough transform results of the low resolution pattern no.5. Only the results from the Wigner transform of the column function are given. a) The binary output pattern of the Canny edge detector applied onto the Wigner transform of a 1D slice of the given diffraction pattern no.5. b) Lines detected by the Hough transform from the pattern shown in a superposed onto Wigner transform pattern. Both figures are given as an example, this is done for both row and column slices. c) The result for the first step of the variance filter d) The result for the second step of the variance filter e) The result for the third step of the variance filter. (The details of this filter is explained in Section 5.1.2) Axis properties: For (a) and (b); the horizontal axis is the frequency variable $\theta = \pi p/N$ where integer $p \in [-255, 256]$. The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is p -axis going from -255 to 256 from left to right as shown in Appendix A. The dynamic range of the pattern is scaled to interval $[0, 255]$, 255 being the brightest pixel in the figure. For (c),(d),(e); the vertical axis represents the maximum local variance in each iteration. Horizontal axis is the iteration number called s which satisfies $d = z_{est} + (s - 11)\Delta_z$ as explained in Subsection 5.1.2.

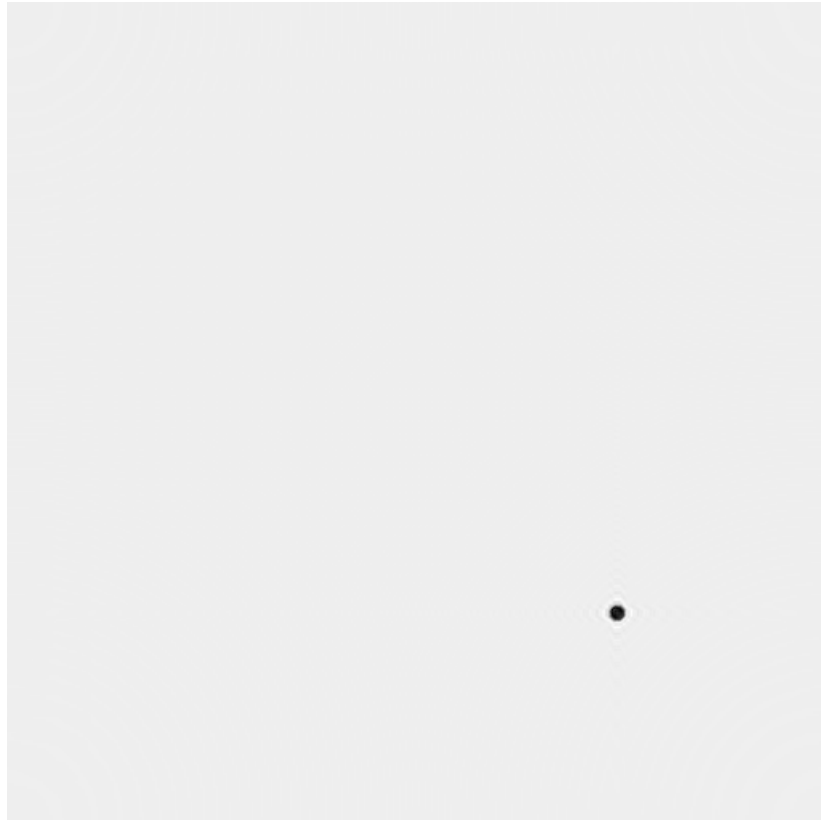


Figure 6.15: Reconstruction of low resolution pattern no.5 with the parameters estimated after the iterative process. Its size is (512×512) . The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure.

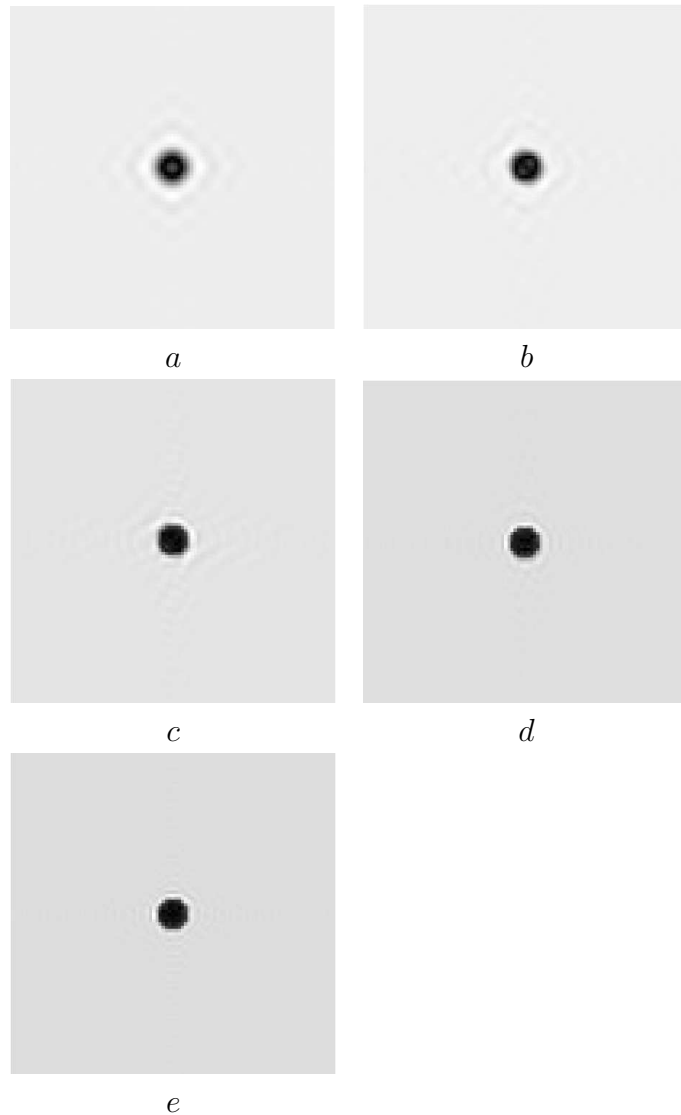


Figure 6.16: The results from the stitching process (they are zoomed to make details clear). All the figures obey the figure properties described in A, their sizes are (100×100) a) The reconstruction of the single diffraction pattern given in 6.1. The resultant pattern is a low pass pattern with blurring effect due to non-existent high frequency components of the fringe pattern in Figure 6.1 because of the resolution of the sensor array. b) The reconstruction of a higher resolution pattern formed by patterns shown in Figures 6.1 and 6.4. There is less blurring effect due to enhancement in resolution and higher frequency components coming from the pattern shown in Figure 6.4. c)d)e) The reconstruction of the patterns formed by stitching of 3,4 and 5 low resolution patterns shown in Figures 6.1, 6.4, 6.7, 6.10, 6.13. As the number of the merged patterns increases the sharpness of the reconstructed object also increases due to increasing resolution and information content. Axis properties: The original patterns are 512×512 in size. These are the zoomed versions of the original patterns. The vertical axis is n -axis going from 206 to 306 from top to down and horizontal axis is m -axis going from 206 to 306 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel.

6.2 Other Simulation Results

In Section 6.1, an example of a complete registration and stitching processes is illustrated. It is possible to divide this complete simulation into two parts as the registration part and the stitching part. It is observed that if the registration part gives results in a predetermined error range, it is possible to adjust stitching process according to these error ranges to get a good high resolution result. This result is due to the nature of stitching process which is a complete set of iterative methods. As an example, if we are guaranteed that the registration process will yield no more error than 10 pixels in x, y locations and no more error than 5% in z locations, it is possible to design the stitching process to handle these maximum errors by appropriately choosing the steps of the iterative methods. So, the performance of the overall process will simply depend on the performance of the registration process, i.e. the performance of the proposed method will simply depend on how precisely the registration process (Wigner transform, edge detection and Hough transform) can register the low resolution recordings. Some further simulations are held by registration process to be able to understand the performance of the proposed method under different situations.

6.2.1 Radius of the Object

In this simulation set, all the parameters apart from the radius of the object are kept constant. It is placed at the center of a 1024×1024 pattern and a diffraction pattern of this object at a fixed depth is generated. Then, 100 random low resolution (256×256) diffraction patterns are generated from this high resolution initial pattern and all the automated registration methods (Wigner transform, edge detection and Hough transform) are applied to find the 3D position of the object in the recordings. The way to generate these low resolution patterns is to select two random numbers in the interval $[0 \ 768]$ at each iteration step and

R (pixel)	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5
Success Rate (%)	100	100	100	97	100	95	98	91	92	89
R (pixel)	5.5	6	6.5	7	7.5	8	8.5	9	9.5	10
Success Rate (%)	89	87	79	76	62	63	47	42	38	38

Table 6.1: The results of the object radius simulations

treat these numbers as the abscissa and ordinate of the left top point of the low resolution pattern on the initial high resolution pattern. Table 6.1 shows the results of the registration simulation of this set. The corresponding success rate for each radius is the ratio of the number of successfully registered patterns due to a prior success criteria and the total number of patterns tried to be registered.

It is observed that the registration process is sensitive to the radius of the object. The weak part of the registration about object radius is the Wigner transform. The result of the Wigner transform is assumed to have two sharp white lines indicating the 3D position of the recording. Due to the sharpness of these white lines, an edge detection is applied. These sharp white lines are clearly seen in Figures 5.6 and 5.7. However, when the radius of the object increases, the diffraction pattern tends to be a nearer field pattern. And thus, these white lines lose their sharpness and become as it is shown in Figure 6.17. Also the cross effect coming from Wigner transform becomes sharper and degrades the performance of the edge detector. So it is possible to conclude that it is best to use this proposed method with the objects smaller than 6 pixels in radius if fully automated registration is desired. It is also possible to achieve larger object radius registration by the help of a human supervisor since the white lines indeed are still visible. It is experimented that by the help of human interaction it is possible to stitch objects having radius up to 35-40 pixels.

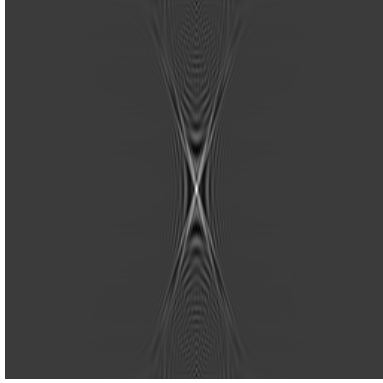


Figure 6.17: The effect of the object radius on Wigner transform

6.2.2 Noise Simulations

Noise in the Wave Field

Noisy recordings are common in digital holography due to reflections and nature of both the laser and recording sensors. In each iteration of this simulation set, all the parameters are kept as constant and a noise pattern having a different standard deviation is added to the initial light wave distribution. The initial field is complex valued with a size of 1024×1024 . The position of the object is set as $(Y_c, X_c) = (512, 512)$ and object radius is $r = 0.5$. Input function is generated as mentioned in Appendix C and propagated to $Z = NX^2/\lambda$ as explained in Appendix D where $N = 1024$, $\lambda = 600nm$ and $X = 10\lambda$. Two independent noise patterns are generated and added to both real and imaginary parts of the wave distribution. Then the intensity of this distribution is taken as the initial high resolution recording. Two different examples of the intensity of such noisy fields are given in Figure 6.18. After that 100 random low resolution patterns are grabbed from initial noisy pattern and an automated registration is applied to each of them. The way to generate these low resolution patterns is to select two random numbers in the interval $[0 \ 768]$ at each iteration and treat

these numbers as the abscissa and ordinate of the left top point of the low resolution pattern on the initial noisy high resolution pattern. The success rate in each table is the ratio of the number of the low resolution patterns that are successfully and automatically registered (having a maximum error defined in Section 6.2 to the number of total patterns in that iteration. The dynamic range of the initial high resolution patterns is [0.9975 1.0025]. Two different noise models are used in this simulation. Table 6.2 is the result of the additive Gaussian noise simulations. Noise patterns are composed of independent identically distributed pixels generated from a Gaussian noise generator with mean 0 and variance 1. They are multiplied with a noise weight in each iteration from the range $V = 0.0001k$ where $k = 1, 2, \dots, 15$ and added directly to the real and imaginary parts of the generated high resolution initial wave field. Table 6.3 shows the effect of the additive uniformly distributed noise to the performance. Noise patterns are generated by independent identically distributed uniform noise in the interval [0 1]. They are multiplied by a noise weight in each iteration from the range $V = 0.0002k$ where $k = 1, 2, \dots, 15$ and added directly to the real and imaginary parts of the generated high resolution initial wave field. For the tables, the PSNR values in dB are used to represent the relation between the original pattern and the added noise. The PSNR is calculated as

$$PSNR = 10 \log_{10} \left(\frac{MAX_I^2}{NoisePower} \right) \quad (6.1)$$

where MAX_I is the peak value of the DC adjusted original pattern (i.e. having a dynamic range of [-0.0025 0.0025] hence $MAX_I = 0.005$) and $NoisePower$ is used as the analytically calculated power of the related noise. That is, power of the noise for Gaussian case when it is added to wave field is $2\sigma_g^2 V^2$ where $\sigma_g^2 = 1$ and it is $\sigma_g^2 V^2$ when it is added to the intensity function, power of the noise for uniformly distributed case when it is added to wave field is $2\sigma_u^2 V^2$ where $\sigma_u^2 = 1/12$ and it is $\sigma_u^2 V^2$ when it is added to the intensity function.

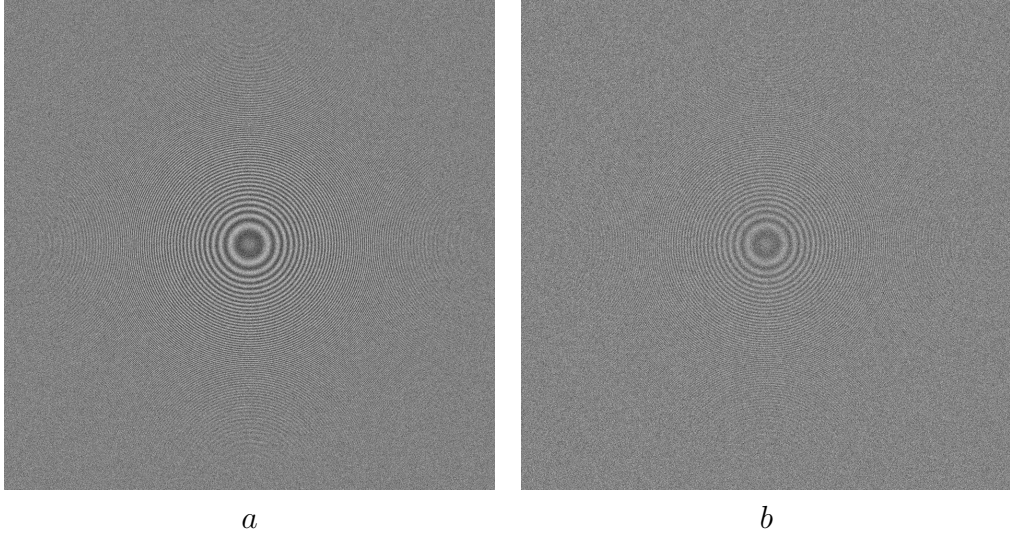


Figure 6.18: Examples of the intensity of initial noisy fields. These fields are 1024×1024 is size and are generated as explained in Section 6.2.2. The vertical axis is n -axis going from 1 to 512 from top to down and horizontal axis is m -axis going from 1 to 512 from left to right as shown in Appendix A. The dynamic range of the original pattern is scaled to interval $[0 \ 255]$, 255 being the brightest pixel in the figure. (a) has a PSNR value of 12.62 dB and (b) has a PSNR value of 6.59 dB. Both patterns are generated using a Gaussian distributed noise.

PSNR (dB)	26.60	20.58	17.06	14.55	12.60
Success Rate (%)	100	100	100	100	100
PSNR (dB)	11.03	9.69	8.52	7.51	6.59
Success Rate (%)	100	99	86	66	26
PSNR (dB)	5.77	5.01	4.31	3.67	3.08
Success Rate (%)	7	0	0	0	0

Table 6.2: The results of Gaussian noise in high resolution wave field simulations. The dynamic range of the initial pattern is $[0.9975 \ 1.0025]$ when its intensity is considered. At each step two Gaussian random patterns with mean 0 and variance 1 are generated and added to the real and imaginary parts of the initial wave field with a prior multiplication with the noise weight. Then the intensity of this field is calculated and the difference between this intensity and the intensity of the original field is taken as noise pattern. PSNR values in the table are calculated from this noise pattern.

PSNR (dB)	25.35	19.32	15.79	13.31	11.36
Success Rate (%)	100	100	100	100	100
PSNR (dB)	9.78	8.43	7.27	6.25	5.33
Success Rate (%)	100	100	98	77	59
PSNR (dB)	4.51	3.75	3.06	2.40	1.80
Success Rate (%)	40	24	15	9	3

Table 6.3: The results of the uniformly distributed noise in high resolution wave field simulations. The dynamic range of the initial pattern is $[0.9975 \ 1.0025]$ when its intensity is considered. At each step two uniformly distributed random patterns in the interval $[0 \ 1]$ are generated and added to the real and imaginary parts of the initial wave field with a prior multiplication with the noise weight. Then the intensity of this field is calculated and the difference between this intensity and the intensity of the original field is taken as noise pattern. PSNR values in the table are calculated from this noise pattern.

Recording Noise

In the simulations above, only the noise in the original wave field was examined. However, recording sensor is another important source of noise. The noise caused by the recording sensors is an addition to the noise in the original pattern and it is independent in each recording sensor. To simulate the additional effect of the recording noise, we selected a constant original field noise weight from the Tables 6.2 and 6.3 (we chose a weight in which registration is 100% successful for both noise types in order to ease comparison of the results. Any other weight could also be selected). The initial noisy high resolution patterns (1024×1024) are generated with parameters $N = 1024$, $M = 1024$, $Yc = 512$, $Xc = 512$, $r = 0.5$, $\lambda = 600nm$, $X = 10\lambda$ and $Z = NX^2/\lambda$ as explained in Appendices C and D by an addition of two noise patterns to the real and imaginary parts of the field with a prior multiplication with the selected noise weight for each noise type ($V = 0.0005$ for both types). Then the intensity of this field is recorded as the initial high resolution pattern with field noise in it. After that, 100 random low resolution patterns (256×256) are grabbed from this noisy high resolution pattern as it is done in earlier simulations. However, apart from

PSNR (dB)	12.44	11.96	11.28	10.47	9.60
Success Rate (%)	100	100	100	100	98
PSNR (dB)	8.74	7.89	7.10	6.33	5.63
Success Rate (%)	92	74	44	16	2

Table 6.4: The results of the Gaussian recording noise simulations. The initial high resolution noisy pattern (1024×1024 , with the initial dynamic range for intensity $[0.9975 \ 1.0025]$) was generated with addition of Gaussian noise to both real and imaginary parts of the original field. These initial noise patterns (1024×1024) were generated by Gaussian noise generator with mean 0 and variance 1. They were added to the real and imaginary parts of the original field after multiplication with the selected initial noise weight (0.0005 for this simulation). The intensity of this noisy field is recorded. Then, an independent Gaussian noise pattern (256×256) was also generated with mean 0 and variance 1. It was added to the low resolution pattern, which is grabbed randomly from the intensity of the noisy initial high resolution pattern as it is described in text, with a prior multiplication with the noise weight from the range $V = 0.0002k$ where $k = 1, 2, \dots, 15$ at each simulation iteration. The PSNR values given in the table are calculated with the total noise power.

earlier simulations, this time an additional independent noise pattern of the size 256×256 is generated and added to the low resolution pattern at each iteration with a prior multiplication with the noise weight from the range $V = 0.0002k$ where $k = 1, 2, \dots, 15$. Tables 6.4 and 6.5 show the results of the recording noise simulations.

From these simulations, it is observed that the registration process performs well up to a limit of noise. For the additive Gaussian noise case in the wave field, the initial noise patterns are always generated by a Gaussian noise generator with mean 0 and variance 1. The dynamic range of the intensity of the original pattern is $[0.9975 \ 1.0025]$. The noise patterns are added to the real and imaginary parts of the initial field after a multiplication by the noise weight. The registration process of the intensity of this field is successful up to a PSNR value of 8 dB. For the additive uniformly distributed noise case in the wave field, the dynamic range of the intensity of the initial pattern is same. The noise patterns are generated by a uniformly distributed noise generator in the interval $[0 \ 1]$. The

PSNR (dB)	16.12	14.93	13.85	12.85	11.94
Success Rate (%)	100	100	100	100	100
PSNR (dB)	11.11	10.34	9.64	8.97	8.36
Success Rate (%)	99	98	78	44	9

Table 6.5: The results of the uniformly distributed recording noise simulations. The initial high resolution noisy pattern (1024×1024 , with the initial dynamic range for intensity $[0.9975 \ 1.0025]$) was generated with addition of uniformly distributed noise to both real and imaginary parts of the original field. These initial noise patterns (1024×1024) were generated by a uniformly distributed noise generator in the interval $[0 \ 1]$. They were added to the real and imaginary parts of the original field after multiplication with the selected initial noise weight (0.0005 for this simulation). The intensity of this noisy field is recorded. Then, an independent uniformly distributed noise pattern (256×256) was also generated in the interval $[0 \ 1]$. It was added to the low resolution pattern, which is grabbed randomly from the intensity of the noisy initial high resolution pattern as it is described in text, with a prior multiplication with the noise weight from the range $V = 0.0002k$ where $k = 1, 2, \dots, 15$ at each simulation iteration. The PSNR values given in the table are calculated with the total noise power.

noise patterns are again added to the real and imaginary parts of the initial field after a multiplication with the related noise weight. Registration of the intensity of this field is successful up a PSNR value of 7 dB. Results show us that registration performs well up to limited PSNR value. In the case of a recording noise, the results are similar. For those simulations given in Tables 6.4 and 6.5, a constant initial field noise is assumed to exist with a noise weight $V = 0.0005$ since under this noise weight registration can still be 100% successful as seen from Tables 6.2 and 6.3. An additional noise is added to low resolution parts grabbed from the intensity of these noisy fields as a recording noise. Registration is successful up to a PSNR value of approximately 8 dB in both cases. It cannot perform well under a noise which degrades the sharpness of the fringe patterns. As a result, the Wigner transform of such a noisy fringe pattern does not result with sharp enough Wigner lines. Hence, edge detection can not detect them and the diffraction pattern can not be registered.

6.2.3 Presence of Multiple Particles

In this simulation setup, performance of the proposed method under multiple particle scenerios is evaluated. In the problem definiton of this thesis, the object field was assumed to be consist of only one particle. Just like as it is in noisy recording case, it is more practical to have more than one particle in real scenerios. Although the algorithm is not designed to detect more than one particle, the Table 6.6 gives the performance of the method under multiple particle case.

The input pattern for these simulations is calculated as explained in Appendix C with $N = 1024$, $r = 0.5$, $(X_c, Y_c) = (512, 512)$. The diffraction pattern due to this input pattern is calculated by the methods in Appendix D with $\lambda = 600nm$, $X = 10\lambda$ and $z = NX^2\lambda$. The contribution of the extra particle is calculated with randomly determined X_c, Y_c, Z parameters at each iteration and superposed onto the pattern calculated for original input pattern. λ, X and r are the same for the extra particles.

The simulation results in Table 6.6 are evaluated when there is one extra particle in the object field different than the reference particle. The radius of the extra particle is the same as the reference particle and 3D position of the extra particle is randomly determined in each step. During this simulation the extra particle is constraint to be between the reference object and the recording equipment.

In each simulation step in Table 6.6, a diffraction pattern of the two particle system is generated with random 3D position of the extra particle (which is constrained to be between reference object and recording equipment) and 100 different random low resolution parts of this pattern is tried to be registered by automatic methods. Success rate is again the percentage of the low resolution patterns which are registered within the predetermined error rates. The total success of the proposed method is 79% (i.e. 3939 successful registrations in 5000

SIM No	SIM1	SIM2	SIM3	SIM4	SIM5
Success Rate (%)	87	99	87	79	71
SIM No	SIM6	SIM7	SIM8	SIM9	SIM10
Success Rate (%)	52	90	47	72	92
SIM No	SIM11	SIM12	SIM13	SIM14	SIM15
Success Rate (%)	80	100	99	78	88
SIM No	SIM16	SIM17	SIM18	SIM19	SIM20
Success Rate (%)	100	64	71	53	82
SIM No	SIM21	SIM22	SIM23	SIM24	SIM25
Success Rate (%)	96	83	99	75	95
SIM No	SIM26	SIM27	SIM28	SIM29	SIM30
Success Rate (%)	44	13	85	73	88
SIM No	SIM31	SIM32	SIM33	SIM34	SIM35
Success Rate (%)	89	89	75	70	89
SIM No	SIM36	SIM37	SIM38	SIM39	SIM40
Success Rate (%)	100	42	78	100	26
SIM No	SIM41	SIM42	SIM43	SIM44	SIM45
Success Rate (%)	95	76	97	90	100
SIM No	SIM46	SIM47	SIM48	SIM49	SIM50
Success Rate (%)	97	91	55	40	98

Table 6.6: The results of the multiple particle simulations with one extra particle between object and recording equipment

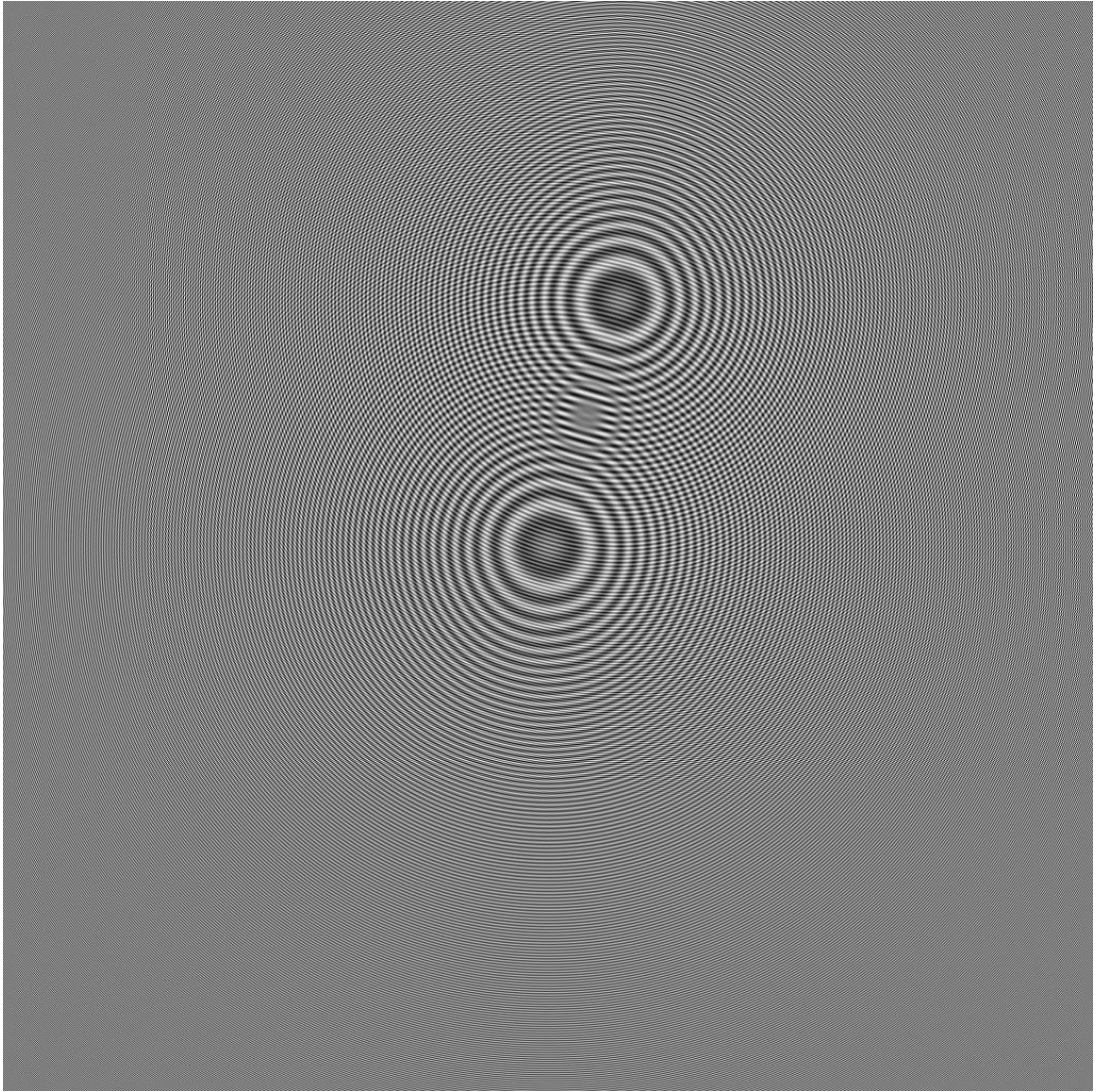


Figure 6.19: Initial pattern of the simulation 27 from Table 6.6

trials). However, Table 6.6 tells more than this. The minimum success rate occurred at simulation 27 as 13%. The Figure 6.19 shows the initial diffraction pattern which is used in simulation 27. As it is seen clearly, the extra particle is very close to reference particle hence the proposed method could not differentiate between these particles. The proposed method tries to identify the strongest edges after Wigner transform and then the longest lines in the edge detected pattern by Hough transform. So it achieves still good success rate if the extra particle is far enough from the reference particle as it is shown in Figure 6.20 which is the initial diffraction pattern in simulation 13 (with 99% success rate).

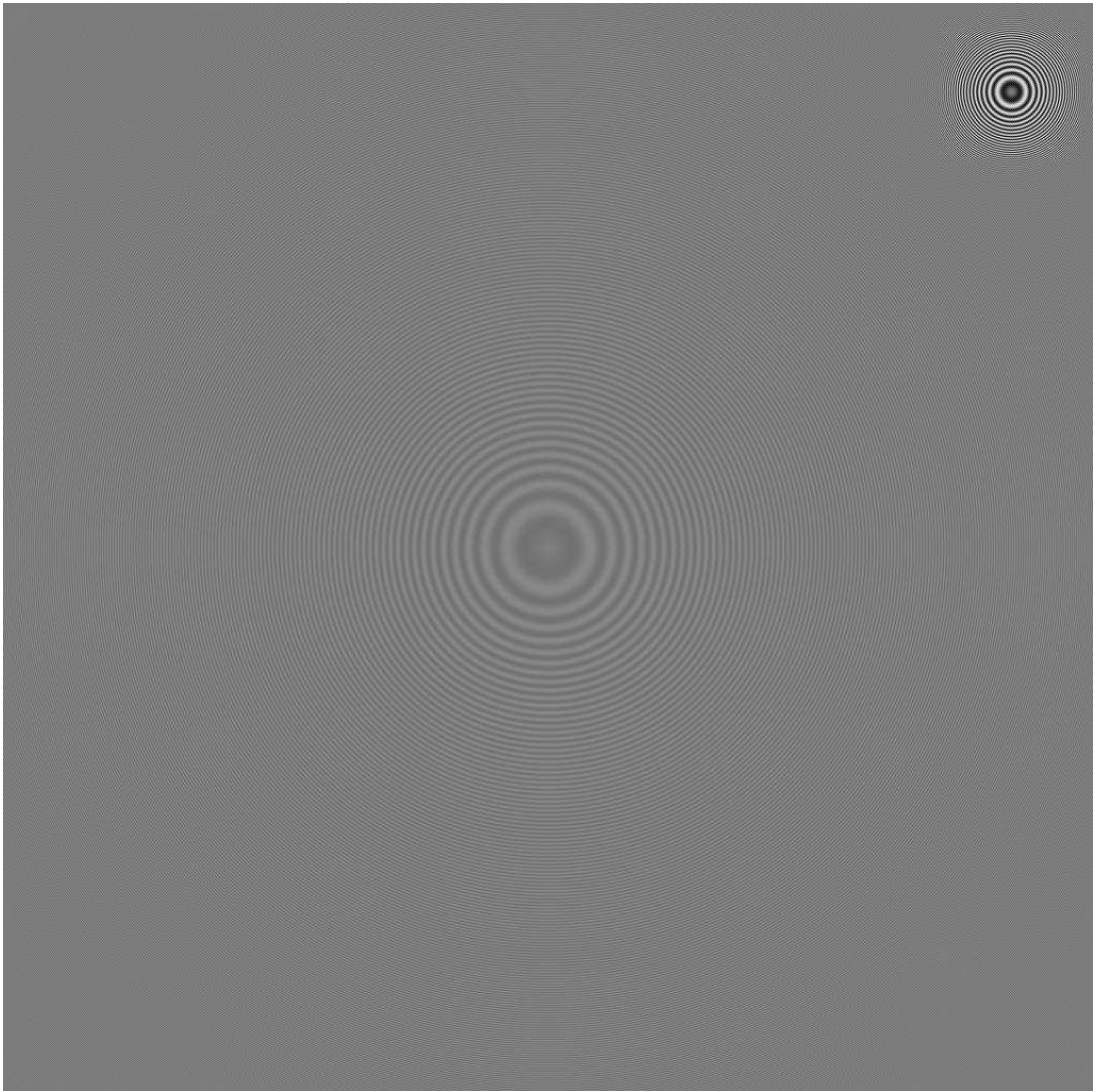


Figure 6.20: Initial pattern of the simulation 13 from Table 6.6

SIM No	SIM1	SIM2	SIM3	SIM4	SIM5
Success Rate (%)	83	85	85	92	95
SIM No	SIM6	SIM7	SIM8	SIM9	SIM10
Success Rate (%)	95	92	94	20	38
SIM No	SIM11	SIM12	SIM13	SIM14	SIM15
Success Rate (%)	94	95	88	94	93
SIM No	SIM16	SIM17	SIM18	SIM19	SIM20
Success Rate (%)	96	100	84	93	96
SIM No	SIM21	SIM22	SIM23	SIM24	SIM25
Success Rate (%)	33	59	96	96	98
SIM No	SIM26	SIM27	SIM28	SIM29	SIM30
Success Rate (%)	91	98	93	73	77
SIM No	SIM31	SIM32	SIM33	SIM34	SIM35
Success Rate (%)	71	98	96	92	93
SIM No	SIM36	SIM37	SIM38	SIM39	SIM40
Success Rate (%)	84	97	75	89	55
SIM No	SIM41	SIM42	SIM43	SIM44	SIM45
Success Rate (%)	83	98	99	92	93
SIM No	SIM46	SIM47	SIM48	SIM49	SIM50
Success Rate (%)	83	93	88	78	88

Table 6.7: The results of the multiple particle simulations with one extra particle behind object with respect to recording equipment

The simulation procedure of the Table 6.7 is same as the simulation procedure for Table 6.6 except that this time extra particle is constrained to be behind the reference object with respect to recording equipment.

All the formulation is the same as it is explained for Table 6.6. The total success rate for this simulation is 85% (4271 successful registrations in 5000 trials). All the comments are also same for this simulation as it is seen in Figure 6.21 and 6.22. If the extra particle is near to the reference particle the proposed automatic method is not well suitable to differentiate between these particles. If they are seperated well enough as it is in Figure 6.22, the proposed method still works.

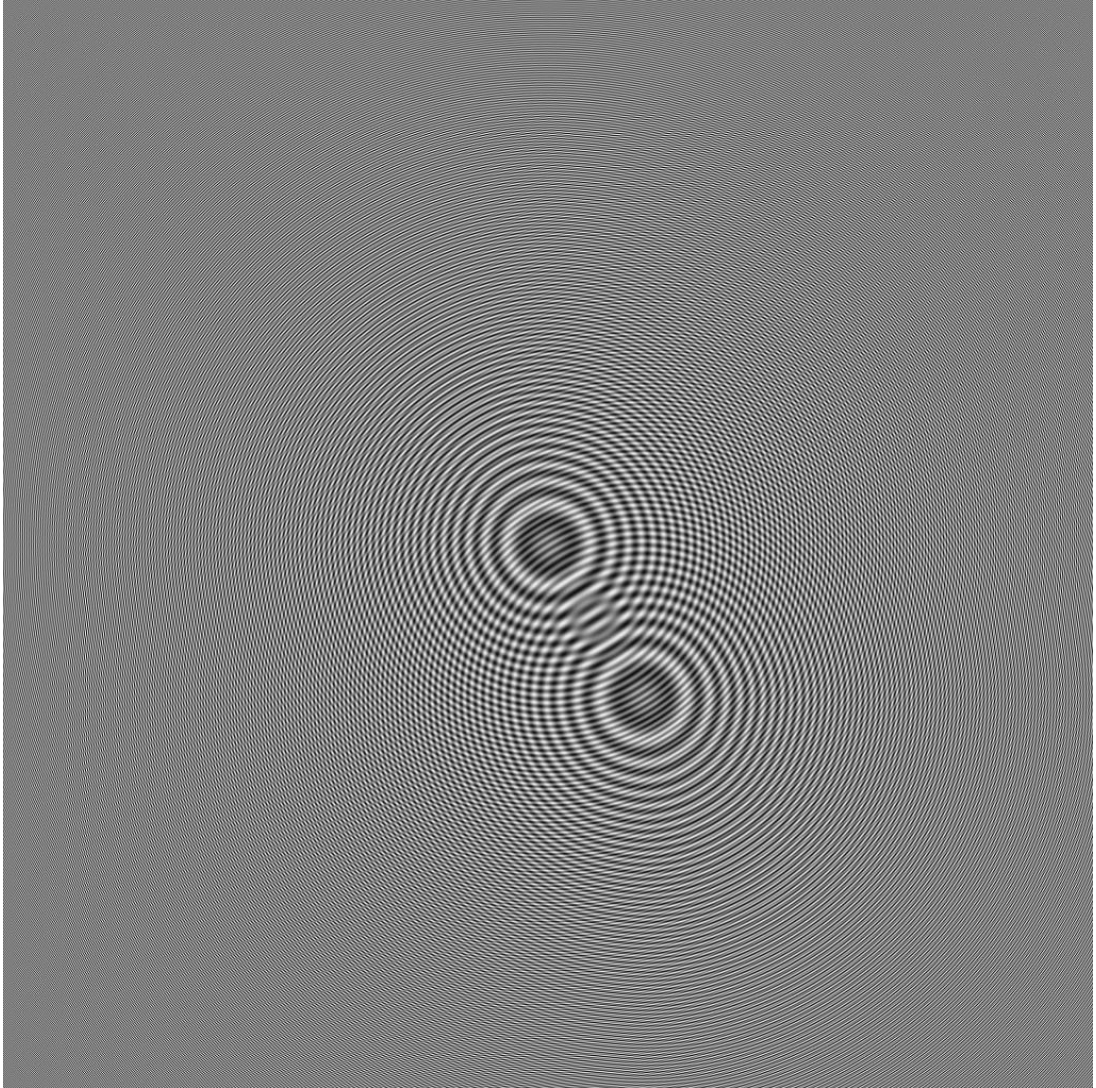


Figure 6.21: Initial pattern of the simulation 9 from Table 6.7

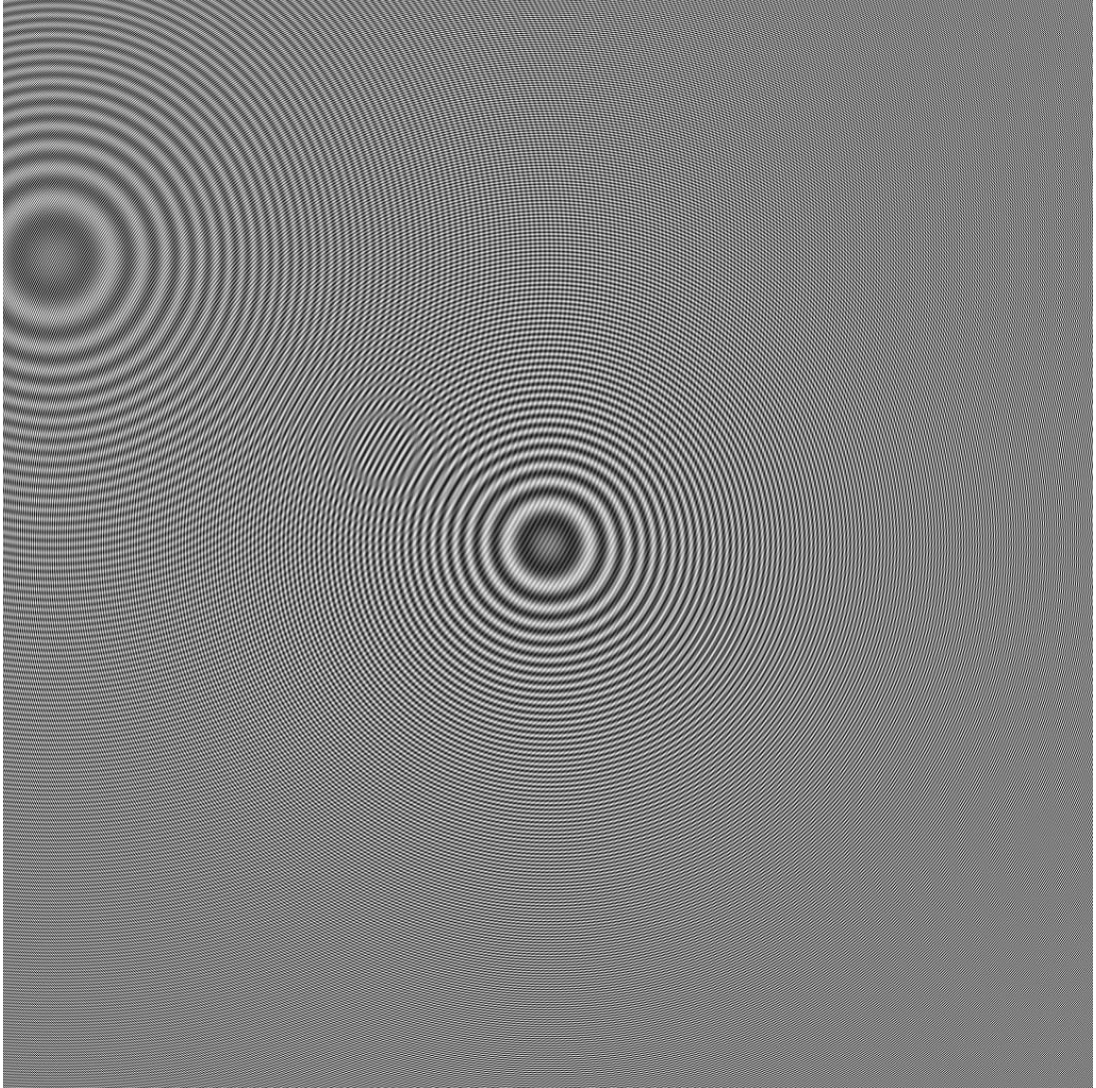


Figure 6.22: Initial pattern of the simulation 32 from Table 6.7

SIM No	SIM1	SIM2	SIM3	SIM4	SIM5
Success Rate (%)	50	57	81	17	49
SIM No	SIM6	SIM7	SIM8	SIM9	SIM10
Success Rate (%)	75	47	94	45	33
SIM No	SIM11	SIM12	SIM13	SIM14	SIM15
Success Rate (%)	10	89	47	53	80
SIM No	SIM16	SIM17	SIM18	SIM19	SIM20
Success Rate (%)	57	28	69	55	37
SIM No	SIM21	SIM22	SIM23	SIM24	SIM25
Success Rate (%)	88	89	87	46	85
SIM No	SIM26	SIM27	SIM28	SIM29	SIM30
Success Rate (%)	58	97	64	55	33
SIM No	SIM31	SIM32	SIM33	SIM34	SIM35
Success Rate (%)	58	86	65	31	8
SIM No	SIM36	SIM37	SIM38	SIM39	SIM40
Success Rate (%)	96	22	57	53	89
SIM No	SIM41	SIM42	SIM43	SIM44	SIM45
Success Rate (%)	36	36	87	32	41
SIM No	SIM46	SIM47	SIM48	SIM49	SIM50
Success Rate (%)	48	63	19	48	69

Table 6.8: The results of the multiple particle simulations with two extra particles, one is strictly behind object with respect to recording equipment and the other is randomly placed in depth

The last simulation procedure is the three particles case. One of the extra particles is guaranteed to be behind the reference particle with respect to recording equipment and the other one is randomly placed in depth, either behind the reference or in front of it. Table 6.8 gives the simulation results of this setup. Again in each simulation step a three particle diffraction pattern is generated and 100 low resolution parts of this pattern are tried to be registered by the automatic registration method proposed in this thesis.

The total success rate for three particle system is 56% i.e. 2819 successful registrations in 5000 trials. It is clear that as the number of particles increase the performance of the proposed method decreases seriously since it is more likely to have an extra particle near the reference object as the number of extra particle

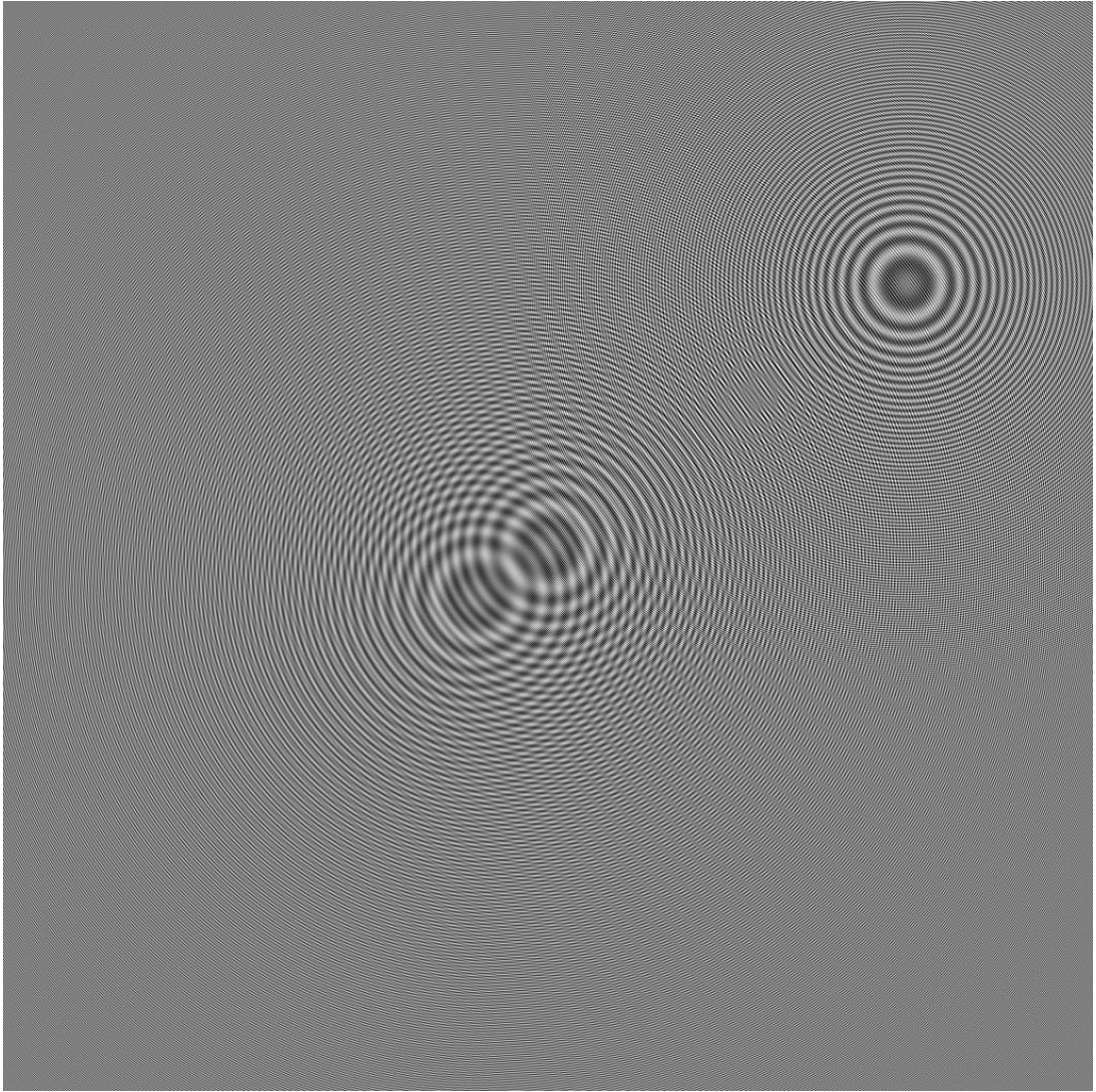


Figure 6.23: Initial pattern of the simulation 35 from Table 6.8

increases. The extreme case is shown in simulation 35 from Table 6.8 with a success rate of 8%. The corresponding initial diffraction pattern is shown in Figure 6.23.

It is clear from the Figure 6.23 one of the extra particles is so much close to reference particle both in depth and x, y plane. The other extra particle is a bit far in x, y plane but it is also close to reference object in depth. This formation yields the worst results within these simulations because the sharpness of the Wigner lines of all three particles are nearly the same due to close depth relation

and the length of these lines are also close due to x, y plane arrangement of these particles.

6.2.4 Radius of the Object and Noisy Recording

In this simulation setup, two parameters are changed at each step to examine the cross effect of these parameters to the registration. In each step, an object of the given radius is placed at the center of the initial pattern and a diffraction pattern is generated at a predetermined depth. Then the additive uniform noise is added to this pattern multiplied by the given noise weight to gain noisy recording. Finally, 25 random patterns are grabbed from this noisy pattern and automated registration is applied. Each cell in Table 6.9 shows the number of successful registration at that step. As it is observed from the Table 6.9, the success of the registration decreases as the noise level increases or as the object radius increases. All these results are expected after the simulation steps above.

	30.79	24.76	21.24	18.74	16.81	15.22	13.89	12.72
0.5	25	25	25	25	25	25	22	16
1	25	25	25	25	24	25	25	23
1.5	25	22	25	22	24	21	24	22
2	25	25	25	21	22	21	21	21
2.5	25	25	25	24	25	25	25	25
3	24	24	25	25	24	25	25	24
3.5	24	25	25	25	25	24	25	23
4	23	23	24	25	23	24	24	25
4.5	24	24	23	23	24	22	25	23
5	24	24	22	25	24	24	24	24
5.5	25	23	25	18	24	24	24	21
6	22	24	24	24	20	22	24	20
6.5	20	18	22	21	23	20	19	19
7	20	22	16	22	21	20	20	19
7.5	14	20	19	12	18	14	16	16
8	16	15	17	16	15	17	18	17
8.5	16	13	14	14	18	12	13	15
9	10	12	14	14	14	13	15	11
9.5	12	18	13	12	14	11	15	9
10	7	15	12	14	8	10	8	10

Table 6.9: The results of the simulations with respect to both radius and noise changes (Horizontal variable is PSNR in dB and vertical variable is object radius in pixels). The dynamic range of the initial pattern is $[0.9975 \ 1.0025]$. At each step, an initial high resolution diffraction pattern is generated for the given object radius. Then, a uniformly distributed random pattern is generated in the interval $[0 \ 1]$ and is multiplied by the given noise weight in the related column. Then 25 random low resolution patterns are grabbed from this high resolution noisy initial pattern as it is described earlier in text and registration method is applied.

Chapter 7

Conclusion

In this thesis, we have presented a set of techniques that can be used to register and stitch some low resolution diffraction patterns with no prior knowledge about the positions of the recording sensors, in order to form a higher resolution diffraction pattern. As explained in the introduction chapter, a higher resolution diffraction pattern has some visual advantages when it is reconstructed (i.e. larger viewing angle). To deal with a simpler problem, we stuck ourselves to a situation in which the recorded diffraction patterns (or computed ones for simulation purposes) are parallel to the object field (i.e. no tilted recordings or reconstructions). In this case, the problem reduces to the extraction of the 3D position of the recorded object from each low resolution diffraction pattern itself.

Wigner transform is well explained in the literature as a good candidate to extract position information from the diffraction patterns. All the mathematics beyond this transform is well stated in literature [18]. In this thesis, an edge detection (i.e. Canny edge detection is applied and it is well known in literature [23]) and Hough transform (it is also well known in the literature [24]) is applied to Wigner transform output in order to be able to automatically extract the position information just from the diffraction patterns themselves. Simulations

show that these estimates of the position parameters are not fine enough to stitch these diffraction patterns in order to form a high resolution diffraction pattern. Some iterative methods are used to make these rough estimates finer.

The criterion in all the iterative methods used during this thesis is the local variance of the reconstructed patterns. A moving local variance filter is used to generate the variance parameter to use as a constraint and the patterns having the maximum local variance are used all over the thesis. The effect of a wrong depth parameter is blurring (i.e. defocusing). Since diffraction pattern generation is a kind of coherent imaging, this blurring effect takes place as fringe patterns and seriously degrades the local sharpness of the pattern. The effect of bad (x, y) parameter estimation is the generation of a virtual object which also seriously degrades the local sharpness of the pattern. Hence, as it is seen from the relevant figures, local variance is a good constraint to converge to the actual 3D position parameters. It is also used in stitching process to be able to make (x, y) estimates finer and to get rid of virtual object problem.

In the stitching process a zeroth order interpolator is used to merge low resolution diffraction patterns. As seen from the results of the stitching process, this interpolator is good enough to serve our stitching purposes. As it is clear from the stitching results, the entire system works well together to form a high resolution diffraction pattern.

The enhancement in the sharpness of the reconstructed object is clearly observable in Figure 6.16 through items a to e. This enhancement is due to increasing information content in the high resolution resultant pattern and due to included higher frequency content of the fringe patterns as the number of low resolution patterns increase in the process. It is also observable from Figure 6.16 and is provable by local variance filter that the contribution of the second pattern (i.e. the difference between first and the second patterns in Figure 6.16) enhance the sharpness more than the contribution of the fifth pattern (i.e. the difference

between fourth and the fifth patterns in Figure 6.16). To simply explain this difference, let us assume that all the low resolution patterns are independent (i.e. they carry no redundant information, a non overlapping scenario can be assumed). In this case, the contribution of one pattern to another pattern after a successful registration and stitching can be thought as doubling the total information in the initial pattern (a successful registration and stitching can be accepted as merging all the information content in different patterns), however, the contribution of one pattern to other four patterns which have been successfully registered and stitched increases the information content just by 1.25 times the initially stitched pattern from first four patterns. Hence, even if all the patterns are independent, the marginal contribution of a single pattern reduces as the number of total patterns included increases.

Optimization of the simulation is conducted in the programming to reduce the required time to finish the simulations. These are very basic programming tricks (i.e. using vector notations in MATLAB) and some simple assumptions about the iterations (i.e. local variance filter is applied to non-overlapping sections of the patterns instead of applying to all possible sections on a pattern to reduce time and iterations are done in a small neighborhood assuming that the Wigner parameters are well enough to fit on these neighborhood which are explained in Section 5.3). However, it may be possible to further reduce the time of simulations if faster result responses are required.

The simulations are conducted only for the registration process under the proposed method using various scenarios. Some error limits were predetermined for the registration process under which the stitching process is guaranteed to converge to a solution (i.e. error less than 10 pixels in (x, y) plane and error less than 5% of the actual depth in z direction). The patterns which could be registered with an error smaller than these predetermined limits were accepted to be successfully registered.

After these simulations, it is understood that the proposed method is quite sensitive to object radius. As the object radius increases the success rate of the method decreases appreciably. That is because of the sensitivity of the Wigner transform to the information content. The effect of an object to the diffraction pattern with a bigger radius is the same as the effect of an object with a smaller radius but which is nearer to the recording equipment in depth. So, it is better to use this method with small objects (i.e. objects having radius smaller than 6-7 pixels). However, it may be still possible to use this method with larger objects (i.e. objects nearly having radius of 40 pixels) if human interaction is allowed because Wigner lines may still be visible when they can not be detected by edge detection.

Noise is an other important parameter. It is not possible to record a hologram without noise in real life. The performance of the proposed method is evaluated under various noise scenarios; i.e. additive Gaussian noise in recording on top of additive Gaussian noise in original pattern and additive uniformly distributed noise in recording on top of additive uniformly distributed noise in original pattern. The method is successful up a PSNR value of 8 dB for all cases.

A final realistic error source was the presence of the multiple particles in the system. The problem for this thesis was formed to have only one particle (i.e. the reference particle) in the object field. However, it is much more realistic to have more than one particle in the object field. The simulation of this case showed that proposed method still works well if the extra particles are far enough from the reference particle. Since the method does not have an extra algorithm to differentiate between different particles, it does not work well with multiple particles. As a future work, this case can be investigated to handle different particles since the Wigner transform of the different particles results in different line pairs having different slopes and positions in Wigner domain according to their 3D position.

Another important contribution to this study is to include the tilted plane recordings and reconstructions. Since in real life, it is very hard to have perfect parallelism, a tilted plane analysis is a far more realistic scenario than a parallel one. In addition to 3D position parameter estimation, some angle estimations should be included in the study in registration process and a parallelling algorithm should be provided within the automated registration process. In such case, the same stitching algorithm is then applicable to solve the superresolution problem.

APPENDIX A

General Information About Patterns

The axis properties of all the figures within this thesis is given in Figure A.1 unless otherwise is mentioned for a specific figure. Size of the figures and axis

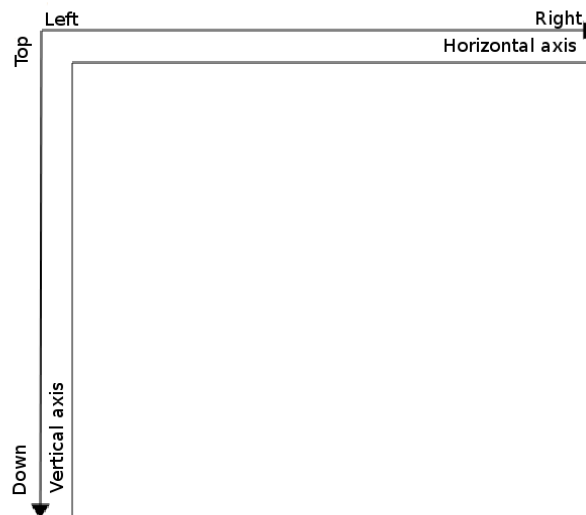


Figure A.1: Axis properties of the figures within thesis

labels are written in the caption part if necessary. For all the pattern figures, the pixel values are between $[0\ 255]$ 255 being the brightest pixel in the pattern.

APPENDIX B

Discrete Fourier Transform

During this thesis, discrete Fourier transform is used to transform space signals to frequency domain. 1D and 2D, forward and inverse transform equations are given below.

- 1D forward discrete Fourier transform of $f[n]$ of length N (i.e. $n \in [1, 2, \dots, N]$):

$$F[k] = \sum_{n=1}^N f[n] \exp\left(-j2\pi \frac{(n-1)(k-1)}{N}\right) \quad (\text{B.1})$$

where $k \in [1, 2, \dots, N]$.

- 1D inverse discrete Fourier transform of $F[k]$ of length N (i.e. $k \in [1, 2, \dots, N]$):

$$f[n] = \frac{1}{N} \sum_{k=1}^N F[k] \exp\left(j2\pi \frac{(n-1)(k-1)}{N}\right) \quad (\text{B.2})$$

where $n \in [1, 2, \dots, N]$

- 2D forward discrete Fourier transform of $f[n, m]$ of size $N \times M$ (i.e. $n \in [1, 2, \dots, N]$ and $m \in [1, 2, \dots, M]$)

$$DFT2(f) \triangleq F[k, l] = \sum_{n=1}^N \sum_{m=1}^M f[n, m] \exp\left(-j2\pi \frac{(n-1)(k-1)}{N} \frac{(m-1)(l-1)}{M}\right) \quad (\text{B.3})$$

where $k \in [1, 2, \dots, N]$ and $l \in [1, 2, \dots, M]$

- 2D inverse discrete Fourier transform of $F[k, l]$ of size $N \times M$ (i.e. $k \in [1, 2, \dots, N]$ and $l \in [1, 2, \dots, M]$)

$$IDFT2(F) \triangleq f[n, m] = \frac{1}{NM} \sum_{k=1}^N \sum_{l=1}^M F[k, l] \exp \left(j2\pi \frac{(n-1)(k-1)}{N} \frac{(m-1)(l-1)}{M} \right) \quad (\text{B.4})$$

where $n \in [1, 2, \dots, N]$ and $m \in [1, 2, \dots, M]$

APPENDIX C

Input File Generation

The object function used in this thesis is a circular opaque object in front of a transparent background which is given by the following equation for size $(N \times M)$.

$$f[y, x] = \begin{cases} 0 & , \text{ if } (y - Y_c)^2 + (x - X_c)^2 \leq r^2 \\ 1 & , \text{ otherwise} \end{cases} \quad (\text{C.1})$$

where $y \in [1, 2, \dots, N]$, $x \in [1, 2, \dots, M]$, (Y_c, X_c) is the center coordinate of the circular object and r is the radius of the object.

APPENDIX D

Propagation of Diffraction Patterns

The propagation method used in this thesis is explained by 3.30 in continuous domain. This chapter is about how to calculate these patterns in discrete domain.

Required parameters:

λ = *the wavelength of the used coherent light source*

X = *sampling period of the recording equipment*

N = *size of onr side of $(N \times N)$*

Z = *relative depth with respect to object field at which the diffraction pattern will be calculated*

Using these parameters the following ones are calculated:

$$\beta = NX/\lambda \tag{D.1}$$

$$P = Z/X \tag{D.2}$$

After that, using these parameters the transfer function associated with propagation is generated.

$$K[n, m] = \exp\left(j\frac{2\pi}{N}\sqrt{\beta^2 - \alpha^2 - \gamma^2 P}\right) \quad (\text{D.3})$$

where K is the matrix that represents propagation and $n, m \in [1, 2, \dots, N]$ and α, γ are defined as follows:

$$\alpha = \left\{ \begin{array}{ll} n - 1 & \text{for } n \in [1, N/2] \\ n - N - 1 & \text{for } n \in [N/2 + 1, N] \end{array} \right\} \quad (\text{D.4})$$

$$\gamma = \left\{ \begin{array}{ll} m - 1 & \text{for } m \in [1, N/2] \\ m - N - 1 & \text{for } m \in [N/2 + 1, N] \end{array} \right\} \quad (\text{D.5})$$

$DFT2(x)$ means the discrete Fourier transform of the 2D array x as it is explained in Appendix B Equation B.3.

$IDFT2(X)$ means the inverse discrete Fourier transform of the 2D array X as it is explained in Appendix B Equation B.4.

I is the size $N \times N$ input matrix to be propagated.

G is the size $N \times N$ result of the propagation.

Operator '.' is used to mention element by element multiplication of matrices.

$$G = IDFT2(DFT2(I) . K) \quad (\text{D.6})$$

The output matrix G is a complex valued matrix. Absolute value of this matrix is used as the diffraction pattern intensity where necessary. An element of matrix G is shown as $G(n, m)$ where $n, m \in [1, 2, \dots, N]$ and n being the variable to represent rows, m being variable to represent columns.

APPENDIX E

Discrete Wigner Transform

The method to calculate discrete Wigner transform of a 1D function is taken from [18]. Some basic steps are restated here to provide completeness.

The expression for discrete Wigner transform is given by Equation 4.2. $J[n]$ is the constant term adjusted version of the hologram function as given by Equation 4.18 of length N i.e. $n \in [1, 2, \dots, N]$. Let $W_j(n, \theta)$ be the Wigner transform of the function $J[n]$. From the properties of Wigner transform explained in Chapter 4, we can conclude that $W_j(n, \theta)$ will be nonzero only for $n \in [1, 2, \dots, N]$. It is also stated that $W_j(n, \theta)$ is a periodic function with respect to variable θ with a period of π . Hence, we need to calculate $W_j(n, \theta)$ only for the intervals $n \in [1, 2, \dots, N]$ and $\theta \in (-\pi/2, \pi/2]$.

Let us also assume that $W_j(n, \theta)$ is periodically sampled version such that $\theta = \pi p/N$ where $p \in [-(N/2) + 1, (N/2)]$ is an integer. Then the discrete transform becomes as follows:

$$W_j\left(n, \frac{\pi p}{N}\right) = 2 \sum_{k=-N/2+1}^{N/2-1} y_n(k) \exp\left(-j \frac{2\pi}{N} kp\right) \quad (\text{E.1})$$

[18] where $y_n(k) = J(n+k)J(n-k)$ for integer $k \in [-N/2 + 1, N/2 - 1]$.

Then we can define a new sequence $z_n(k)$ such that [18]

$$z_n(k) = \left\{ \begin{array}{ll} 2y_n(k) & 0 \leq k \leq N/2 - 1 \\ 0 & k = N/2 \\ 2y_n(k - N) & N/2 + 1 \leq k \leq N - 1 \end{array} \right\} \quad (\text{E.2})$$

then the Equation E.1 becomes [18]

$$W_j \left(n, \frac{\pi}{N} p \right) = 2 \sum_{k=0}^{N-1} z_n(k) \exp \left(-j \frac{2\pi}{N} kp \right) \quad (\text{E.3})$$

where $p \in [-N/2 + 1, \dots, -1, 0, 1, \dots, N/2]$.

Since we know the definition of 1D discrete Fourier transform from Equation B.1 for $p \in [1, 2, \dots, N]$, if we call $W(p) = DFT(z_n(k))$ then it is possible to write the relation below:

$$W_j \left(n, \frac{\pi}{N} p \right) = \left\{ \begin{array}{ll} W(p+1) & 0 \leq p \leq N/2 \\ W(p+1+N) & -N/2 + 1 \leq p \leq -1 \end{array} \right\} \quad (\text{E.4})$$

APPENDIX F

Canny Edge Detection

Edges in an image can be defined as the places with high intensity contrast. They are generally boundaries of the objects on a regular image. They may be used to extract object or feature information from the image. An edge detection algorithm tries to find these edges on a given image. These algorithms reduce the information content on the images, however, they highlight the desired edge information on the image under process.

Canny edge detection algorithm has four major steps[23].

F.1 Smoothing and enhancement

This step is conducted with a Gaussian filter to reduce the effect of noise in edge detection. The image of concern is convolved with a 2D Gaussian filter in order to suppress noise and enhance the quality of edges in the image. The discrete domain details of the 2D convolution are given in Algorithm 2. Let us call $I(x, y)$ the image of concern and $S(x, y)$ the noise suppressed smoothed image then the relation

$$S(x, y) = I(x, y) ** G(x, y) \tag{F.1}$$

holds where

$$G(x, y) = \sigma^2 \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (\text{F.2})$$

for $x, y \in [1, 2, \dots, N]$ and σ being a parameter for edge detection algorithm. Details of the smoothing and enhancement method is given in Algorithm 1 where $\sigma = 1$ is used in this thesis and convolution method is given in Algorithm 2. The variable *width* is a dependent variable of σ which shows the number of elements in the Gaussian filter, for the last element to be less than 0.0001. Since $\sigma = 1$ for this thesis then *width* = 4 is also constant through this thesis.

Algorithm 1 SMOOTH($I, \sigma, width$)

Require: A 2D image I

Ensure: All values of $I \in [0 \ 255]$, $\sigma > 0$, $width > 0$

- 1: {Apply 2D Gaussian filter to smooth the image}
 - 2:
 - 3: $T \leftarrow [-width \ (-width + 1) \ \dots \ (width - 1) \ width]$
 - 4: $GAU \leftarrow$ zero vector of size $2 * width + 1$
 - 5: **for** $i = 1 \rightarrow (2 * width + 1)$ **do**
 - 6: $GAU(i) \leftarrow \exp\left(\frac{-T(i)^2}{2\sigma^2}\right) / (2\pi\sigma^2)$ {1D Gaussian Filter}
 - 7: **end for**
 - 8:
 - 9: {Since Gaussian Filter is seperable we can convolve with 1D Gaussian in each direction to get the result of a 2D Gaussian}
 - 10: $I_S \leftarrow \text{CONV}(I, GAU)$ {Run the filter on rows}
 - 11: $I_S \leftarrow \text{CONV}(I_S, GAU^T)$ {Run the filter on columns}
 - 12: **return** I_S
-

Algorithm 2 CONV(I, C)

Require: Matrices I and C **Ensure:** # of dimensions of $I \geq$ # of dimensions of C

```
1:  $I_S \leftarrow$  a zero matrix with size equal to the size of  $I$ 
2:  $W \leftarrow$  a zero matrix with size equal to the size of  $C$ 
3:  $R_C \leftarrow$  (# rows of  $C - 1) / 2$ 
4:  $C_C \leftarrow$  (# columns of  $C - 1) / 2$ 
5:
6: {For all elements of  $I$ }
7: for  $i = 1 \rightarrow$  # rows of  $I$  do
8:   for  $j = 1 \rightarrow$  # columns of  $I$  do
9:     {Select the appropriate elements for convolution, for elements that are
10:    outside the boundaries choose the closest boundary element}
11:   for  $k = 0 \rightarrow R_C$  do
12:     if  $(i - k) > 0$  then
13:        $K_S \leftarrow (i - k)$ 
14:     else
15:        $K_S \leftarrow 1$ 
16:     end if
17:     if  $(i + k) \leq$  # rows of  $I$  then
18:        $K_L \leftarrow (i + k)$ 
19:     else
20:        $K_L \leftarrow$  # rows of  $I$ 
21:     end if
22:     for  $l = 0 \rightarrow C_C$  do
23:       if  $(j - l) > 0$  then
24:          $J_S \leftarrow (j - l)$ 
25:       else
26:          $J_S \leftarrow 1$ 
27:       end if
28:       if  $(j + l) \leq$  # columns of  $I$  then
29:          $J_L \leftarrow (j + l)$ 
30:       else
31:          $J_L \leftarrow$  # columns of  $I$ 
32:       end if
33:        $W(R_C - k, C_C - l) \leftarrow I(K_S, J_S)$ 
34:        $W(R_C - k, C_C + l) \leftarrow I(K_S, J_L)$ 
35:        $W(R_C + k, C_C - l) \leftarrow I(K_L, J_S)$ 
36:        $W(R_C + k, C_C + l) \leftarrow I(K_L, J_L)$ 
37:     end for
38:   end for
39:    $I_S(i, j) \leftarrow C \cdot W$  {Apply Convolution}
40: end for
41: return  $I_S$ 
```

F.2 Detection

The gradient is calculated in this step of edge detection algorithm. Gradient is a unit vector pointing at the direction of maximum change. Hence, it is legitimate to use gradient to detect edges since edges are defined to be the places with high contrast in an image. First, gradient in X and Y directions are calculated from the output of the previous step by using a 2D Gaussian filter as explained in detail in Algorithm 3, then the magnitude and the phase of the gradient is calculated as in Equations F.3 and F.4. Details of the gradient calculation is given in Algorithm 3 where $\sigma = 1$ and $width = 4$ is used as explained before in this thesis.

$$\|\nabla S(x, y)\| = \sqrt{(D_X(x, y))^2 + (D_Y(x, y))^2} \quad (\text{F.3})$$

$$\theta(x, y) = \tan^{-1} \left(\frac{D_Y(x, y)}{D_X(x, y)} \right) \quad (\text{F.4})$$

for $x, y \in [1, 2, \dots, N]$.

Algorithm 3 GRADIENT($I, \sigma, width$)

Require: A 2D image I

Ensure: I is smoothed using SMOOTH($I, \sigma, width$), $\sigma > 0$, $width > 0$

- 1: {Apply the derivative of a 2D Gaussian filter to detect edges}
 - 2: $MID \leftarrow width + 1$
 - 3: $DGAUX \leftarrow$ zero matrix of size $(2 * width + 1)$ by $(2 * width + 1)$
 - 4: **for** $i = 1 \rightarrow (2 * width + 1)$ **do**
 - 5: **for** $j = 1 \rightarrow (2 * width + 1)$ **do**
 - 6: $x \leftarrow (i - MID)$
 - 7: $y \leftarrow (j - MID)$
 - 8: $DGAUX(i, j) \leftarrow -x \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) / (\pi\sigma^2)$
 - 9: **end for**
 - 10: **end for**
 - 11:
 - 12: {2D Gaussian filter in Y direction is the transpose of the filter in X direction}
 - 13: $D_X \leftarrow \text{CONV}(I, DGAUX)$ {Run the filter on rows}
 - 14: $D_Y \leftarrow \text{CONV}(I, DGAUX^T)$ {Run the filter on columns}
 - 15: **return** D_X and D_Y
-

F.3 Edge thinning

This step is designed to reduce the ridges on the edges to detect edges as thin as possible (i.e. one pixel edges). The algorithm compares the gradient at each point with the edge normal and decides to erase that pixel if it forms a ridge on that edge. A pixel is assumed to form a ridge if its magnitude is smaller than one of the two pixels in the perpendicular direction to the gradient as explained in Algorithm 4.

Algorithm 4 THIN(M, A)

Require: A 2D gradient magnitude matrix M and a gradient angle matrix A
Ensure: M and A are calculated using GRADIENT function and equations F.3 and F.4.

```
1:  $M_S \leftarrow$  zero matrix of size equal to the size of  $I$ 
2: for  $i = 1 \rightarrow \#$  rows of  $M$  do
3:   for  $j = 1 \rightarrow \#$  columns of  $M$  do
4:     {Rounding after bounding to [0 180)}
5:      $\theta \leftarrow |A(i, j)|$  rounded to the nearest multiple of 45
6:
7:     if  $\theta == 0$  then
8:        $ii \leftarrow 1, jj \leftarrow 0$ 
9:     else if  $\theta == 45$  then
10:       $ii \leftarrow 1, jj \leftarrow 1$ 
11:    else if  $\theta == 90$  then
12:       $ii \leftarrow 0, jj \leftarrow 1$ 
13:    else if  $\theta == 135$  then
14:       $ii \leftarrow 1, jj \leftarrow -1$ 
15:    end if
16:
17:     $V_1 \leftarrow M(i + ii, j + jj), V_2 \leftarrow M(i - ii, j - jj)$ 
18:    if  $M(i, j) > V_1 \wedge M(i, j) > V_2$  then
19:       $M_S(i, j) \leftarrow M(i, j)$ 
20:    else
21:       $M_S(i, j) \leftarrow 0$ 
22:    end if
23:  end for
24: end for
25: return  $M_S$ 
```

F.4 Thresholding

The aim of this step is to remove the false edges within the detected ones. It is not easy nor sometimes possible to find a single threshold to detect all the desired edges in a pattern. Thus, Canny edge detection algorithm uses hysteresis thresholding with two thresholds. Let us call high threshold as t_H and low threshold as t_L . The algorithm begins with applying the high threshold t_H and marking the *strong edges*. Due to the high threshold, it is possible to lose some weak real edges in the original pattern that are connected to the edges. To remedy, strong edges are traced along the direction of the gradient and encountered edges are added to the edge list if their magnitude is greater than t_L . Details of this step is given in Algorithm 5.

Algorithm 5 THRESHOLD(M, A, t_L, t_H)

Require: A 2D gradient magnitude matrix M and a gradient angle matrix A

Ensure: M, A are outputs of the THIN function, $t_L > 0, t_H > t_L$

```
1:  $EDG \leftarrow$  zero matrix of size equal to  $M$  {All edges}
2:  $SEDG \leftarrow$  zero matrix of size equal to  $M$  {Strong edges}
3:
4: {First apply the high threshold}
5: for  $i = 1 \rightarrow \#$  of rows of  $M$  do
6:   for  $j = 1 \rightarrow \#$  of columns of  $M$  do
7:     if  $M(i, j) > t_H$  then
8:        $EDG(i, j) \leftarrow 1$ 
9:        $SEDG(i, j) \leftarrow 1$ 
10:    end if
11:  end for
12: end for
13:
14: {Then add weak edges to the list if they are connected to strong edges}
15: for  $i = 1 \rightarrow \#$  of rows of  $M$  do
16:   for  $j = 1 \rightarrow \#$  of columns of  $M$  do
17:     if  $SEDG(i, j) == 1$  then
18:        $(ic, jc) \leftarrow$  indices according to  $(i, j)$  and the direction given by  $A(i, j)$ 
19:
20:       {Continue until magnitude is below low  $t_L$  or a strong edge is hit}
21:       while  $M(ic, jc) > t_L \wedge SEDG(ic, jc) == 0$  do
22:          $EDG(ic, jc) \leftarrow 1$ 
23:          $(ic, jc) \leftarrow$  indices considering  $(ic, jc)$  and  $A(ic, jc)$ 
24:       end while
25:     end if
26:   end for
27: end for
28:
29: {No need to differentiate between strong and weak edges now, return all edges}
30: return  $EDG$ 
```

APPENDIX G

Hough Transform

Hough transform is a 2D-to-2D mapping of binary patterns that aims to detect lines in the input pattern [24]. It maps a non-background point from the input binary pattern to a line in the output pattern by using the parametric line equation (Equation G.1).

$$\rho = x \cos \theta + y \sin \theta \quad (\text{G.1})$$

The input pattern is assumed to be in (x, y) coordinates and the output is generated in (θ, ρ) coordinates. ρ is the magnitude of the vector perpendicular to the line and originating from the origin as shown in Figure G.1. θ is the angle of this vector in degrees clockwise from the positive x-axis.

The output of the Hough transform is a 2D accumulator matrix (HT) whose rows correspond to ρ values and columns correspond to θ values. Initially all the values in HT are zero. For every non-background pixel in input pattern ρ is calculated for each θ in the desired interval with the Equation G.1. Then ρ is rounded to select a suitable row from HT and then the corresponding cell in HT is incremented. At the end of this method, any value from the accumulator matrix $N = HT(t, r)$ means that there are N pixels in the original pattern lying

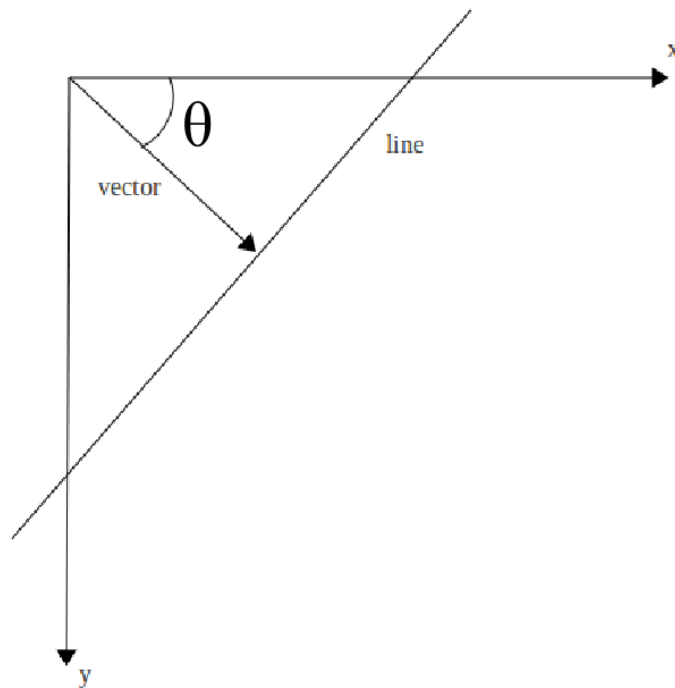


Figure G.1: This figure explains the relation between (x, y) coordinates and (θ, ρ) coordinates in Hough transform.

on the line specified by $\theta(t)$ and $\rho(r)$. Hence, the maximum values in the HT represents the possible lines in the original pattern.

It is possible to define a desired θ range in the algorithm. It is also possible to conduct this algorithm for any desired θ and ρ resolutions, however, it is important to note that higher resolutions will yield larger accumulation matrices which can become difficult to process. It is legitimate to search for a predetermined number of peaks in the accumulator matrix if the potential number of lines to be detected in the original pattern is known. It is also legitimate to suppress a close neighborhood of a peak in the accumulator matrix due to numerical errors. As a last remark on Hough transform, it is also practical to search for peaks which are larger than a given threshold in order to suppress false line detections.

Required inputs to generate discrete Hough transform:

- $I =$ input binary pattern of size $N \times N$

- ρ resolution = the step to increment ρ values. Assumed to be 0.5 during this thesis.
- θ resolution = the step to increment θ values. Assumed to be 0.1 degrees during this thesis.
- NH = tells the neighbourhood of a peak to be suppressed after detection. Assumed to be 5 pixels during this thesis.
- NOL = tells how many peaks to search in the accumulator matrix (i.e. how many different lines to detect). Assumed to be 8 during this thesis.
- PH = the threshold for which a peak is accepted if it is greater than this threshold. Assumed to be 0.8 times the maximum value in HT.

Parameters:

- $\rho \in [0, 0.5, 1, \dots, N - 1, N - 0.5, N]$ length = $2N-1$
- $\theta \in [-90, -89.9, -89.8, \dots, -0.1, 0, 0.1, \dots, 88.8, 88.9, 89]$ length = 1800
- $HT = (2N - 1 \times 1800)$ matrix of all entries are initially zero.

The algorithm of discrete Hough transform:

- For each pixel in I satisfying $I(y, x) = 1$, calculate ρ for each possible θ by Equation G.1.
 - round ρ to fit an integer in $[1, 2N-1]$.
 - increment the cell in HT by 1 with row number equal to rounded ρ and corresponding column number for related θ .
- Find the position and value of the global maximum of HT (i.e. $m = \max(HT(\rho, \theta))$)

- Record corresponding θ and ρ values for this maxima
- Define $PH = 0.8m$
- Delete the data in HT in the neighbourhood of this maxima (i.e. a $NH \times NH$ matrix segment maxima being in the middle)
- Go ahead to find new global maximum of HT , record the corresponding θ and ρ values if this maxima is greater than PH , delete the neighbourhood. Repeat this step until finding NOL peaks or until the newly found peak is not greater than PH .
- Recorded θ and ρ values represent the detected lines by Hough transform.

Bibliography

- [1] T. Okoshi, *Three Dimensional Techniques*. New York Academic Press, 1976.
- [2] M. Lucente, “Optimization of hologram computation for real-time display,” in *SPIE Proceedings 1667 Practical Holography (SPIE, Bellingham, WA, Feb 1992)*, pp. 32 – 43, Feb. 1992.
- [3] M. McKenna and D. Zeltzer, “Three dimensional visual display systems for virtual environments,” in *Presence: Teleoperators and Virtual Environments*, vol. 1, pp. 421–458, 1992.
- [4] M. Lucente and T. A. Galyean, “Rendering interactive holographic images,” in *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, (New York, NY, USA), pp. 387–394, ACM, 1995.
- [5] O. Nishikawa, T. Okada, H. Yoshikawa, K. Sato, and T. Honda, “High-speed holographic-stereogram calculation method using 2D FFT,” in *Diffraction and Holographic Device Technologies and Applications IV*, vol. 3010, pp. 49–57, SPIE, 1997.
- [6] Y. Kajiki, H. Yoshikawa, and T. Honda, “Auto-stereoscopic 3D video display using multiple light-beams with scanning,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, pp. 254 – 260, March 2000.
- [7] J. H. Massig, “Digital off-axis holography with a synthetic aperture,” *Opt. Lett.*, vol. 27, no. 24, pp. 2179–2181, 2002.

- [8] R. Binet, J. Colineau, and J.-C. Leheureau, “Short-range synthetic aperture imaging at 633 nm by digital holography,” *Appl. Opt.*, vol. 41, no. 23, pp. 4775–4782, 2002.
- [9] B. M. Hennelly, T. J. Naughton, and J. McDonald, “Digital holographic superresolution by rotating the object wavefield,” in *Adaptive Optics: Analysis and Methods/Computational Optical Sensing and Imaging/Information Photonics/Signal Recovery and Synthesis Topical Meetings on CD-ROM*, p. DTuD5, Optical Society of America, 2007.
- [10] J. Di, J. Zhao, H. Jiang, P. Zhang, Q. Fan, and W. Sun, “High resolution digital holographic microscopy with a wide field of view based on a synthetic aperture technique and use of linear ccd scanning,” *Appl. Opt.*, vol. 47, no. 30, pp. 5654–5659, 2008.
- [11] T. Kreis and K. Schlüter, “Resolution enhancement by aperture synthesis in digital holography,” *Optical Engineering*, vol. 46, no. 5, p. 055803, 2007.
- [12] Ryle, M., Hewish, A., *Mon. Not. R. Astron. Soc.* 120, 220, 1960.
- [13] A. R. Thompson, J. M. Moran, and G. W. Swenson, *Interferometry and synthesis in radio astronomy*. John Wiley & Sons, New York, 1986.
- [14] J. P. Fitch, *Synthetic Aperture Radar*. New York:Springer-Verlag, 1988.
- [15] T. Sato, M. Ueda, and S. Fukuda, “Synthetic aperture sonar,” *The Journal of the Acoustical Society of America*, vol. 54, no. 3, pp. 799–802, 1973.
- [16] S. E. Robinson and J. Vrba, “Functional neuroimaging by synthetic aperture magnetometry (SAM),” in *Recent Advances in Biomagnetism*, pp. 302–305, Tohoku Univ. Press, Sendai, 1999.
- [17] E. Wigner, “On the quantum correction for thermodynamic equilibrium,” *Phys. Rev.*, vol. 40, pp. 749–759, Jun 1932.

- [18] L. Onural and M. T. Özgen, “Extraction of three-dimensional object-location information directly from in-line holograms using Wigner analysis,” *J. Opt. Soc. Am. A*, vol. 9, no. 2, pp. 252–260, 1992.
- [19] T. A. C. M. Claasen and W. F. G. Mecklenbrauker, *The Wigner distribution—a tool for time-frequency signal analysis.*, ch. Part 1: Continuous-Time Signals, pp. 217–250. Philips J. Res. 35, 1980.
- [20] T. A. C. M. Claasen and W. F. G. Mecklenbrauker, *The Wigner distribution—a tool for time-frequency signal analysis.*, ch. Part 2: Discrete-Time Signals, pp. 276–300. Philips J. Res. 35, 1980.
- [21] T. A. C. M. Claasen and W. F. G. Mecklenbrauker, “The aliasing problem in discrete-time Wigner distributions,” in *IEEE Trans. Acoust. Speech Signal Process*, vol. ASSP-31, pp. 1067–1072, 1983.
- [22] B. Boashash, “Note on the use of the Wigner distribution for time-frequency analysis,” in *IEEE Trans. Acoust. Speech Signal Process*, vol. 36, pp. 1518–1521, 1988.
- [23] J. Canny, “A computational approach to edge detection,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, pp. 679–698, Nov. 1986.
- [24] R. O. Duda and P. E. Hart, “Use of the Hough transformation to detect lines and curves in pictures,” *Commun. ACM*, vol. 15, no. 1, pp. 11–15.