Novel Computational Technique for Determining Depth Using the Bees Algorithm and Blind Image Deconvolution

A thesis submitted to Cardiff University in the candidature for the degree of

Doctor of Philosophy

By

Baris Yuce, B.Sc., M.Sc.

Manufacturing Engineering Centre

School of Engineering

Cardiff University

United Kingdom

September 2012

ABSTRACT

In the past decade the Scanning Electron Microscope (SEM) has taken on a significant role in the micro-nano imaging field. A number of researchers have been developing computational techniques for determining depth from SEM images. Depth from Automatic Focusing (DFAF) is one of the most popular depth computation techniques used for SEM. However, images captured with SEM may be distorted and suffer from problems of misalignment due to internal and external factors such as interaction between electron beam and surface of sample, lens aberrations, environmental noise and artefacts on the sample.

Distortion and misalignment cause computational errors in the depth determination process. Image correction is required to reduce those errors. In this study the proposed image correction procedure is based on Phase Correlation and Log-Polar Transformation (PCLPT), which has been extensively used as a pre-processing stage for many image processing operations.

The computation process of PCLPT covers the pixel level interpolation process but it cannot deal with sub-pixel level interpolation errors. Hence, an image filtering stage is necessary to reduce the error. This enhanced PCLPT was also utilised as a pre-processing step for DFAF which is the first contribution of this research.

Although DFAF is a simple technique, it was found that the computation involved becomes more complex with image correction. Thus, the priority to develop a less complicated and more robust depth computation technique for SEM is needed. This study proposes an optimised Blind Image Deconvolution (BID) technique using the Bees Algorithm for determining depth.

The Bees Algorithm (BA) is a swarm-based optimisation technique which mimics the foraging behaviour of honey bees. The algorithm combines exploitative neighbourhood search with explorative global search to enable effective location of the globally optimal solution to a problem. The BA has been applied to several optimisation problems including mechanical design, job shop scheduling and robot path planning. Due to its promise as an effective global optimisation tool, the BA has been chosen for this work.

The second contribution of the research consists of two improvements which have been implemented to enhance the BA. The first improvement focuses on an adaptive approach to neighbourhood size changes. The second consists of two main steps. The first step is to define a measurement technique to determine the direction along which promising solutions can be found. This is based on the steepness angle mimicking the direction along which a scout bee performs its figure-of-eight waggle dance during the recruitment of forager bees. The second step is to develop a hybrid algorithm combining BA and a Hill Climbing Algorithm (HCA) based on the threshold value of the steepness angle.

The final contribution of this study is to develop a novel technique based on the BA for optimising the blurriness parameter with BID for determining depth.

The techniques proposed in this study have enabled depth information in SEM images to be determined with 68.23 % average accuracy.

ACKNOWLEDGEMENTS

I would like to thank my supervisor Professor Duc Truong Pham for his excellent supervision, help and continuous encouragement. My thanks go to him for accepting me to be one of his students in the Manufacturing Engineering Centre (MEC) at Cardiff University. Also, through the recommendation of Professor Duc Truong Pham, I have achieved the Dorothy Hodgkin Postgraduate Award.

In addition, without the assistance of Dr. Michael Packianather and Dr. Emmanuel Brousseau this thesis might not have been possible to produce.

Special thanks to Professor Cemalettin Kubat, Professor Ercan Oztemel and Assistant Professor Ozer Uygun who helped me to meet Professor D.T. Pham.

Also, special thanks to Dr. Siti Azfanizam Ahmad and Mr. Mario Javier Gonzalez Romo, Mr. Azar Imanguliyev, Miss Janyarat Phrueksanant, Mr Fuad Omar.

I also thank to Cardiff University for the Dorothy Hodgkin Postgraduate Awards for sponsoring me during my study.

Last but not least, I thank my lovely family for always supporting me; mentally, emotionally and physically during this four years journey.

Specially dedicated to my family.

With millions of thanks and love.

Contents

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

SEM	Scanning Electron Microscope
DFAF	Depth From Automatic Focusing
AF	Automatic Focusing
MTF	Modulate Transfer Function
PC	Phase Correlation
SHC	Stochastic Hill Climbing
SA	Simulated Annealing
TS	Tabu Search
GA	Genetic Algorithm
GP	Genetic Programming
EP	Evolutionary Programming
PSO	Particle Swarm Optimisation
ACO	Ant Colony Optimisation
SBOT	Swarm Based Optimisation Technique
BID	Blind Image Deconvolution
PSF	Point Spread Function
BCO	Bee Colony Optimisation
ICA	Independent Component Analysis
PCLPT	Phase Correlation and Log Polar Transformation
SAMD	Sum of The Absolute Matching Difference
BA	The Bees Algorithm
ANSSA	Adaptive Neighbourhood Size Site Abandoned
SACHCA	Slope Angle Computation and Hill Climbing Algorithm
HCA	Hill Climbing Algorithm
RMS	Root Mean Square

LIST OF SYMBOLS

Chapter 2.	
$g(x,y)$	The intensity of the observed image on point (x,y)
$h(x,y)$	The PSF of the System
$f(x,y)$	The intensity of the original image on point (x,y)
A_0	The aperture size of the SEM
d_0	The distance from focused image plane to the objective lens of the SEM
d_1	The distance between surface of the specimen and the objective lens of the SEM
A_1	The SEM beam diameter on the specimen surface
δ	The distance between the specimen and the focused image plane
$f(X)$	The fitness function of the optimisation problem
$g(X)$	Inequality constraints in the optimisation problem
$h(X)$	Equality constraints in the optimisation problem
p	Probability distribution function
T	Temperature
S	Step size
i	The iteration number (i.th iteration)
ΔE	The Energy Change
k_B	The Boltzaman Constant
$p_{ij}^{m}(t)$	The transition probability from node i to node j
τ_{ij}	The posterior effectiveness of the move from node i to node j

α	The parameter to control the influence of τ_{ij}
η_{ij}	The prior effectiveness of the move from node i to node j
δ	The distance between the specimen and the focused image plane
β	The parameter to control the influence of
N_i^m	The set of feasible nodes for ant m when located on node i
ρ	The reduction rate of the pheromone level
n_A	The number of ants
$\Delta\tau_{ij}^m(t)$	The amount of pheromone deposited by ant m from node j to node j at time step t
X^*	The Observed Signal
$H(Y)$	The entropy of the discrete variable
Y	The discrete variable
$H(y)$	The entropy of the continuous variable
y	The continuous variable
$J(y)$	The negentropy of the y
$E(y^3)$	The third moment function of the y
$I(y_1, y_2, ... y_n)$	The mutual information between $y_1, y_2, ..., y_n$
Chapter 3.	
$I(x,y)$	The image intensity value on point (x, y)
$F(u,v)$	The Fourier transformation of the image $I(x,y)$
$F^*(u,v)$	The complex conjugate of the $F(u,v)$
$(\Delta x, \Delta y)$	The amount of the translation on (x,y) directions
ρ	The radial distance of the position (x,y) in polar domain
θ	The position angle of the (x,y) according to the

	reference plain in polar domain
β	The rotation angle according to the original image
γ	The scaling factor for the given image
$M(u,v)$	The magnitude of the (u,v) in Fourier domain
Chapter 4.	
ngh	The neighbourhood size
sc	The shrinking constant
$aband\_site$	The abandoned site
$keep\_point$	The number of repetition for each site
rep_nenh	The number of the repetition for the enhancement strategy
rep_naban	The number of the repetition for the abandoned site strategy
$slope\_angle$	The slope angle of the each site
$\nabla F(X)$	The gradient of the position X
h	The incremental size for the Hill Climbing Algorithm
$angle\_limit$	The angle limit for starting the Hill Climbing Algorithm
Chapter 5.	
$median$	The median value of the image histogram
$standard\_deviation$	The standard deviation of the image histogram
α	The scale factor
$Threshold$	The threshold value for the whitening process
$Maximum$	The maximum number in the image histogram
$\hat{f}(x,y)$	The predicted image
$\hat{h}(x,y)$	The predicted PSF
R	The blurriness radius
σ	The blurriness parameter

Chapter 1

INTRODUCTION

1.1. Background

Over the years, the development of depth computation techniques has been one of the main tasks in the machine vision area. Depth computation is required in several applications such as surface imaging, robotics, pattern recognition, manipulation, and nano-manufacturing.

There are different types of classification for depth computation techniques. The classification can be defined according to the image focus level, such as depth from automatic focusing (DFAF), depth from defocusing (DFD) and depth from automatic defocusing (DFAD). The DFAF technique utilises the sharpness function to find the focused image position with respect to the max / min value of the sharpness function. The DFD technique is based on the image blurriness level. The technique utilises the blurriness parameter to find depth information. The DFAD technique is based on reducing the blurriness of the image by varying the optical system parameters, and then utilising DFAF rules to determine depth.

Applications of depth computation techniques have been implemented on several types of imaging systems such as: white-light microscope, interferometer, Transmission Electron Microscope (TEM), and Scanning Electron Microscope (SEM). Firstly, this study focuses on the DFAF technique on SEM, which is one of the key systems for micro-nano imaging. Even though SEM is essential for the micro-nano area, a SEM image may be affected by distortion, translation and rotation during image acquisition. This is due to internal and external noise such as interaction between the specimen surface and electron beam, artefact on the surface, and environmental vibrations (Snella, 2010). Therefore, the captured image may need to be adjusted by an image correction process.

Various types of image registration models have been developed to correct the orientation of the image according to a reference image. Image registration models can be classified into different groups. One group is based on feature matching such as the feature-based and area-based image registration models (Zitovaa and Flusser, 2003). Feature-based image registration models cannot perform without a geometrical descriptor. However, area-based image registration techniques utilise statistical criteria in the Spatial / Fourier domain.

Fourier domain-based image registration models have a computational advantage compared to spatial domain-based models. Phase correlation is one the most widely used Fourier domain-based image registration techniques. Although phase correlation can find image translation it cannot work without a log-polar transformation to solve translation, rotation and scaling problems. The computational process of phase correlation and log-polar transformation covers the pixel level interpolation process but there is an error in the sub-pixel level and so it requires pre-processing. One of the pre-processing stages can be image filtering. This is an essential tool in the image processing area as the filtering process can change the image sharpness level. Different types of image filters can be utilised in a pre-processing step.

The blurriness parameter is utilised in the DFD technique to find depth. The blurriness parameter of the image cannot be determined easily, unless it has been found by deconvolution using a known sharp original image. In practice, there is no information about the sharp image and the Point Spread Function (PSF) which is convolved with the former to produce the blurred actual image. Therefore, a blind deconvolution process has been utilised to find both the original image and the PSF. The blind deconvolution process is an optimisation problem where a sharp image is found at the optimum point of some objective function.

There are various types of optimisation techniques. They can be classified based on the variable types such as deterministic and stochastic optimisation techniques. Deterministic optimisation techniques can solve problems in polynomial time. However, most of the optimisation problems cannot be solved in polynomial time and stochastic techniques are required to solve them. There are several stochastic optimisation techniques such as Stochastic Hill Climbing, Random Optimisation, Simulated Annealing, Tabu Search, Genetic Algorithm, Evolutionary Programming, Particle Swarm Optimisation, Ant Colony Optimisation, and the Bees Algorithm. Stochastic Hill Climbing is a direct search optimisation algorithm. Random Optimisation is a random search technique which is one of the simplest stochastic optimisation techniques. Simulated Annealing is inspired by the annealing process of metals. Tabu Search is inspired by the human memory process. The Genetic Algorithm technique mimics natural evolution. Evolutionary Programming is also inspired by the same natural process. Particle Swarm Optimisation models the way a flock of birds or a school of fish moves. The Ant Colony technique is based on how ants behave. Finally, the Bees Algorithm is inspired by the foraging routine of honey bees.

The Bees Algorithm is a stochastic and population-based optimisation algorithm. The algorithm has both local and global search ability, which makes the Bees Algorithm effective at finding the true optimum solution to a problem. The Bees Algorithm has been successfully tested on various types of problems. The algorithm has been improved with different types of strategies such as neighbourhood size change, site abandonment strategy and population size change. These improvements were focused on the neighbourhood search site. Other improvements have been carried out with hybrid approaches involving combining the Bees Algorithm with other techniques such as Particle Swarm Optimisation and Ant Colony Optimisation.

1.2. Motivation

The automatic focusing technique has been employed on SEM (Nicolls, 1995 and Batten 2000). (Nicolls, 1995) developed an automatic focusing-based depth computation process for SEM. The process is based on utilising the optical transfer function of two images. Depth computation with automatic focusing requires pre-processing to correct the captured image (Snella, 2010). Phase correlation and log-polar transformation have been utilised for image correction (Zitova and Flusser, 2003). However, phase correlation and log-polar transformation can be sensitive to errors due to the interpolation process at the sub-pixel level. Thus the first motivation of this work was to improve pre-processing for the phase correlation and log-polar transformation technique with image filtering. Then, effort was focused on relating the sharpness function value to depth. This work was carried out by utilising the DFAF technique on a SEM.

The pre-processed DFAF technique has been successfully applied in some particular cases but requires much computational effort. Therefore, it makes the process very slow. To avoid the heavy computational demands an alternative depth computation technique was developed using blind deconvolution which is an optimisation problem. There are several blind deconvolution techniques developed based on the Genetic Algorithm, Tabu Search, and Simulated Annealing. A novel Bees Algorithm-based blind deconvolution process was developed in this study to predict the blurriness parameter. As mentioned before, the Bees Algorithm has been applied to various types of optimisation problems and the algorithm has been enhanced with several approaches such as improving the neighbourhood search and combining it with other optimisation algorithms. In this work, the Bees Algorithm was hybridised with Hill Climbing and its neighbourhood search parameters were allowed to change adaptively. These modifications were aimed at producing a more efficient optimisation algorithm.

1.3. Aim and objectives

The overall aim of this research was to develop a robust depth computation technique for SEM. The first proposed depth computation technique was based on the pre-processed DFAF technique. The second depth computation technique was developed from blind deconvolution optimised using the Bees Algorithm. Initially, the Bees Algorithm was improved in two ways. First, the neighbourhood size was allowed to change adaptively and site abandonment was adopted. Second, hybridisation with Hill Climbing was implemented.

The objectives of this work were:

1. To develop a pre-processed phase correlation and log-polar transformation technique with image filtering.
2. To develop a method of correcting SEM images by employing the proposed pre-processed phased correlation and log-polar transformation technique.
3. To develop a method of computing the sharpness of the image for different distances, and then relating the distance and sharpness of corrected images.

4. To enhance the Bees Algorithm with adaptive neighbourhood size change and site abandonment strategy.
5. To improve the Bees Algorithm by combining it with the Hill Climbing Algorithm.
6. To apply the enhanced Bees Algorithm to the blind deconvolution technique.
7. To compute the blurriness parameter with the optimised blind deconvolution technique.
8. To relate depth and the blurriness parameter.

1.4. Research methods

In carrying out this research, the following methodologies were adopted:

1. Surveying previous work related to depth computation, image registration, optimisation and deconvolution techniques.
2. Implementing the proposed algorithms in MATLAB
3. Testing the pre-processed phase correlation and log-polar transformation technique on benchmark images and SEM images.
4. Applying the pre-processed phase correlation and log-polar transformation technique on different depth levels of SEM images and correcting them.
5. Applying the DFAF technique on corrected SEM images and relating sharpness values with distances.

6. Improving the Bees Algorithm and then testing the improved algorithm on benchmark functions and an optimisation problem.
7. Applying the improved Bees Algorithm to blind deconvolution.
8. Testing the optimised blind deconvolution techniques on benchmark images and SEM images.
9. Applying the optimised blind deconvolution algorithm on different SEM images obtained at different depths.
10. Relating the predicted blurriness parameter and depth.

1.5. Outline of the thesis

The remainder of the thesis is organised as follows.

Chapter 2 reviews the DFAF technique. The image registration technique is presented. Then optimisation techniques are described and the Bees Algorithm is highlighted. Finally, blind deconvolution is discussed.

Chapter 3 introduces the phase correlation and log-polar transformation technique. Image filtering is introduced as a pre-processing step. The chapter describes the testing of the pre-processed phase correlation and log-polar transformation technique on both benchmark images and SEM images of a copper sample. Then it gives the results of applying the proposed technique to SEM images at different distances. Finally, the chapter presents the application of the

DFAF technique to corrected images and assignment of specific depths to corresponding sharpness values.

Chapter 4 presents two improvements to the Bees Algorithm. The first improvement is based on using adaptive neighbourhoods and site abandonment. The second improvement is based on combining the Bees Algorithm and Hill Climbing. The improved Bees Algorithm has been tested on benchmark functions and a given optimisation problem. The test results are presented in the chapter.

Chapter 5 introduces the Bees Algorithm-based optimised blind deconvolution technique and presents its application to benchmark images and SEM images of a copper sample at different distances from the SEM. The chapter gives the relationship between image depth and the values of the blurriness parameter obtained.

Chapter 6 lists the contributions of this research, summarises the conclusions reached and provides suggestions for further research.

Chapter 2

LITERATURE REVIEW

2.1. Preliminaries

This chapter overviews the implementation of Depth from Automatic Focusing (DFAF) on a SEM, and image registration, deconvolution and stochastic based optimisation techniques. Non-Gaussianity-based blind deconvolution techniques are reviewed in detail.

2.2. Depth from Automatic Focusing on a SEM

DFAF is one of the most widely used technique for depth determination in the area of machine vision (Tenenbaum, 1970; Krotkov, 1987; Xiong and Shafer, 1993; Tyan, 1997; Subbarao and Tyan, 1998). To determine depth DFAF finds the sharpest image from different depth levels using sharpness measurement functions. These are the functions used with the Automatic Focusing (AF) technique which has been an important development of the SEM (Yousefi, et al., 2011).

Originally proposed for use on the SEM by Nicolls (1995) the AF technique uses the Modulation Transfer Function (MTF) in the Fourier domain. This was developed by Batten (2000) as an iterative-based sharpness search algorithm to find the position of the sharpest image from a set of captured images.

The image captured from an out of focus plane is blurred, where the degree of blurring is determined by the convolution operation, given in Equation 2.1 (Pentland, 1987; Subbarao and Surya, 1994; Aslantas, 1997):

$$g(x,y) = \iint f(x,y)h(x-m,y-n)dmdn \tag{2.1}$$

where g is the observed image, h is the Point Spread Function (PSF) of the system and f is the focused image. Equation 2.1 is the basis of the computational process used for cleaning blurred images (Dobes et al., 2010).

Image formation in a SEM is shown in Figure 2.1, where the focused image occurs on the crossover point of the beams. If the position of specimen is far from the beam crossover, the captured image will be blurred. In Figure 2.1, A_0 denotes the aperture size of the SEM, d_0 denotes the distance from the focused image plane to the objective lens, d_1 denotes the distance from the surface of the specimen to the objective lens, A_1 denotes the beam diameter size on the specimen surface and δ denotes the distance from the specimen to the focused image plane.

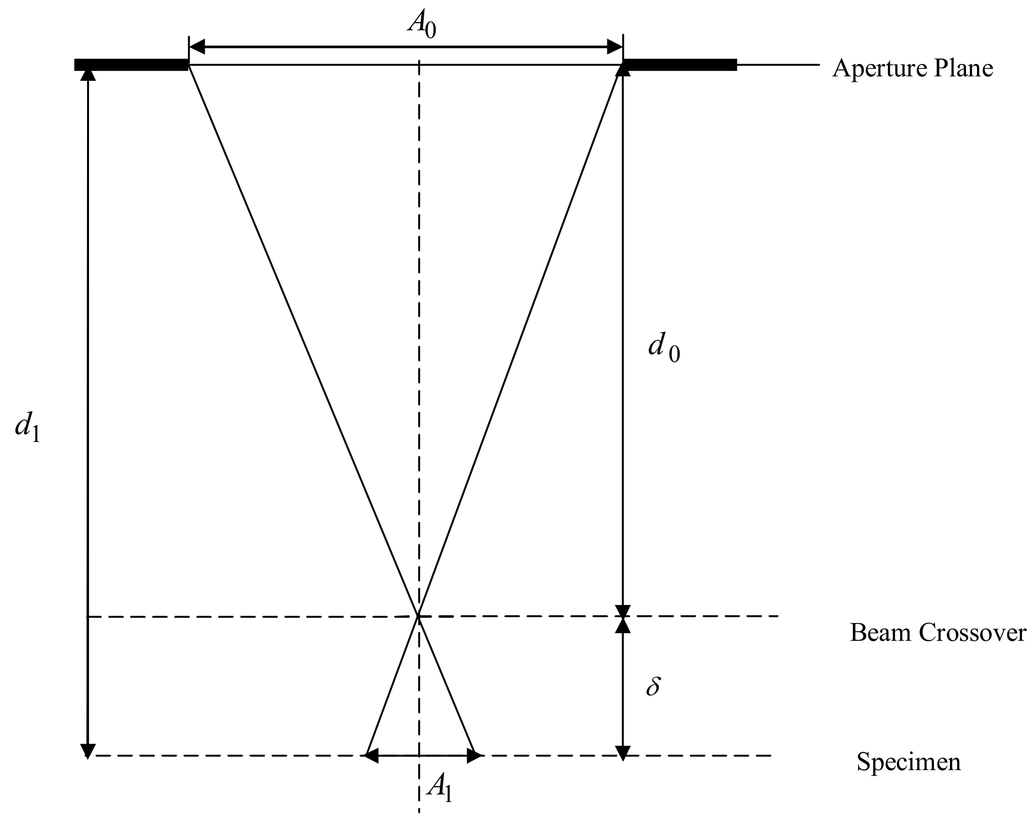


Figure 2.1 Image formation with SEM.

The sharpness function will be a maximum or minimum (depending on the given sharpness function) on the focused image plane. The sharpness functions have been classified into five groups by Aslantas (1997) as shown in Figure 2.2. The functions considered for use in this research are given in Table 2.1. A full list of sharpness measurement functions is given in Appendix A.

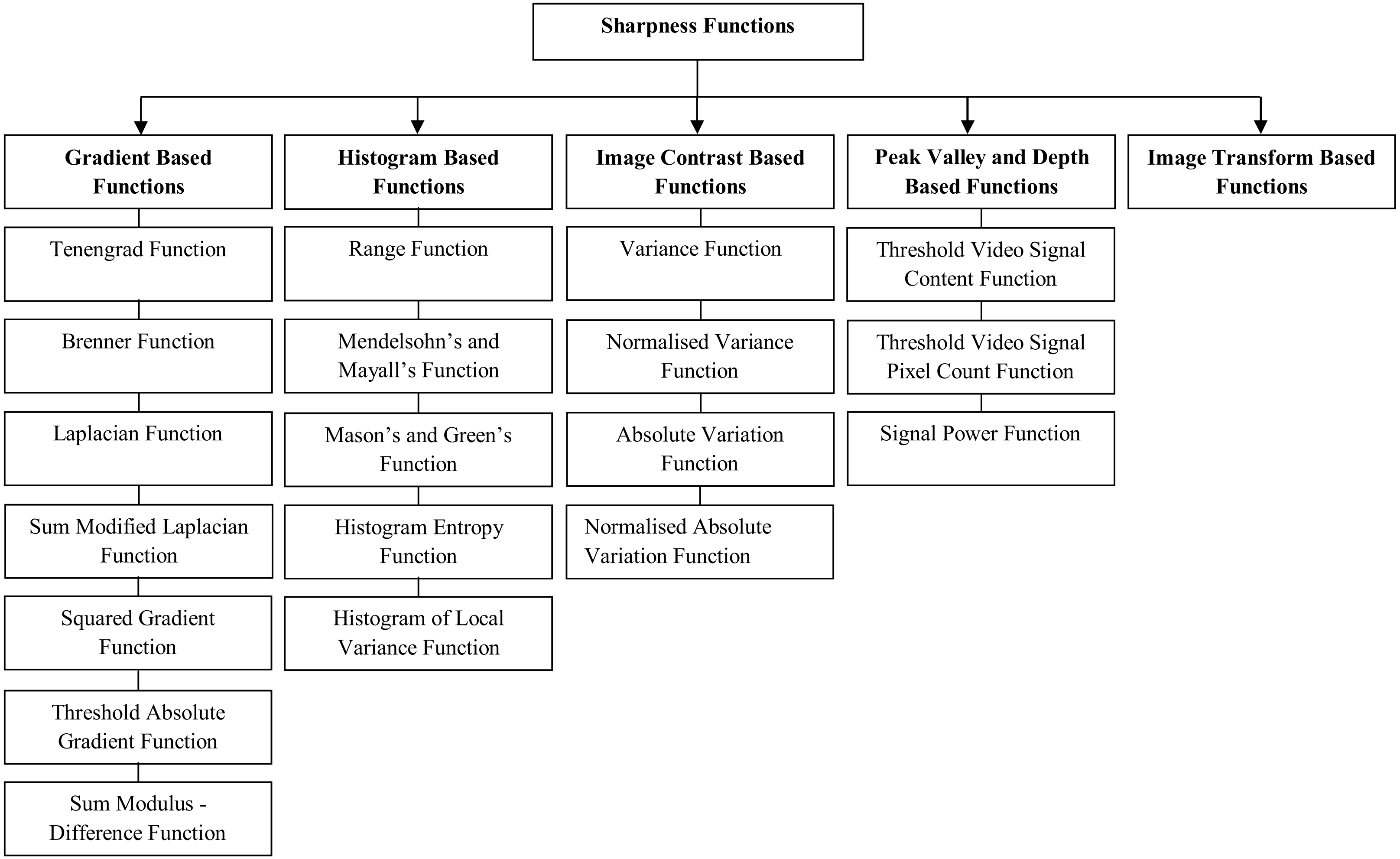


Figure 2.2 Classification of sharpness functions for the AF technique.

Table 2.1 Sharpness functions for the AF technique used in this project.

No	Function Name	Explanation	Function	Parameter / Operator
1	Tenengrad Function	The function computes the gradient of the image, the sharpest image has maximum value (Schlag, et al., 1983 and Aslantas 1997)	$F = max\left\{ \sum_{x=1}^{N} \sum_{y=1}^{N} I(x,y)^2 \right\}$	$I(x,y) = \sqrt{I_x(x,y)^2 + I_y(x,y)^2}$; where $I(x,y)^2 > T$ $I_x(x,y) = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, I_y(x,y) = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$ (Sobel Operator is available to compute the derivations).
2	Brenner Function	Brenner function computes the sum of squared differences between pixels and pixels two units away, the sharpest image has maximum value (Brenner, et. al, 1984 and Aslantas 1997).	$F = max\left\{ \sum_{x=1}^{N} \sum_{y=1}^{N} \left(I(x,y+2) - I(x,y) \right)^2 \right\}$	where $I(x,y)^2 > T$
3	Squared Gradient Function	The function computes the sum of the squared differences of each pixel and neighbouring pixels one unit away, the sharpest image has maximum value (Santos, et al., 1997).	$F = max\left\{ \sum_{x=1}^{N} \sum_{y=1}^{N} \left\| I(x,y+1) - I(x,y) \right\|^2 \right\}$	where $\left\| I(x,y+1) - I(x,y) \right\| \geq T$
4	Range Function	The function is based on the search for the maximum value of the difference between the maximum grey level and minimum grey level of the image (Firestone et al., 1991)	$F = max\{ I_{max} - I_{min} \}$	where I_{max} and I_{min} are the maximum and minimum grey level, respectively.

Table 2.1 Sharpness functions for the AF technique used in this project (cont.).

5	Histogram Entropy Function	The function minimises the entropy of the histogram diagram (Aslantas, 1997; Santos, et al., 1997; Aslantas and Kurban, 2009).	$F = min\left\{ -\sum_{I} P(I)\,ln(P(I)) \right\}$	where $P(I)$ is the probability of the grey level of I in histogram diagram. and $P(I) \neq 0$
6	Laplacian Function	Computes the sum squared Laplacian of the image and sharpest image has maximum value (Muller et al., 1974; Krotkov, 1987).	$F =$ $max\left\{ \sum_{x=1}^{N} \sum_{x=1}^{N} \left(I(x,y+1) - 2I(x,y) - I(x,y-1) \right)^2 \right\}$	where $I(x,y)^2 > T$ The following operators are available. $F = \{ \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}, \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}, \begin{bmatrix} -1 & -4 & -1 \\ -4 & 20 & -4 \\ -1 & -4 & -1 \end{bmatrix} \}$
7	Variance Function	The variance function computes variance of images and the sharpest image has the maximum value (Aslantas, 1997; Yousefi, et al., 2011).	$F = max\left\{ \frac{1}{N^2} \sum_{x=1}^{N} \sum_{y=1}^{N} \left[I(x,y) - \bar{I} \right]^2 \right\}$	

2.2.1. Gradient-Based Functions

Gradient-based optimisation functions are widely used for image enhancement. The aim is to maximise the gradient image brightness in the region of interest and is based on the physical observation that image focus quality affects edge characteristics (Aslantas and Kurban, 2009; Rudnaya et al., 2011). The best focused image has the sharpest edges and maximum value for the gradient.

2.2.2. Histogram-Based Functions

Histogram-based functions are a type of sharpness measurement function. This group of functions deal with image pixel values. The image focus level can be measured from the histogram of pixel values. For example, the focused image has a higher range of grey level but a blurred image contains a lower range of grey level (Rudnaya and Ochshorn, 2011).

2.2.3. Image Contrast-Based Functions

Image contrast-based sharpness functions measure the variation of image pixel value. A sharp image has more variation than a blurred one (Aslantas, 1997).

2.2.4. Peak Height and Valley Depth-Based Functions

These types of sharpness measurement functions compute the amplitudes of peaks or valleys in the image (Aslantas, 1997). The maximisation of these parameters gives the sharpest image because the best image has the sharpest edges and thus the pixel grey level has its widest range.

2.2.5. Image Transform-Based Functions

Image transformation-based functions transform the image from the spatial domain to the desired domain. Rudnaya, et al., (2010) have used image transformation based on the Fourier transform and shown that because well-focused images contain sharper edges than blurred images, they have higher frequency components. Thus, the Fourier transform can be employed to find the sharpest image by searching for the transform with maximum value.

Other types of transformations are orthogonal transformations such as Discrete Cosine and Walsh-Hadamard transformations and can be used to measure image sharpness (Aslantas, 1997).

2.3. Image Registration

Image registration process refers to the matching of two images that share common information such as images of the same scene or images of the same object which can be taken either at different times or using different sensors.

The image registration process has been applied in many applications in machine vision; e.g. medical imaging, computer vision, military-based image vision, and satellite images. The image registration process consists of the following steps (Istenic et. al, 2007; Zitova and Flusser, 2003):

- Feature detection,
- Feature matching,
- Transform model estimation, and
- Image sampling and transform.

Image registration techniques can be classified into different groups based on the above steps. One of the classifications was by Zitovaa and Flusser (2003). They classified image registration techniques based on feature matching properties (step 2 in the list). According to the feature matching process, the image can be classified into either Feature Based Image Registration (FBIR) or Area Based Image Registration (ABIR). FBIR techniques are based on the correspondence between features in the images such as geometrical shapes and so require a geometrical descriptor (Oztireli and Basdogan, 2008).

ABIR techniques are based on a matching criterion which computes pixel values and variations in both source and target images (Zitovaa and Flusser, 2003). ABIR techniques use statistical criteria such as normalised cross correlation in the spatial domain and the phase correlation in the Fourier domain (Hong and Zhang, 2007).

FBIR techniques are sensitive to the image content, that is to say, whether the image content is rich in detail or not (Matungka et al., 2009). ABIR techniques have been much more widely used than FBIR ones. The latter carry out the image registration process without attempting to detect the salient objects, which is a weakness compared to ABIR. However, ABIR techniques while showing good performance on individual problems of translation, rotation and scaling may fail when having to deal with combinations of translation, rotation and scaling. To overcome this weakness, the technique needs to be improved.

2.3.1. Phase Correlation and Log-polar Transformation

Phase correlation is one of the most popular ABIR techniques. The phase correlation is performed in the Fourier domain after the Fourier transform has been completed but has difficulty when dealing with combined problems of image translation, rotation and scaling. To overcome these problems the log-polar transformation is used (Reddy and Chatterji, 1996; Ding et al., 2010).

The log-polar transformation is a nonlinear and non-uniform transformation to map the Cartesian coordinate system onto the log-polar coordinate system (Zokai and Wolberg, 2005). The nonlinearity is related with the polar mapping and the non-uniformity is the property of logarithmic scaling.

2.4. Optimisation

Optimisation is a mathematical technique concerned with finding the “best” solution of a problem. The best solution is the fittest solution in the solution space.

A multitude of optimisation techniques have been used in many fields such as engineering, physics, chemistry, medicine, manufacturing and economic analysis. However, there is no optimisation technique that is suitable for all problems (Wolpert and Macready, 1997).

Optimisation techniques use the following when searching for optimum values:

- Objective functions,
- Variables,
- Constraints, and
- Search Space.

The purpose of an optimisation technique is to minimise / maximise the objective function with respect to the constraints and search space, given below:

Given:

function: $f(X)$ can be defined as $f : A \rightarrow \Re^n$

variable: $X = \{x_1, x_2, x_3, ...x_n\}$, $X \in A$ and A is subset of n dimensional Euclidian space $\Re^n$,

constraints:

inequality constraints: $g_i(X) \leq 0$, $i = 1,2,3....m$

equality constraints: $h_i(X) = 0$, $i = 1,2,3....p$

Sought:

Minimisation:

$\hat{X} \in A$ such that $f(\hat{X}) \leq f(X)$ for all $X \in A$

Maximisation:

$\hat{X} \in A$ such that $f(\hat{X}) \geq f(X)$ for all $X \in A$

The domain A of f is referred to as the "search space" or "parameter space", each element of A is called a "candidate solution", the function "$f(X)$" is called the "objective function" (Blondin, 2009). Candidate solutions are substituted in the objective function and seek the minimised / maximised value. The minimised / maximised solution of an objective function is called the "optimal solution". The optimum value searching technique is also called the "mathematical programming technique" (Rao, 1996).

2.4.1. Classification of Optimisation Techniques

This section classifies the different types of optimisation techniques. One classification is based on the types of variables, whether deterministic or stochastic variables. Deterministic variables are used in the deterministic optimisation technique and stochastic variables are used in stochastic optimisation techniques. Deterministic optimisation techniques are straightforward methods to solve the optimisation problem in polynomial time, and there is a clear relationship between the characteristics of the possible solutions and their utility. When the relationship between candidate solution and problem's fitness are complicated, not obvious, and have no solution in polynomial time, then stochastic optimisation techniques provide an alternative solution that looks for optimum value in the desired time by generating random variables. The variable based classification of the optimisation techniques is shown in Figure 2.3 (Weise, 2009). In this study, the focus is on Stochastic-based optimisation techniques. The Bees Algorithm is reviewed in detail which will be utilised in further chapters.

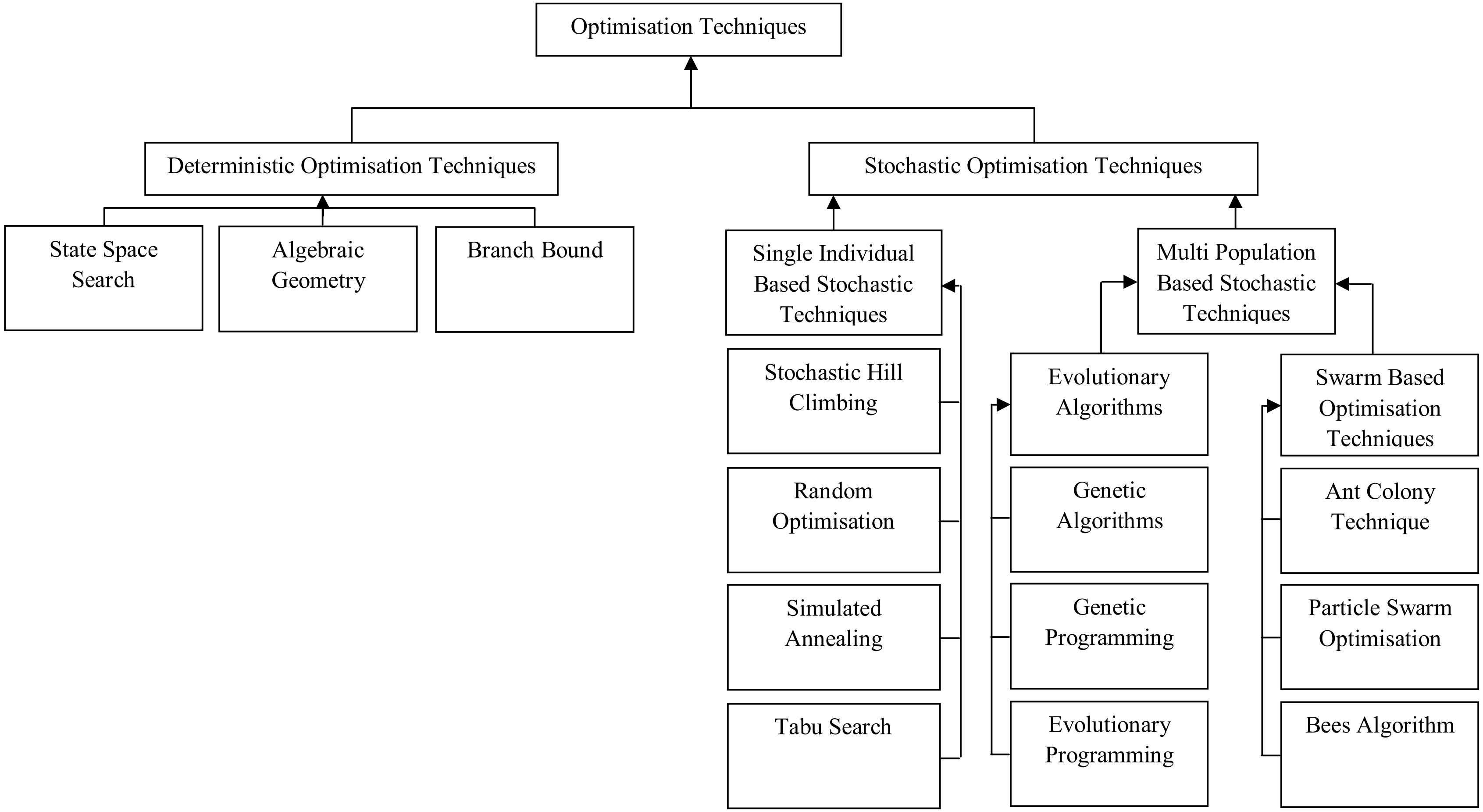


Figure 2.3 Classifications of optimisation techniques based on parameters used.

2.4.2. Stochastic Hill Climbing

The Stochastic Hill Climbing (SHC) technique is a local search technique which is based on a direct search strategy (Schmidhuber and Zhao, 1999). SCH chooses the next iteration in proportion to the steepness of an uphill move. SHC looks for the first best neighbour, and the population size of SCH is selected in a random manner (Rosete-Suarez et al., 1999; Brownlee, 2011).

2.4.3. Random Optimisation

The Random Optimisation (RO) technique is one of the simplest numerical techniques used to search for the global optimum when the gradient is difficult / impossible (Li and Rhinehart, 1998). The starting point of stochastic-based optimisation techniques is often RO (Kristoffersen, 2007).

The initial point of the RO technique is chosen randomly. There is a "*reproduce*" operator defined in RO that looks like the mutation operator of an evaluation strategy. The purpose of the reproduce operator is to reach all of the points in the search space from every other point (Weise, 2009).

2.4.4. Tabu Search

Tabu Search is a single-point local search technique with local-optima avoidance modelled on the human memory process (Rayward-Smith et al., 1996; Pham and Karaboga, 2000). The memory process is the recorded list of previously seen solutions (Gendreau and Potvin, 2010) and is called a "*tabu list*". To avoid memory problems the length of the tabu list is limited and older solutions are removed (Rothlauf, 2011). Thus the tabu list improves the local search mechanism. If the search process stops at a local minimum and this does not allow a better solution, the process cycles back to previously recorded solutions encountered before the local minimum.

2.4.5. Simulated Annealing

In metallurgical science annealing is a process which consists of the heating and controlled cooling of metals that can change some of their physical properties including strength, hardness and ductility (Koppen and et al., 2011). The annealing process is applied to relieve the metals' internal stresses so that if the temperature change is correct the metal will have the right hardness and ductility, but if the temperature change is too rapid the metal may be too brittle. Simulated Annealing (SA) is a single-point random search technique derived from the simulation of the annealing process and is a general method of locating an approximation of the global minimum / maximum when the search space is large (Koziel and Yang, 2011). SA is also considered as a version of the classical "Metropolis Method" which simulates the behaviour of atoms at a given temperature (Zhigljavsky and Zilinskas, 2008; Belegundu and Chandrupatla, 2011). The annealing process can be defined in terms of energy change (heating) modelled as a transition probability given by the Boltzmann probability distribution function as shown in Equation (2.2):

$$p = e^{-\frac{\Delta E}{k_B T}} \tag{2.2}$$

where k_B is the Boltzmann's constant, T is the temperature of the annealing process and ΔE is the energy change level of the process.

SA can be performed as follows for a stated minimisation problem. A feasible arbitrary chosen starting point is $X = \{x_1, x_2, x_3, ...x_n\}$, and its objective function $f(X)$ is selected. The step size is S and an arbitrary random number is defined as R. The initial selected X vector will be assigned as the minimum point. Then the next probable point will be predicted in the vicinity of the current point as given by Equation 2.3:

$$X_{i+1} = X_i + RS \tag{2.3}$$

The objective function value of current point is given by $f(X_i)$ and the next probable point's objective value is given by $f(X_{i+1})$. Equation (2.2) can be modified for a given SA by defining the link between the energy change ΔE and the objective function changes of problem Δf as shown in Equation 2.4:

$$\Delta E = \gamma \Delta f \tag{2.4}$$

where $\Delta f = f_{i+1} - f_i$.

γ is usually given the value unity for simplicity, see Equation 2.5:

$$p = e^{-\frac{\Delta f}{k_B T}} \tag{2.5}$$

The value of k_B affects the convergence characteristics of the method and is usually given the value unity (Rao, S., S., 1996).

The probability of the next probable point is defined by (Koppen at al., 2011) as in Equation 2.6:

$$p = \begin{cases} 1 & \text{if} \quad \Delta f \leq 0 \\ e^{-\frac{\Delta f}{T}} & \text{otherwise} \end{cases}$$

(2.6)

A promising new point X_{i+1} is accepted unconditionally when $f(X_{i+1}) \leq f(X_i)$; a non-promising new point, where $f(X_{i+1}) \geq f(X_i)$, is accepted with probability $p = e^{-\frac{\Delta f}{T}}$ when the value of $p \geq r$, where r is a random number, $0 < r < 1$, which is defined as a threshold for acceptance of a non-promising location.

In annealing an important parameter is the rate of cooling which controls the annealing process. Here the widely-used geometric cooling schedule has been implemented; see Equation 2.7:

$$T_{i+1} = T_i \alpha^i \tag{2.7}$$

where "T_i" is the current temperature, "i" is the number of the current iteration and "α" is the cooling factor which is a constant between 0 and 1.

2.4.6. Genetic Algorithm

The Genetic Algorithm (GA) is a population-based algorithm which is inspired from genetic systems in nature. The philosophical basis of the GA was proposed by John Holland in the early 1960s and its engineering applications were studied by Goldberg in 1983. In nature weak and unhealthy species are faced with extinction by natural selection, but stronger and healthier ones are able to pass their genes to future generations. The healthier genes will be passed to future generation and can be strengthened by crossover and mutational processes.

GA performs the genetic reproduction process and assesses the survival of the best individuals' strategies (Beledungu and Chandrupatla, 2011). The canonical version of GA has two fundamental operators which are crossover and mutation (Rutkowski, 2008). The canonical version consists of binary strings which represent the chromosomes of individuals.

Crossover in GAs creates offspring by randomly mixing sections of the parental genome and mutational processes generate random variations in populations, both of which are adaptation processes found in nature. The transformations can be defined as (Zelinka et al., 2010):

- Structure of child's chromosome = Hereditary material (chromosomes of organism),

- Crossover operation = Crossover in the parental chromosomes (chromosomes exchange between parental chromosomes to provide a child chromosome),
- Mutation operation = Random changes in the child's chromosomes,
- Fitness evaluation = Natural selection process in nature.

2.4.7. Evolutionary Programming

Evolutionary Programming (EP) is another type of evolutionary algorithm developed by Fogel (1960) to simulate evolution as a learning process with artificial intelligence (Back et al., 1997). EP is implemented in machine learning by finite state machines and numerical optimisation techniques (Grosan and Abraham, 2011). Traditional EP uses the Gaussian mutation operator and no crossover operator. Modern EP uses self-adaptation of the mutation operator. The population of the crossover operator will be selected by the mutation operator. EP uses the real value representation for its chromosomes. The steps in EP are given below:

- The EP technique starts by generating an initial population,
- The second step is replication of initial solutions. Each of these crossover solutions are mutated according to the Gaussian distribution function (or any chosen distribution function). The mutation operator is judged based on the distribution function,

- The last step is the evaluation of the crossover solution of population. Determination of the retained solution depends on the selected technique such as stochastic tournament.

2.4.8. Particle Swarm Optimisation

Particle Swarm Optimisation (PSO) is a type of Swarm-Based Optimisation Technique (SBOT) which was inspired by the behaviour of socially organised populations in nature such as animal herds or bird flocks (Blum and Merkle, 2008). PSO was proposed by Kennedy and Eberhart, (1995). The population in PSO is called the swarm and each individual is called a particle (Li and Liu, 2011). Each particle represents a possible solution of the problem (Lamm and Unger, 2011). The particles in PSO collectively search for the global optimum with given velocities (Floreano and Mattiussi, 2008). The velocities of individual particles are stochastically adjusted based on the previous best position of each individual particle. The performance of the algorithm is measured with a fitness function and it will run until any given termination criterion is satisfied.

2.4.9. Ant Colony Optimisation

Ant Colony Optimisation (ACO) technique is also an SBOT, which was inspired by pheromone-based strategies of ant foraging.

When looking for food, ants initially randomly search the area immediately surrounding their nest. After finding a food source an ant carries some of it back to the nest. During this return trip, the ant deposits trail of a chemical pheromone on the ground. The quantity of pheromone deposited is an indicator to other ants of the direction to the food source and the quantity and quality of the source (Dorigo & Stützle, 2004).

In nature, the foraging behaviour of ants is based on finding the shortest path between the source and their nest (Panigrahi et al, 2011). When ants find a food source, they leave their pheromone trails on the path when they return to their nest from the source and the other ants find the path by using the pheromone trails and the pheromone level. If the selected path is the shortest path, the pheromone level will be reinforced in other ways but it will evaporate as time passes (Sumathi and Surekha, 2010). A short path will be visited by more ants, and thus the pheromone level will be higher compared to other paths. ACO was inspired by the above behaviour of ants and applied to optimisation problems to determine the optimum value. ACO was introduced by Dorigo et al., (1996), for solving combinatorial optimisation problems such as the Travelling Salesman Problem (TSP), which entails the cost function being optimised. The construction of a candidate solution is done according to a probabilistic state transition rule; a candidate solution is defined as moving from one node (i) to another (j), which is given in Equation 2.8:

$$p_{ij}^{m}(t) = \begin{cases} \dfrac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum_{k\in N_i^m(t)} \tau_{ik}^{\alpha}(t)\eta_{ik}^{\beta}(t)} & if\ j \in N^m(t) \\ 0 & if\ j \notin N^m(t) \end{cases} \tag{2.8}$$

where $p_{ij}^{m}(t)$: Transition probability from node i to node j ,

τ_{ij} : The posterior effectiveness of the move from node i to node j ,

α : The parameter to control the influence of τ_{ij} ,

η_{ij} : The prior effectiveness of the move from node i to node j ,

β : The parameter to control the influence of η_{ij} ,

N_{i}^{m} : The set of feasible nodes for ant m when located on node i .

The pheromone concentration τ_{ij} represents how desirable it is to make a move from node i to node j . The evaporation of pheromone is given in Equation 2.9:

$$\tau_{ij}(t) = (1-\rho)\tau_{ij}(t) \tag{2.9}$$

where ρ : The reduction rate of the pheromone level, ($\rho \in [0.1]$).

The purpose of evaporation is to avoid all the ants premature converging on a single good solution and to have diversification in the search space.

The pheromone update starts with completion of a path by each ant, which is associated with the best found solution among all ants. The added value depends on the quality of the selected solutions. The update process is shown in Equations 2.10-2.11:

$$\tau_{ij}(t+1) = \tau_{ij}(t) + \Delta\tau_{ij}(t) \tag{2.10}$$

$$\Delta\tau_{ij} = \sum_{m=1}^{n_A} \Delta\tau_{ij}^{m}(t) \tag{2.11}$$

where n_A : The number of ants,

$\Delta\tau_{ij}^{m}(t)$: The amount of pheromone deposited by ant m from node i to node j at time step t.

2.4.10. The Fundamentals of the Bees Algorithm

The Bees Algorithm is a type of SBOT, which was proposed by Professor D. T. Pham and his team at MEC-Cardiff University in 2005 (Pham et. al. 2005; 2006a).

The Bees Algorithm was inspired by the food foraging behaviour of honey bees. A swarm of honey bee colony consists of a queen bee, which is the mother of all colony members, and thousands of worker bees (Seeley, 1995). After honey bees emerge from their cell, they have a duty to clean the cell, store food, construct the comb, feed the larva and when they are three weeks old, they start to forage and stop doing the other tasks.

2.4.10.1. The Foraging Behaviour of Honey Bees

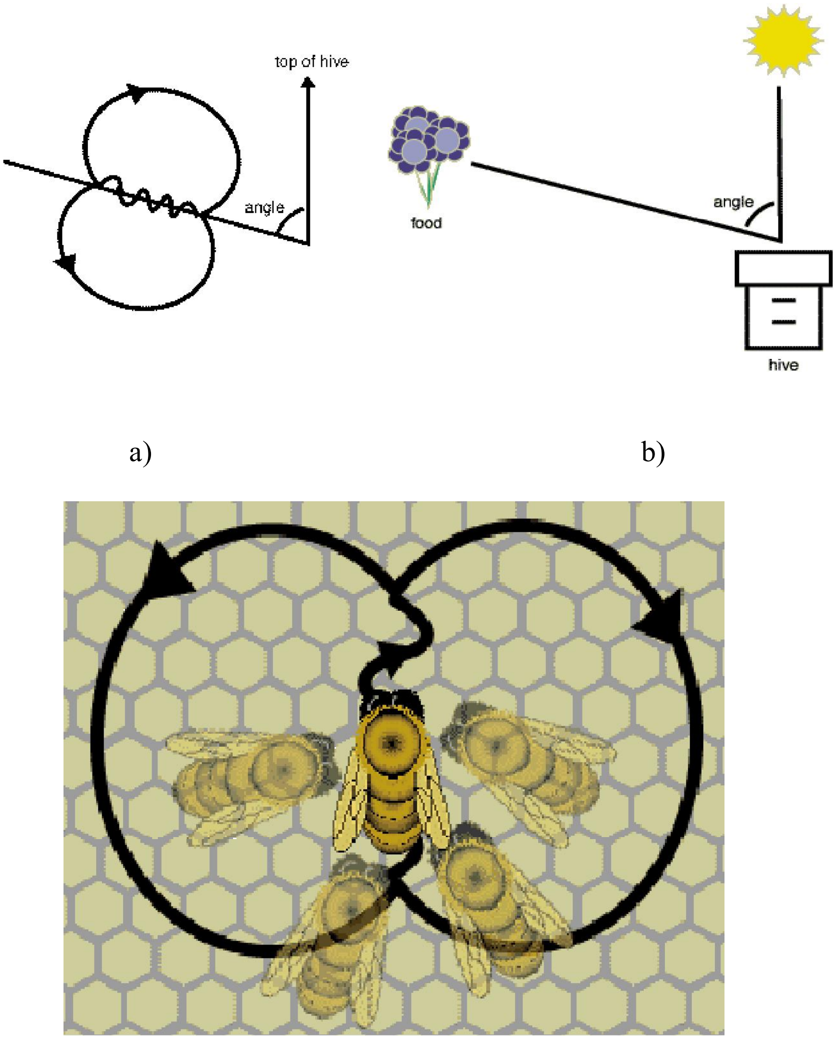
A colony of honey bees can exploit a large number of food sources in big fields and they can fly up to 11 km to exploit food sources (Seeley, 1995 and Gould and Gould, 1988). The colony employs about one-quarter of its members as forager bees. The foraging process begins with searching out promising flower patches by scout bees. The colony keeps a percentage of the scout bees during the harvesting season. When the scout bees have found a flower patch, they will look further in hope of finding an even better one (Gould and Gould, 1988). The scout bees search for the better patches randomly (Von Frisch, 1955).

The scout bees inform their peers waiting in the hive as to the quality of the food source, based amongst other things, on sugar levels. The scout bees deposit their nectar and go to the dance floor in front of the hive to communicate to the other bees by performing their dance, known as the ‘waggle dance’ (Seeley, 1995).

2.4.10.2. The Waggle Dance of Honey Bees

The waggle dance is named from the wagging run (in which the dancers produce a loud buzzing sound by moving their body from side to side), which is used by the scout bees to communicate information about the food source to the rest of the colony. The scout bees provide the following information by means of the waggle dance: the quality of the food source, the distance of the source from the hive and the direction of the source (Gould and Gould, 1988; Von Frisch, 1955).

The waggle dance path has a figure of eight shape. Initially the scout bee vibrates its wing muscles which produces a loud buzz and runs in a straight line the direction of which is related to the vertical on the hive and indicates the direction of the food source relative to the sun’s azimuth in the field, see Figure 2.4 a and b (Huang, 2008). The scout then circles back, alternating a left and a right return path (Talbi, 2009). The speed / duration of the dance indicates the distance to the food source; the frequency of the waggles in the dance and buzzing convey the quality of the source; see Figure 2.4c (Huang 2008). These will influence the number of follower bees.



a)

b)

c)

Figure 2.4 a) Orientation of waggle dance with respect to the sun, b) Orientation of waggle dance with respect to the food source, hive and sun, c) The Waggle Dance and followers.

2.4.10.3. Applications of the Bees Algorithm

This section reports the applications of the Bees Algorithm to various types of optimisation problem.

The first application of the Bees Algorithm was on the continuous type benchmark functions optimisation when the algorithm was performed on nine continuous type benchmark functions (Pham et al., 2006a). The Bees Algorithm was then tested on more continuous functions and the results compared with those of other optimisation algorithms (Pham and Castellani, 2009a). It was found that the performance of the Bees Algorithm was very effective on the benchmark function compared to the other algorithms.

The Bees Algorithm was also implemented on multi-objective optimisation problems. The first application was on the welded beam problem by (Ghanbarzadeh, 2007). The goal of the study was to minimise the cost of fabrication by finding a feasible weld thickness, weld length, beam thickness and beam width under the stress constraints. The second application was on a carbon energy system and environmental dispatch problem (Lee, 2010). The aim of the study was to design a low carbon system by minimising both total cost and CO_2 emission. The performance of the Bees Algorithm was most promising compared to all the other examined algorithms.

Another application of the Bees Algorithm was with neural networks. The first implementation was on the training of a Learning Vector Quantisation network

(Pham et al., 2006b). The second was the training of a Multi Layered Perceptron neural network (Pham et al., 2006c; Koc, 2010). The third was on the training of a Radial Basis neural network (Pham et al., 2006d). The simulation results of training and testing proved that the Bees Algorithm is a strong classifier and optimisation tool.

The Bees Algorithm was also implemented on manufacturing systems. Pham et al., (2007a) applied the Bees algorithm to cellular manufacturing systems where the cell information problem was optimised by the Bees Algorithm. The results showed that the Bees Algorithm is applicable for combinatorial applications.

The Bees Algorithm was tested on the job scheduling problem and the results were compared with TS, GA, PSO and their combinations (Pham et al., 2007b). The Bees Algorithm performed better than other optimisation algorithms on this combinatorial optimisation problem.

Another application of the Bees Algorithm was on clustering problems. The Bees Algorithm was implemented on the K-means and C-means clustering algorithms (Pham et al., 2007c; Al-Jabbouli, 2009). And the results showed that the Bees Algorithm could be a powerful tool for clustering applications.

A robotic application of the Bees Algorithm has been proposed (Pham et al., 2008). The Bees Algorithm was employed for learning the inverse kinematics of a robot manipulator. The results from this application were excellent. Another robotic application was proposed by (Pham et al., 2009b), whereby the Bees Algorithm was employed to optimise the membership functions of fuzzy logic. The proposed algorithm tuned the fuzzy logic controller parameter for stabilising and balancing an acrobatic robot. The results showed that the Bees Algorithm improved the gain of the fuzzy logic controller.

The performance of the Bees Algorithm was improved using a hybrid approach such as combining it with PSO (Sholedolu, 2009). PSO has advantages in an adaptive neighbourhood search and by including the advantages of PSO in the Bees Algorithm; the convergence rate of the algorithm (called modified PSO) was improved. The performance of the PSO - Bees hybrid algorithm on the examined problems was very promising and fast.

The Bees Algorithm can be optimised, tuning its parameters will provide an even more robust and efficient algorithm. One of these improvements was carried out by (Otri, 2011) who improved the Bees Algorithm by adapting it to optimise itself and to decrease the sensitivity of the parameters.

The Bees algorithm has also several enhancements, one of the enhancements was proposed based on an early neighbourhood search and efficiency based approach and tested on benchmark function (Pham et al., 2012a). According to the results, the proposed enhanced Bees Algorithm performed better than other optimisation algorithms.

From general considerations, it has been said that there will be no single optimisation algorithm that will perform equally well on all problems (Wolpert and Macready 1997). However, the performance of the Bees Algorithm is very impressive because it has the capability to perform both local and global searches using exploitation and exploration strategies. These features are the main strengths of the Bees Algorithm and it has been shown to perform very effectively on a wide range of problems. The algorithm was also available in the author's laboratory where extensive research had been carried out on it. Therefore it was selected for this study.

Further, the Bees Algorithm will be utilised in blind deconvolution and the results of the proposed algorithm will be compared with a Simulated Annealing-based blind deconvolution technique in chapter 5.

2.5. Image Deconvolution

Image formation can be identified as the transformation of a 3D object onto the 2D sensor plane which is a depth-weighted, mapped from the image intensity of the object. The image formation process uses the data from a sensor and prior information about the image. The mathematical modelling of image formation can be written using a convolution given in Equation 2.1. The convolution operation can be denoted with "*" operator, so Equation 2.1 can be written as:

$$g(x,y) = f(x,y) * h(x,y) \tag{2.12}$$

If the observed image contains noise then Equation 2.12 can be rewritten as:

$$g(x,y) = f(x,y) * h(x,y) + n(x,y) \tag{2.13}$$

where $g(x,y)$ is the observed image, $f(x,y)$ is the original image, $h(x,y)$ is the PSF of the imaging system and $n(x,y)$ is the noise in the observed image.

Equation 2.13 can be written in the Fourier domain as:

$$G(u,v) = F(u,v)H(u,v) + N(u,v) \tag{2.14}$$

where $G(u,v)$, $F(u,v)$, $H(u,v)$ and $N(u,v)$ are the Fourier transforms of $g(x,y)$, $f(x,y)$, $h(x,y)$ and $n(x,y)$, respectively.

The original image can be recovered using a deconvolution process which is the inverse process of the convolution operation. In the spatial domain this inverse operation is not cost efficient. Thus the Fourier domain operation gives a better and faster response for the deconvolution process, as shown in Equation 2.15 (Madden et al., 1996):

$$F(u,v) = \frac{G(u,v)}{H(u,v)} - \frac{N(u,v)}{H(u,v)} \tag{2.15}$$

The inverse Fourier transform of Equation 2.15 gives the original image $f(x,y)$. With respect to the deconvolution problem, it is assumed that the PSF of the original image is known. When there is no information about the PSF, deconvolution becomes a very complex and difficult process. Recovering the original image and PSF without or having less prior information is, however, possible with blind image deconvolution.

2.5.1. Blind Image Deconvolution

Blind Image Deconvolution (BID) is a fundamental process in machine vision. BID looks for the original image and its PSF when there is only partial information or no information available about the original image and its PSF.

Several methods have been proposed for finding the original image and PSF. One of the most basic approaches is to obtain information about the original image; physical information about the original image such as the non-negativity and finite support, or statistical information such as entropy or probability density function (Kundur and Hatzinakos, 1996).

Kundur and Hatzinakos (1996) grouped BID techniques into five main categories:

- Zero sheet separation methods,
- A priori blur identification methods,
- Auto-regressive moving average methods,
- Non-parametric methods based on high order statistics methods, and

- Non-parametric deterministic image constraints restoration methods.

The Zero Sheet Separation (ZSS) methods were developed by Lane and Bates (1987) and are computationally intensive. They use the z-transform and assume that the observed image is formed by the convolving of several individual components.

The Priori Blur Identification (PBI) methods are based on the estimation of prior information about the PSF to use in the BID model. The technique works with a known parametric form of the blur such as the linear camera motion of an out of focus system (Kundur and Hatzinakos, 1996). Liu et al. (2003) employed a PBI method with information of the blur type and this information was used as a blur prior to the Maximum A-Posteriori (MAP) to recover the original image and PSF. The image recovery process cannot perform well when there is no parametric form of the blur type.

Auto-Regressive Moving Average (ARMA) methods are one of the most widely used methods in BID. The original image is modelled as an Auto-Regressive (AR) process and the PSF is modelled as a Moving Average (MA) (Blume, 2007). Thus, the BID problem is transformed into an ARMA parameter estimation problem (Yap et al., 2003). To overcome the problem of accurate prediction of parameters under conditions of instability and non-uniqueness, additional techniques have been employed such as Maximum Likelihood (ML), Generalised

Cross Validation (GCV) and Neural Networks (NN), which are distinct among ARMA methods (Reeves and Mersereau, 1992).

Non-Parametric Methods Based on the Higher Order Statistics have been used for solving the BID problem. The methods are based on the minimisation of the given cost function that accounts for the probabilistic non-Gaussian nature of the original image (Kundur and Hatzinakos, 1996; Jayaraman et al., 2009). Typically these models have been applied when the image or the edges are modelled as sparse (Campisi and Egiazarian, 2007).

Non-Parametric Deterministic Image Constraints Restoration (NPDICR) methods use deterministic constraints with an iterative solution approach instead of using a parametric model for the original image or the PSF. NPDICR methods are different from the rest of BID methods (Kundur and Hatzinakos, 1996). BID techniques proposed as NPDICR methods include iterative blind image deconvolution (Lam and Goodman, 2000; Biggs and Andrews 1997; Ayers and Dainty, 1988), genetic algorithm-based blind image deconvolution (Chen et al., 1996; Yin, 2006), blind image deconvolution using least squares minimisation (Law and Lane, 1996) and simulated annealing-based blind image deconvolution (McCallum, 1990; Yu, 2008). Each of these techniques has an iterative-based approach to find the optimum solution for the sharp image and PSF. Apart from the model of Yu (2008) all other techniques try to recover the original image from the observed image by a recursive degradation model. Yu (2008) used an

Independent Component Analysis (ICA) based non–Gaussianity analysis with a simulated annealing technique.

2.5.2. ICA-Based Non-Gaussianity Analysis

Independent Component Analysis (ICA) is one of the Blind Source Separation (BSS) methods, which has been used for separating data into its original informational components (Stone, 2004). ICA is widely used in the signal processing and image processing area. It assumes that the observed signal is a linear combination of each independent component, as given in Equation 2.16 (Hyvarinen et al., 2001):

$$X^* = a_1 S_1 + a_2 S_2 + \ldots + a_n S_n \tag{2.16}$$

where X^* is observed signal, $a_{1,}, a_2, \ldots a_n$ are the mixing coefficients of the n components respectively and $S_1, S_2, \ldots S_n$ are the source signals.

It is assumed that all the components of the observed signal are statistically independent and non-Gaussian (Wang et al., 2009). The non-Gaussianity is based on the Central Limit Theorem, which states that the distribution of the sum of n independent random variables approaches the Gaussian. It means that the observed signal is closer to the Gaussian distribution than its components.

A blurred image can also be separated into two main components, the original image and the PSF, using non-Gaussianity analysis. The non-Gaussianity can be measured with the following techniques:

- Negentropy analysis,
- Approximated negentropy analysis,
- Mutual information analysis, and
- Kurtosis analysis.

2.5.2.1. Negentropy Analysis

Negentropy is a non-Gaussianity measurement technique which is an information theoretic unit contrast function. Negentropy is based on differential entropy. The entropy of a random variable can be defined as a degree of information that the observation of the variables gives. The entropy of a discrete variable can be written as in Equation 2.17:

$$H(Y) = -\sum P(Y) log P(Y) \tag{2.17}$$

where Y denotes a discrete random variable, $H(Y)$ is the entropy of $P(Y)$ that denotes the probability density function of Y.

The entropy is identified as differential entropy for continuous type variables, given in Equation 2.18 (Wang et al., 2009):

$$H(y) = -\int f(y) \log f(y) dy \tag{2.18}$$

The negentropy can be defined as:

$$J(y) = H(y_{gauss}) - H(y) \tag{2.19}$$

where y is a continuous type random variable, $J(y)$ denotes the negentropy of y, y_{gauss} is a Gaussian random variable of the same covariance matrix as y.

2.5.2.2. Approximated Negentropy Analysis

The estimation of the negentropy is a difficult task and thus it remains as a theoretical function (Hyvarinen et al., 2001). However it has been proposed that some approximated negentropy functions can be used to measure non-Gaussianity. One of these approaches is to use higher-order moment functions, as given in Equation 2.20 (Hyvarinen, 1998):

$$J(y) \approx \frac{1}{12} E\{y^3\}^2 + \frac{1}{48} kurtosis(y)^2 \tag{2.20}$$

where y is a random variable which is assumed to be of zero mean and unit variance, $E\{y^3\}$ is the third moment function of y, $kurtosis(y)$ denotes the kurtosis.

Another approximation to the negentropy is based on the maximum entropy principle (Hyvarinen, 1998). It is shown in Equation 2.22:

$$J(y) \approx \sum_{i=1}^{p} k_i \left[E\{G_i(y)\} - E\{G_i(v)\} \right]^2 \tag{2.22}$$

where y is assumed to be a random variable with zero mean and unit variance, v is a Gaussian variable of zero mean and unit variance, k_i are the constants, $E(G_i(y))$ are the expected value of the $G_i(y)$ functions, and $J(y)$ is the negentropy of y, $p = 2$. For simplicity, G functions are chosen as in Equation 2.23 and 2.24 (Hyvarinen, 1999):

$$G_1(x) = log\, cosh\, a_1 x \tag{2.23}$$

$$G_2(x) = e^{\frac{-a_2 x^2}{2}} \tag{2.24}$$

where $a_1, a_2 \geq 1$ are constants.

2.5.2.3. Mutual Information Analysis

Mutual information is a measure between random variables that can be used to measure the amount of information shared between them (Hyvarinen, 1999). The shared information among n random variables is shown in Equation 2.25:

$$I(y_1, y_2, \ldots, y_n) = \sum_{i=1}^{n} H(y_i) - H(y) \tag{2.25}$$

where $y_1, y_2, ..., y_n$ are the n random variables, $I(y_1, y_2, ... y_n)$ denotes the mutual information among n random variables, $H(y_i)$ is the entropy of the ith element and $H(y)$ is the joint entropy.

It is expected that the mutual information between independent components is zero. Therefore, the minimisation of the mutual information can be used for non-Gaussianity analysis. The determination of non-Gaussianity with mutual information-based techniques requires a heavy computational process.

2.5.2.4. Kurtosis Analysis

Kurtosis, which is a fourth-order cumulant, is one the classical measures for non-Gaussianity. Kurtosis is a measure of distribution, indicating whether the data are peaked or flatter than a Normal distribution which has a zero mean and unit variance (Crawley, 2005). The kurtosis of a random variable y is given in Equation 2.26:

$$kurtosis(y) = E\{y^4\} - 3(E\{y^2\})^2 \tag{2.26}$$

If it is assumed that y has unit variance, Equation 2.26 can be rewritten as:

$$kurtosis(y) = E\{y^4\} - 3 \tag{2.27}$$

Kurtosis can be negative, positive or zero. The kurtosis of a Gaussian random variable y is zero; on the other hand the kurtosis of a non-Gaussian random variable is non-zero. If the random variable has a negative kurtosis, it is called a sub-Gaussian such as a Raised Cosine distribution (RCD) which has a kurtosis value of -0.5937 (Sun, 2010). When the kurtosis of the random variable is positive, it is called super-gaussian such as a Laplace Distribution (LD) which has a kurtosis value of 3 (Sun, 2010). Density graphs for the RCD, LD and Normal distributions, are shown in Figure 2.5.

The calculation process of kurtosis is easier than calculation of negentropy, approximated negentropy and mutual information. Therefore kurtosis analysis will be utilised in the ICA-based image recovery process to determine the original image and PSF.

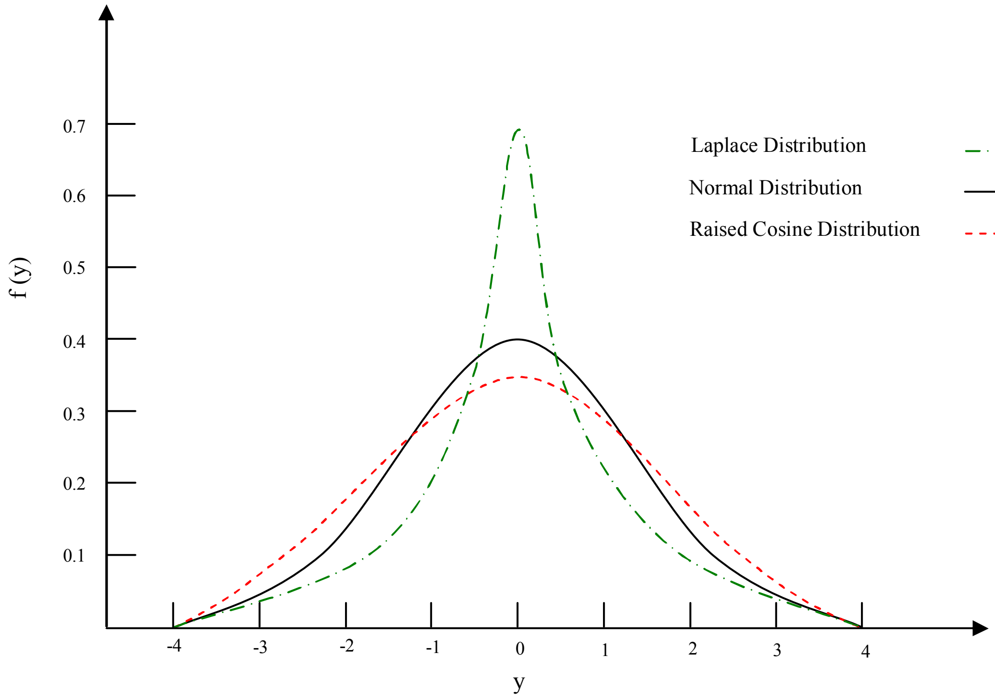


Figure 2.5. Density graphs for Laplace distribution, Gaussian distribution and Raised Cosine distribution.

2.6. Summary

This chapter has briefly described the DFAF technique, image registration techniques, optimisation techniques, deconvolution and BID techniques. Kurtosis-based non-Gaussianity analysis has been highlighted. The aim of this chapter was to provide information on and background for the subsequent chapters. In Section 2.2, the image formation in SEM and DFAF technique is presented and will be discussed with phase correlation and logarithmic polar transformation in Chapter 3. In section 2.4, the stochastic optimisation techniques were discussed. One of the stochastic based optimisation algorithms, the Bees Algorithm, is presented in detail, which will be used in Chapters 4 and 5. In Chapter 4, an improvement to the Bees Algorithm will be presented. Finally, a novel depth computation technique will be described in Chapter 5 based on BID optimised with the Bees Algorithm for SEM images and the results of proposed algorithm will be compared with a Simulated Annealing-based optimised BID. Further, the Bees Algorithm-based optimised BID will be utilised to determine the depth between surface and objective lens in SEM. Note that none of the techniques surveyed are related to blind deconvolution and optimisation with depth computation.

Chapter 3

DEPTH FROM AUTOMATIC FOCUSING USING PRE-PROCESSED IMAGE REGISTRATION TECHNIQUES

3.1. Preliminaries

DFAF is a popular technique for depth computation and has been used on various types of imaging systems such as white-light microscopes, white-light interferometers and Transmission Electron Microscopes (TEM). In this study, the DFAF technique is applied to a Scanning Electron Microscope (SEM) image after an image correction process.

SEM is one of the widely used vision systems in micro and nano imaging. However, SEM images may contain distortion, rotation and translation due to the interaction between the surface of the sample and the electron beam. Therefore, an image correction process is necessary to correct these problems with SEM images.

Image correction is an important part of image processing and in this study an image correction process based on the Phase Correlation and Log-Polar Transformation (PCLPT) technique for SEM images is proposed. The PCLPT technique has been improved with a pre-processing step needed because of interpolation error at the sub pixel level. The pre-processing step consists of applying image filtering and the corrected image is obtained using the DFAF technique to determine the depth.

3.2. Image Correction with PCLPT Technique

Phase correlation is used for the image translation problem but does not perform well on image rotation and scaling problems (Reddy and Chatterji, 1996). Therefore, as described in Chapter 2, phase correlation and log-polar transformation (PCLPT) technique is applied to overcome all three problems (Ding et al., 2010). In the following sub-sections each of the problems is explained sequentially.

3.2.1. Image Translation Correction with Phase Correlation Technique

Image translation is of major concern in the image registration area. The translation between observed image and reference image can be defined as:

$$I_2(x,y) = I_1(x-\Delta x, y-\Delta y) \tag{3.1}$$

where $I_1(x,y)$ and $I_2(x,y)$ are the reference image and observed image, respectively; Δx and Δy are the displacements in x and y directions respectively.

The translation problem can be solved in the Fourier domain with the phase correlation technique. The translation can be written in the Fourier domain as in Equation 3.2 (Reddy and Chatterji, 1996):

$$F_2(u,v) = e^{-i2\pi(u\Delta x + v\Delta y)} F_1(u,v) \tag{3.2}$$

where $F_1(u,v)$ and $F_2(u,v)$ are the Fourier transforms of $I_1(x,y)$ and $I_2(x,y)$ respectively.

The phase correlation of the observed image and reference image is defined as:

$$PC = e^{i2\pi(u\Delta x + v\Delta y)} = \frac{F_1(u,v)F_2^{*}(u.v)}{\left|F_1(u,v)F_2^{*}(u.v)\right|} \tag{3.3}$$

where $F_2^{*}(u,v)$ is the complex conjugate of $F_2(u,v)$.

The translations in both x and y directions are found with an inverse Fourier transform of the phase correlation as:

$$(\Delta x, \Delta y) = max\left\{F^{-1}(PC)\right\} \tag{3.4}$$

where $F^{-1}(PC)$ denotes the inverse Fourier transform of the phase correlation.

The inverse Fourier transform of the phase correlation comprises zero elements everywhere except at the position of the displacement. Because the inverse Fourier transform of the complex exponential is a delta function, the result has only one peak, and the position (both x and y directions) of the peak gives the translation between reference image and observed image.

In practice, the translated image is expected to be a linear shift of the reference image, rather than a circular shift. If the translated image is also circularly shifted relative to the reference image, then $F^{-1}(PC)$ is not expected to be a delta function and the performance of the phase correlation is likely to decrease. To avoid low performance, a window function can be applied such as the Hann window, the Hammingham window or the Blackman window.

In this study, phase correlation was tested on images of Lena, Boats, Cameraman which have been used as benchmark images by many researchers (Levente, 2003), and a "copper sample". The copper sample image was captured with an SEM XB-1504 in MEC Cardiff University for this study. The original images are provided in Figure 3.1 a-d. Then, the original images were translated in x and y directions by -40 pixels and 20 pixels respectively. The translated images are given in Figure 3.2 a-d.

The phase correlation technique uses both the original image and the translated image. The results for the corrected images are shown in Figure 3.3 a-d. The translated image of Lena, Boats, Cameraman and a copper sample captured with the SEM were moved back to the original position according to the results of the phase correlation technique from the centre of translated images and all the results for the translation correction were found 100% correct by checking manually with Gimp 2 Open source image processing software.

a) b) c) d)

Figure 3.1 The test images a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.2 The test images translated in x and y directions by -40 pixels and 20 pixels respectively a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.3 Corrected results for the translated images a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

3.2.2. Image Rotation Correction with a Polar Transformation Based Phase Correlation Technique

Image rotation is another important task in the field of image correction. The rotation can be defined as a translation problem in the Fourier domain obtained by using a polar transformation (Ding et al., 2010). Polar Transformation (PT) is a nonlinear and non-uniform transformation to map Cartesian coordinates into polar (Zokai and Wolberg, 2005).

Any given point, (x,y) in the Cartesian coordinate system can be mapped into the polar coordinate system in the form of $(\rho, \ \theta)$ relative to a given reference point (x_0,y_0), where ρ is the radial distance from the reference to the given point, see Equation 3.5, and θ is the angle made by the line joining the reference point to the given point see Equation 3.6:

$$\rho = \sqrt{(x-x_0)^2+(y-y_0)^2} \tag{3.5}$$

$$\theta = tan^{-1}\left(\frac{y-y_0}{x-x_0}\right) \tag{3.6}$$

The relation between the observed image (rotated β° clockwise about the origin) and the reference image can be defined as:

$$I_2(x,y) = I_1(x\,cos(\beta) + y\,sin(\beta), -x\,sin(\beta) + y\,sin(\beta)) \tag{3.7}$$

The Fourier transform of the rotated image is defined as in Equation 3.8, which is called the Fourier rotation property (Liu and Mason, 2009).

$$F_2(u,v) = F_1(u\cos(\beta) + v\sin(\beta), -u\sin(\beta) + v\cos(\beta)) \tag{3.8}$$

To transform the rotation problem into a translation problem, transformation of Equation 3.8 into the polar domain is required, as given in Equation 3.9:

$$F_2(\rho,\theta) = F_1(\rho,\theta - \beta) \tag{3.9}$$

The rotation is defined as the angular difference in the Fourier domain and it can be determined with phase correlation.

In this study, the same four images as used for translation were also used for rotation. For ease of reference the original images are again shown in Figure 3.4 a-d. These were rotated by 20° in a counter clockwise direction, see Figure 3.5 a-d. The rotated images were rotated back to their original position by rotated clockwise 20° from the centre of the rotated image with proposed technique. The results were determined with PT-based PC. All the results were found 100% correct by checking manually with Gimp2 open source package, given in Figure 3.6 a-d.

a) b) c) d)

Figure 3.4 The test images a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.5 The test images rotated through -20^0 a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.6 Corrected results for the rotated images a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

3.2.3. Image Scaling Correction with Logarithmic Transformation - Based Phase Correlation Technique

Image scaling is another major task in the image processing field, which can be solved as a translation problem using the Logarithmic Transformation (LT) (Reddy and Chatterji, 1996).

When an image is scaled by factors γ_1 and γ_2 in the x and y direction respectively (γ_1 and γ_2 are real numbers), then it can be modelled in the spatial domain as:

$$I_2(x,y) = I_1(\gamma_1 x, \gamma_2 y) \tag{3.10}$$

The Fourier transform of Equation 3.10 can be written as Equation 3.11, in terms of the Fourier scaling property.

$$F_2(u,v) = \frac{1}{\gamma_1 \gamma_2} F_1\left(\frac{u}{\gamma_1}, \frac{v}{\gamma_2}\right) \tag{3.11}$$

To find the scale factor for both directions, both sides of Equation 3.11 need to undergo LT. Ignoring the multiplicative factor ($1/\ \gamma_1\gamma_2$) this can be written as Equation 3.12 (Liu and Mason, 2009).

$$F_2(log_\alpha(u), log_\alpha(v)) = F_1(log_\alpha(u) - log_\alpha(\gamma_1), log_\alpha(v) - log_\alpha(\gamma_2)) \tag{3.12}$$

where α is the base of the logarithm.

According to Equation 3.12, the scaling factors can be defined as phase differences in the Fourier domain and the equation can be solved with the phase correlation technique. The scaling problem was tested on the four images; again the originals are included for ease of comparison in Figure 3.7 a-d. All the images were scaled by 75% of the original sizes ($\gamma 1 = \gamma 2 = 0.75$) and are shown in Figure 3.8 a-d. The scale correction is accomplished with phase correlation based on the LT in the Fourier domain. The scaling factor was found as 1.3333 (1/0.75 = 1.3333 for all entire images, and the given image were scaled by 1.3333 times. The corrected images are shown in Figure 3.9 a-d.

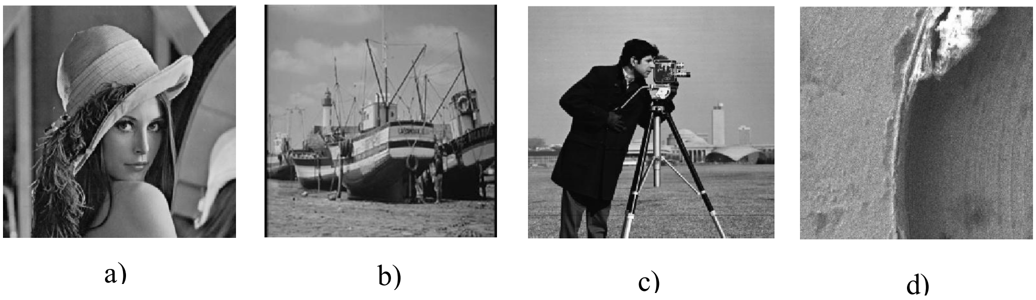

a) b) c) d)

Figure 3.7 The test images a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

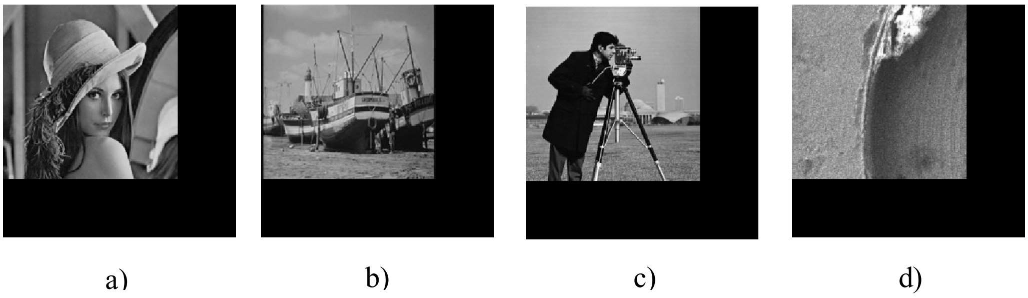

a) b) c) d)

Figure 3.8 The test images scaled by 0.75 in both x and y directions a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

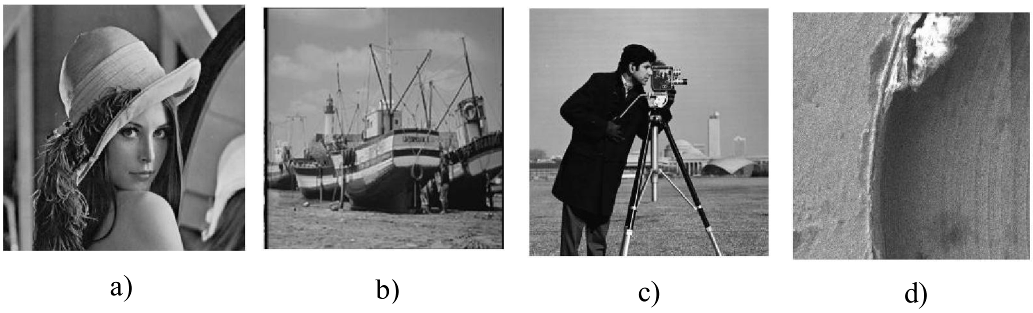

a) b) c) d)

Figure 3.9 The corrected results for the scaled images a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

3.3. Image Correction for the Combined Problems of Rotation-Translation-Scaling

Translation, rotation and scaling problems have been discussed individually in the previous sections. In this section, the technique of combining phase correlation with Log-Polar Transformation (PCLPT) is used to solve the combined problem of translation, rotation and scaling. The general form of a translated, rotated and scaled image is shown in Equation 3.13 and the Fourier transform of Equation 3.13 is given in Equation 3.14:

$$I_2(x,y) = I_1((x\cos(\beta) + y\sin(\beta))\gamma - \Delta x, (-x\sin(\beta) + y\cos(\beta))\gamma - \Delta y) \qquad (3.13)$$

$$F_2(u,v) = e^{-i2\pi(u\Delta x + v\Delta y)} \frac{1}{\gamma^2} F_1\left(\frac{u\cos(\beta) + v\sin(\beta)}{\gamma}, \frac{-u\sin(\beta) + v\cos(\beta)}{\gamma}\right) \qquad (3.14)$$

For simplicity, the x and y scale factors were selected equal (= γ) (Reddy and Chatterji, 1996). The exponential polar domain form of Equation 3.14 is given in Equation 3.15:

$$F_2(\rho,\theta) = e^{-i2\pi(u\Delta x + v\Delta y)} \frac{1}{\gamma^2} F_1\left(\frac{\rho}{\gamma}, \theta - \beta\right) \qquad (3.15)$$

To simplify the above equation, the magnitude of the Fourier transform can be utilised, as in Equation 3.16:

$$M_2(\rho,\theta)=\frac{1}{\gamma^2}M_1(\frac{\rho}{\gamma},\theta-\beta) \tag{3.16}$$

The multiplicative factor $\frac{1}{\gamma^2}$ is ignored, and for simplicity both sides are expressed in a logarithmic form, see Equation 3.17:

$$M_2(log_\alpha(\rho),\theta)=M_1(log_\alpha(\rho)-log_\alpha(\gamma),\theta-\beta) \tag{3.17}$$

The rotation and scaling problems have been transformed to a translation problem, as given in Equation 3.17 and so they can be solved with the phase correlation technique. The computed results of the scaling factor and rotation angle are substituted in Equation 3.15 to determine the amount of translation between two images. To test the performance of the PCLPT technique on the translated, rotated and scaled image, the four images are given in Figure 3.10 a-d. The four test images were firstly rotated by 20° counter clockwise, then translated by (-40, 20) pixels in the (x, y) image coordinate system and finally scaled by 75% from the original size. The modified images are given in Figure 3.11 a-d. The corrected images are shown in Figure 3.12 a-d.

a) b) c) d)

Figure 3.10 The test images a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.11 The test images translated in x and y directions by -40 pixels and 20 pixels respectively, rotated through -20^0 and scaled by 0.75 in both x and y directions a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.12 The correction results for the translated-rotated -scaled images a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

The results for the translation, rotation and scaling tests are given in Table 3.1, and it can be seen that the PCLPT technique did not perform well for the images of Lena and Boats. In the case of Lena, the PCLPT technique could not compute the correct amount of the translation and in the case of the Boats the PCLPT technique was unable to compute either the correct image translation or rotation.

Thus the image correction process failed for the images of Lena and Boats. The computational errors in both cases are related to an interpolation error at the sub-pixel level (Pan et al., 2009). To overcome the computational error with PCLPT, a pre-processing stage is introduced in the following section.

In Table 3.1, the results of the PCLPT technique is given in column denoted with PCLPT. Absolute Difference between expected results and predicted result from PCLPT is utilised to measure the error, which is denoted as Abs. Diff, (see Table 3.1).

Table 3.1 Results of image correction with PCLPT technique.

Test Images	Translation (pixel)						Rotation (degree)			Scaling		
	X-direction			Y-direction								
	PCLPT	Original	Abs. Diff.	PCLPT	Original	Abs. Diff.	PCLPT	Original	Abs. Diff.	PCLPT	Original	Abs. Diff.
Lena	82	40	42	135	-20	155	-20	-20	0	1.33	1.33	0
Boats	-2	40	42	0	-20	20	-10	-20	10	1.33	1.33	0
Cameraman	42	40	2	-20	-20	0	-20	-20	0	1.33	1.33	0
Copper sample	40	40	0	-20	-20	0	-20	-20	0	1.33	1.33	0

3.4. Image Correction with Pre-Processed PCLPT

It was shown in the previous section that the PCLPT based image correction technique can accurately compute corrections for individual problems of translation, rotation and scaling but that the technique does not perform well with combinations of these problems. The major reason for this failing is related to the computational process. The PCLPT-based technique considers the shift values only at the pixel level but not at a sub-pixel level (Weihong et al., 2012). Thus, a cumulated error in sub-pixel level can occur during the interpolation process and this can mean the rotation angle and/or scaling factor may not be computed correctly (Foroosh et al., 2002 and Pan et al., 2009).

To overcome this cumulated computational error in PCLPT in the case of the combined problem of image translation, rotation and scaling, an image filtering-based pre-processing stage was introduced (Pham et al., 2012b). Several types of image filters were tested as the pre-processing stage and the performance of each pre-processed-PCLPT technique was calculated. The flow chart of the proposed algorithm is shown in Figure 3.13.

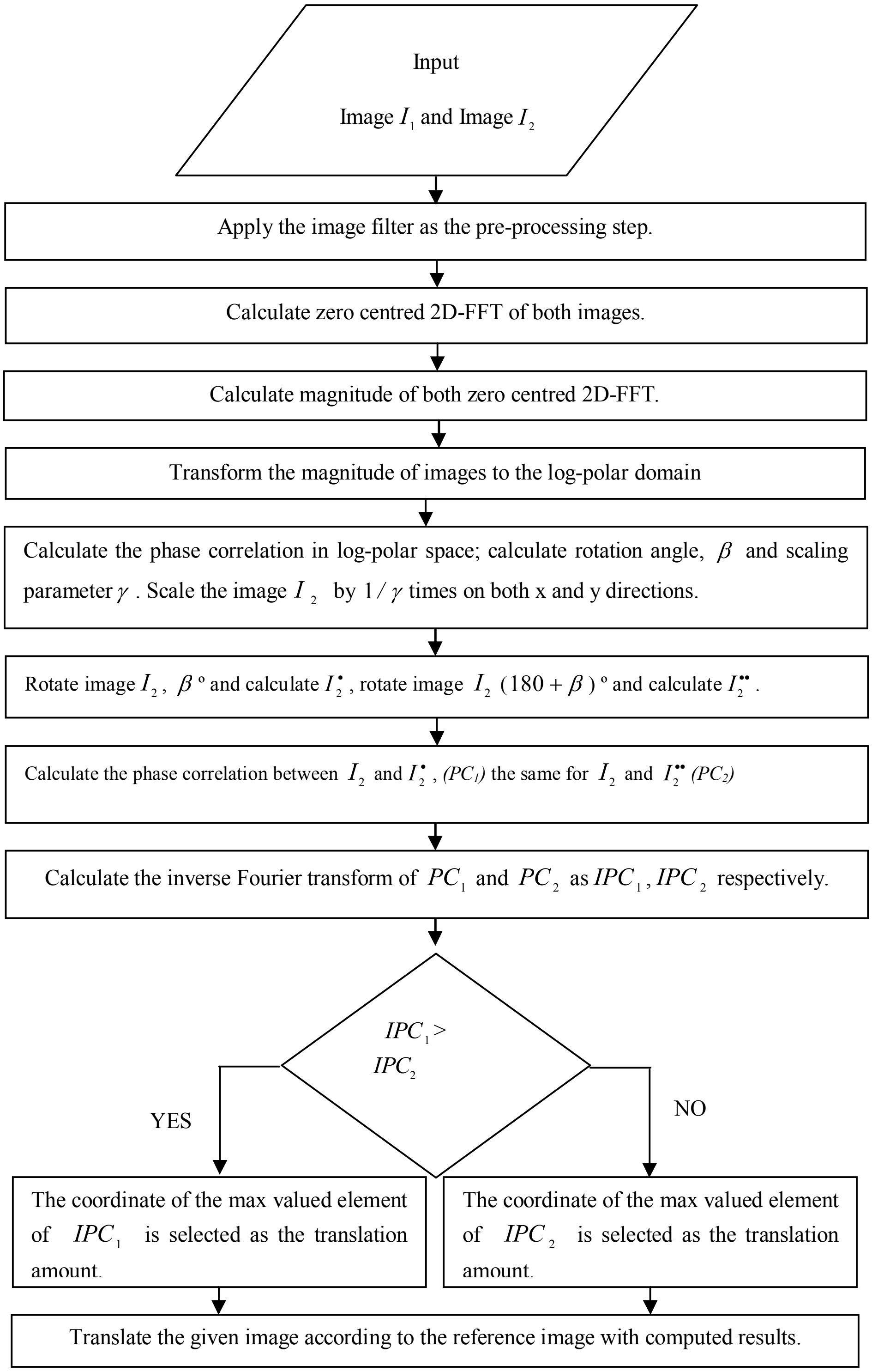


Figure 3.13 The flow chart of the proposed technique.

In this study, the following image filters were tested as the pre-processing stage, the details about each filter are given in Appendix B.

- Gaussian filter (Gonzalez et al., 2004; Semmlow, 2004; (Merchant, 2008),
- Laplacian of the Gaussian filter (Gonzalez et al., 2004),
- Prewitt filter (Gonzalez et al., 2004),
- Sobel operator (Burger and Burge, 2009),
- Median filter (Gonzalez et al., 2004),
- Iterative-based blind image deconvolution (Holmes et al., 1995; Biggs and Andrews, 1997).

The proposed techniques were tested on the four images used previously and given in Figure 3.14 a-d. The images were rotated 20^{o} in an anti-clockwise direction, translated by (-40, 20) pixels and scaled 75% from the original size, as shown in Figure 3.15 a-d.

The amount of rotation, translation and scaling were predicted with pre-processed PCLPT and six different pre-processing techniques have been tested. The predicted images obtained using the six different pre-processing techniques are shown in Figures 3.16-3.21 a-d. The comparative results for rotation, translation and scaling for image of Lena, Boats, Cameraman and Copper sample are given in Table 3.2-3.5, the predicted results of the proposed techniques are given in column headed RPT (Results of the Proposed Technique).

a) b) c) d)

Figure 3.14 The test images a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.15 The images translated by (-40, 20) pixels, rotated 20^o in an anti-clockwise direction and scaled 75% a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.16 Images corrected using Gaussian filter based pre-processed PCLPT technique a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.17 Images corrected using LoG filter based pre-processed PCLPT technique a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.18 Images corrected using Prewitt filter based pre-processed PCLPT technique a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.19 Images corrected using Sobel operator based pre-processed PCLPT technique a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.20 Images corrected using Median filter based pre-processed PCLPT technique a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

a) b) c) d)

Figure 3.21 Images corrected using iterative blind deconvolution based pre-processed PCLPT technique a) Lena, b) Boats, c) Cameraman, and d) Copper sample.

Table 3.2 The results for the rotated-translated and scaled image of Lena.

Test Image: Lena	Translation (pixel)						Rotation (degree)			Scaling		
	X-direction			Y-direction								
	RPT	Original	Abs. Diff.	RPT	Original	Abs. Error	RPT	Original	Abs. Diff.	RPT	Original	Abs. Diff.
Gaussian filter-based PCLPT	5	40	35	87	-20	107	-20	-20	0	1.33	1.33	0
LoG filter-based PCLPT	40	40	0	-12	-20	8	-20	-20	0	1.33	1.33	0
Prewitt filter-based PCLPT	40	40	0	-20	-20	0	-20	-20	0	1.33	1.33	0
Sobel operator-based PCLT	40	40	0	-20	-20	0	-20	-20	0	1.33	1.33	0
Median filter-based PCLPT	41	40	1	-20	-20	0	-20	-20	0	1.33	1.33	0
Iterative Blind deconvolution-based PCLPT	40	40	0	-20	-20	0	-20	-20	0	1.33	1.33	0

Table 3.3 The results for the rotated-translated and scaled image of Boats.

Test Image: Boats	Translation (pixel)						Rotation (degree)			Scaling		
	X-direction			Y-direction								
	RPT	Original	Abs. Diff.	RPT	Original	Abs. Diff.	RPT	Original	Abs. Diff.	RPT	Original	Abs. Diff.
Gaussian filter-based PCLPT	40	40	0	-2	-20	18	-20	-20	7	1.33	1.33	0
LoG filter-based PCLPT	42	40	2	-20	-20	0	-20	-20	0	1.33	1.33	0
Prewitt filter-based PCLPT	40	40	0	-18	-20	2	-20	-20	0	1.33	1.33	0
Sobel operator-based PCLT	41	40	1	-20	-20	0	-20	-20	0	1.33	1.33	0
Median filter-based PCLPT	40	40	0	-20	-20	0	-20	-20	0	1.33	1.33	0
Iterative Blind deconvolution-based PCLPT	40	40	0	-20	-20	0	-20	-20	0	1.33	1.33	0

Table 3.4 The results for the rotated-translated and scaled image of Cameraman.

Test Image: Cameraman	Translation (pixel)						Rotation (degree)			Scaling		
	X-direction			Y-direction								
	RPT	Original	Abs. Diff.	RPT	Original	Abs. Diff.	RPT	Original	Abs. Diff.	RPT	Original	Abs. Diff.
Gaussian filter-based PCLPT	55	40	15	-20	-20	0	-20	-20	0	1.33	1.33	0
LoG filter-based PCLPT	40	40	15	-20	-20	0	-20	-20	0	1.33	1.33	0
Prewitt filter-based PCLPT	50	40	10	-20	-20	0	-20	-20	0	1.33	1.33	0
Sobel operator-based PCLT	40	40	0	-20	-20	0	-20	-20	0	1.33	1.33	0
Median filter-based PCLPT	42	40	2	-13	-20	7	-20	-20	40	1.33	1.33	0
Iterative Blind deconvolution-based PCLPT	40	40	0	-20	-20	0	-20	-20	0	1.33	1.33	0

Table 3.5 The results for the rotated-translated and scaled image of copper sample captured with SEM.

Test Image: Copper sample	Translation (pixel)						Rotation (degree)			Scaling		
	X-direction			Y-direction								
	PT	Original	Abs. Diff.	PT	Original	Abs. Diff.	PT	Original	Abs. Diff.	PT	Original	Abs. Diff.
Gaussian filter-based PCLPT	40	40	0	-20	-20	0	-20	-20	0	1.33	1.33	0
LoG filter-based PCLPT	40	40	0	-80	-20	60	-21	-20	1	1.31	1.33	0.02
Prewitt filter-based PCLPT	83	40	43	-80	-20	60	-46	-20	16	2.55	1.33	1.22
Sobel operator-based PCLT	40	40	0	-71	-20	51	-38	-20	18	2.55	1.33	1.22
Median filter-based PCLPT	40	40	0	-20	-20	0	-20	-20	0	1.33	1.33	0
Iterative Blind deconvolution-based PCLPT	40	40	0	-20	-20	0	-20	-20	0	1.33	1.33	0

According to the absolute difference in Tables 3.2 to 3.5, the addition of a Gaussian and LoG filters as pre-processors to the PCLPT improved performance of the technique but did not provide the original amount of rotation, translation and scaling factor. This is due to certain characteristics of the Gaussian and LoG filters; both filters increase the blurriness level of image which decreases the quality of edges in the image.

The Prewitt filter performed well as a pre-processor for the images of Lena and Boats but not so well on the images of the Copper sample captured with SEM and Cameraman. The Sobel operator performed very well on the images of Lena, Boats and Cameraman but not so well on the image of the Copper sample. The Prewitt filter and Sobel operator increases the edge quality of images. This is why both techniques have a problem with images which contain pixels close together having the same value elements. The Sobel operator utilises the second derivative of images, which decreases its sensitivity as a pre-processor compared to that of the Prewitt filter.

The Median filter and iterative blind deconvolution performed very well as pre-processors on all four images according to the absolute difference between predicted results and original results.

In the next section all six pre-processors based on the PCLPT are combined with the DFAF method to produce an image correction technique which demonstrates the performance of each pre-processer for the images of copper sample captured in different depth levels.

3.5. Pre-Processed PCLPT-Based DFAF Technique

Image distortion is a critical problem with SEMs. Distortion can occur as a result of internal and external factors such as beam motion, interaction between electron and surface of sample, artefacts on the sample, lens aberrations, environmental noise, etc. (Snella, 2010). The factors causing distortion may generate translation and rotation of the image, so SEM images often need to be corrected. The pre-processed PCLT is proposed as a suitable image correction technique. The sharpness function is then applied to different levels of the focus images of the SEM to correlate depth with sharpness level. The block diagram of the proposed technique is shown in Figure 3.22. In this study, the following sharpness measure functions were employed:

- Tenengrad function,
- Brenner function,
- Squared Gradient function,
- Range function,
- Entropy function,
- Laplacian function, and
- Variance function.

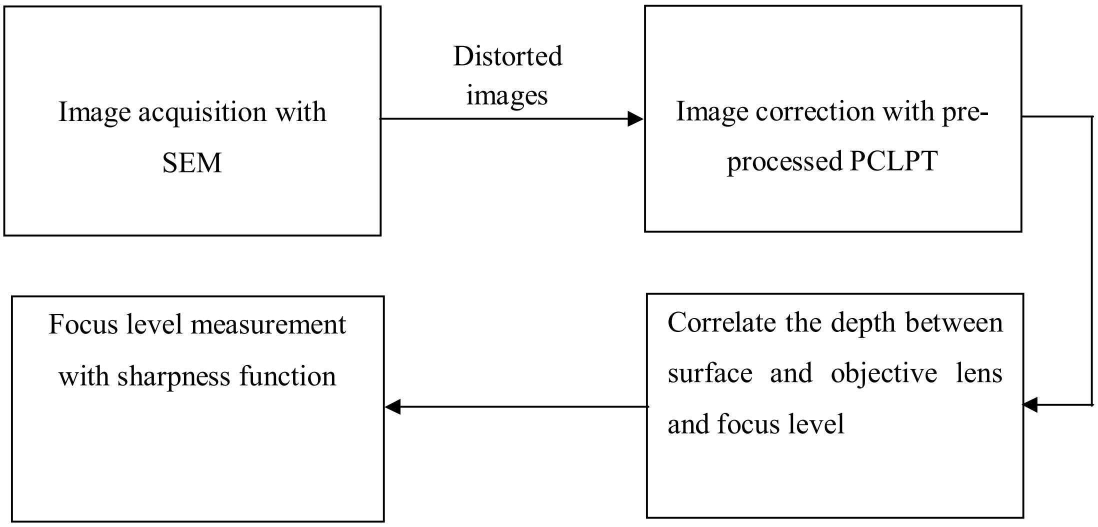


Figure 3.22 Block diagram of the proposed technique.

3.5.1. Experimental Results for the Pre-Processed PCLPT-Based DFAF Technique

In this section the proposed technique was performed on different levels of focused images of a copper sample captured with a SEM. The experiments were carried out with a 1500 XB model SEM system. The SEM parameters were set to: magnification = 135kX, working distance = 5 mm, aperture size = 30 μm, EHT = 30 kV, current mode = high current, and imaging detector = secondary electron detector. The experiments started by calibrating the system for a focused image as shown in Figure 3.14d, then the position of the sample was moved up 0.2 mm and a new image captured. The process was repeated 10 times, each time moving the sample up 0.2 mm. The next step was to replace the sample back in the focused image position. The sample was then moved down 10 steps of 0.2 mm, and at each level, an image was captured. All test images are given in Appendices C.1 a-j and C.2 a-j.

The next step was to search for any distortion or registration problem such as rotation, translation or scaling in the images. Image correction was accomplished with the GIMP 2.6 open source programme for the Gaussian filter, LoG Filter, Prewitt filter, Sobel operator, Median filter and iterative-based blind image deconvolution as pre-processing stages for the PCLPT technique. Then the sharpness measure functions were applied to compute the sharpness levels of each image. Finally, the relationship between sharpness level and distance was defined, which is related to the distance from the focused plane and normalised sharpness function value. The results of the pre-processed PCLPT techniques are given in Table 3.6-3.11 and the error in each technique was determined in terms of absolute difference between the original and predicted results. In the tables, DTFI denotes the distance between the specimen and the focused image plane in SEM.

Table 3.6 Results of the Gaussian filter-based pre-processed PCLPT technique on different levels of the focused SEM images.

Images	DTFI	Translation (pixel)				Rot. (degree)	Rot. Error (degree)	Scale	Scale Error
		Tx	Tx Error	Ty	Ty Error				
Image1	2	0	0	0	0	0	0	1	0
Image2	1.8	0	0	0	0	0	0	1	0
Image3	1.6	0	0	0	0	0	0	1	0
Image4	1.4	0	0	0	0	0	0	1	0
Image5	1.2	0	0	0	0	0	0	1	0
Image6	1	0	0	0	0	0	0	1	0
Image7	0.8	0	0	0	0	0	0	1	0
Image8	0.6	0	0	0	0	0	0	1	0
Image9	0.4	0	0	0	0	0	0	1	0
Image10	0.2	0	0	0	0	0	0	1	0
Image11	0	0	0	0	0	0	0	1	0
Image12	-0.2	0	0	0	0	0	0	1	0
Image13	-0.4	0	0	0	0	0	0	1	0
Image14	-0.6	0	0	0	0	0	0	1	0
Image15	-0.8	0	0	0	0	0	0	1	0
Image16	-1	0	0	0	0	0	0	1	0
Image17	-1.2	0	0	0	0	0	0	1	0
Image18	-1.4	0	0	0	0	0	0	1	0
Image19	-1.6	0	0	0	0	0	0	1	0
Image20	-1.8	0	0	0	0	0	0	1	0
Image21	-2	0	0	0	0	0	0	1	0

Table 3.7 Results of the Laplacian of Gaussian-based pre-processed PCLPT technique on different levels of the focused SEM images.

Images	DTFI	Translation (pixel)				Rot. (degree)	Rot. Error (degree)	Scale	Scale Error
		Tx	Tx Error	Ty	Ty Error				
Image1	2	-48	48	9	9	38	38	2.55	1.55
Image2	1.8	-8	8	-3	3	-17	17	1	0
Image3	1.6	-10	10	15	15	-24	24	1	0
Image4	1.4	-10	10	12	12	-24	24	1	0
Image5	1.2	-11	11	-11	11	58	58	1	0
Image6	1	9	9	10	10	1	1	1	0
Image7	0.8	-2	2	-12	12	-34	34	1	0
Image8	0.6	13	13	-7	7	-30	30	1	0
Image9	0.4	4	4	7	7	28	28	1	0
Image10	0.2	-7	7	8	8	32	32	1	0
Image11	0	0	0	0	0	0	0	1	0
Image12	-0.2	-9	9	-9	9	-15	15	1	0
Image13	-0.4	5	5	-3	3	22	22	1	0
Image14	-0.6	8	8	7	7	-17	17	1	0
Image15	-0.8	8	8	8	8	-1	1	1	0
Image16	-1	1	1	2	2	-31	31	1	0
Image17	-1.2	16	16	-7	7	28	28	1	0
Image18	-1.4	59	59	-69	69	45	45	2.55	1.55
Image19	-1.6	23	23	34	34	34	34	2.55	1.55
Image20	-1.8	-8	8	-8	8	-1	1	1	0
Image21	-2	6	6	8	8	30	30	2.55	1.55

Table 3.8 Results of the Prewitt filter-based pre-processed PCLPT technique on different levels of the focused SEM images.

Images	DTFI	Translation (pixel)				Rot. (degree)	Rot. Error (degree)	Scale	Scale Error
		Tx	Tx Error	Ty	Ty Error				
Image1	2	7	7	4	4	-24	24	2.55	1.55
Image2	1.8	7	7	-10	10	14	14	1	0
Image3	1.6	-11	11	-5	5	6	6	1	0
Image4	1.4	3	3	5	5	-10	10	1	0
Image5	1.2	-9	9	12	12	-1	1	1	0
Image6	1	-10	10	-10	10	-7	-7	1	0
Image7	0.8	9	9	-2	2	21	21	1	0
Image8	0.6	2	2	2	2	20	20	1	0
Image9	0.4	-6	6	10	10	-1	-1	1	0
Image10	0.2	-7	7	4	4	0	0	1	0
Image11	0	0	0	0	0	0	0	1	0
Image12	-0.2	2	2	8	8	20	20	1	0
Image13	-0.4	5	5	4	4	0	0	1	0
Image14	-0.6	16	16	36	36	-28	28	1	0
Image15	-0.8	7	7	2	2	20	20	1	0
Image16	-1	3	3	0	0	-6	6	1	0
Image17	-1.2	6	6	9	9	0	0	1	0
Image18	-1.4	4	4	8	8	0	0	1	0
Image19	-1.6	-14	14	1	1	1	1	1	0
Image20	-1.8	-44	44	1	1	21	21	1	0
Image21	-2	9	9	1	1	1	1	1	0

Table 3.9 Results of the Sobel operator-based pre-processed PCLPT technique on different levels of the focused SEM images.

Images	DTFI	Translation (pixel)				Rot. (degree)	Rot. Error (degree)	Scale	Scale Error
		Tx	Tx Error	Ty	Ty Error				
Image1	2	26	26	-6	6	-24	24	2.55	1.55
Image2	1.8	9	9	-7	7	22	22	1	0
Image3	1.6	8	8	10	10	20	20	1	0
Image4	1.4	-13	13	-10	10	-10	10	1	0
Image5	1.2	17	17	11	11	13	13	2.55	1.55
Image6	1	-3	3	14	14	39	39	1	0
Image7	0.8	3	3	10	10	13	13	1	0
Image8	0.6	-6	6	-1	1	10	10	1	0
Image9	0.4	7	7	-11	11	14	14	1	0
Image10	0.2	-18	18	-2	2	35	35	2.55	1.55
Image11	0	0	0	0	0	0	0	1	0
Image12	-0.2	3	3	-10	10	20	20	1	0
Image13	-0.4	-2	2	12	12	0	0	1	0
Image14	-0.6	16	16	-7	7	-28	28	1	0
Image15	-0.8	5	5	-13	13	-10	10	1	0
Image16	-1	0	0	78	78	39	39	1	0
Image17	-1.2	13	13	28	28	15	15	1	0
Image18	-1.4	1	1	9	9	0	0	1	0
Image19	-1.6	26	26	37	37	25	25	1	0
Image20	-1.8	-3	3	1	1	16	16	1	0
Image21	-2	-11	11	10	10	1	1	1	0

Table 3.10 Results of the Median filter-based pre-processed PCLPT technique on different levels of the focused SEM images.

Images	DTFI	Translation (pixel)				Rot. (degree)	Rot. Error (degree)	Scale	Scale Error
		Tx	Tx Error	Ty	Ty Error				
Image1	2	0	0	0	0	0	0	1	0
Image2	1.8	0	0	0	0	0	0	1	0
Image3	1.6	0	0	0	0	0	0	1	0
Image4	1.4	0	0	0	0	0	0	1	0
Image5	1.2	0	0	0	0	0	0	1	0
Image6	1	0	0	0	0	0	0	1	0
Image7	0.8	0	0	0	0	0	0	1	0
Image8	0.6	0	0	0	0	0	0	1	0
Image9	0.4	0	0	0	0	0	0	1	0
Image10	0.2	0	0	0	0	0	0	1	0
Image11	0	0	0	0	0	0	0	1	0
Image12	-0.2	0	0	0	0	0	0	1	0
Image13	-0.4	0	0	0	0	0	0	1	0
Image14	-0.6	0	0	0	0	0	0	1	0
Image15	-0.8	0	0	0	0	0	0	1	0
Image16	-1	0	0	0	0	0	0	1	0
Image17	-1.2	0	0	0	0	0	0	1	0
Image18	-1.4	0	0	0	0	0	0	1	0
Image19	-1.6	0	0	0	0	0	0	1	0
Image20	-1.8	0	0	0	0	0	0	1	0
Image21	-2	0	0	0	0	0	0	1	0

Table 3.11 Results of the iterative blind deconvolution-based pre-processed PCLPT technique on different levels of the focused SEM images.

Images	DTFI	Translation (pixel)				Rot. (degree)	Rot. Error (degree)	Scale	Scale Error
		Tx	Tx Error	Ty	Ty Error				
Image1	2	0	0	0	0	0	0	1	0
Image2	1.8	0	0	0	0	0	0	1	0
Image3	1.6	0	0	0	0	0	0	1	0
Image4	1.4	0	0	0	0	0	0	1	0
Image5	1.2	0	0	0	0	0	0	1	0
Image6	1	0	0	0	0	0	0	1	0
Image7	0.8	0	0	0	0	0	0	1	0
Image8	0.6	0	0	0	0	0	0	1	0
Image9	0.4	0	0	0	0	0	0	1	0
Image10	0.2	0	0	0	0	0	0	1	0
Image11	0	0	0	0	0	0	0	1	0
Image12	-0.2	0	0	0	0	0	0	1	0
Image13	-0.4	0	0	0	0	0	0	1	0
Image14	-0.6	0	0	0	0	0	0	1	0
Image15	-0.8	0	0	0	0	0	0	1	0
Image16	-1	0	0	0	0	0	0	1	0
Image17	-1.2	0	0	0	0	0	0	1	0
Image18	-1.4	0	0	0	0	0	0	1	0
Image19	-1.6	0	0	0	0	0	0	1	0
Image20	-1.8	0	0	0	0	0	0	1	0
Image21	-2	0	0	0	0	0	0	1	0

According to the above results, the Gaussian filter, Median filter and iterative blind deconvolution-based PCLPT techniques determined the image rotation-translation and scaling results 100% correctly for the different levels of focused SEM images. However, the LoG filter, Prewitt filter and Sobel operator-based PCLPT technique were unable to determine the correct results. Thus it is expected that the depth calculation results of the Gaussian filter, Median filter and iterative blind deconvolution-based PCLPT techniques will be more accurate than the other techniques.

The next step is to measure the sharpness value of the corrected images. In this study, the Tenengrad function, the Brenner function, the Squared Gradient function, the Range function, the Entropy function, the Laplacian function and the Variance function were used for sharpness measurement. The sharpness results for each function were normalised between 0-1. Finally the computed sharpness values were correlated with the distance between the specimen and the focused image plane. The results for each of the possible combinations of techniques are given in Figures 3.23-3.28.

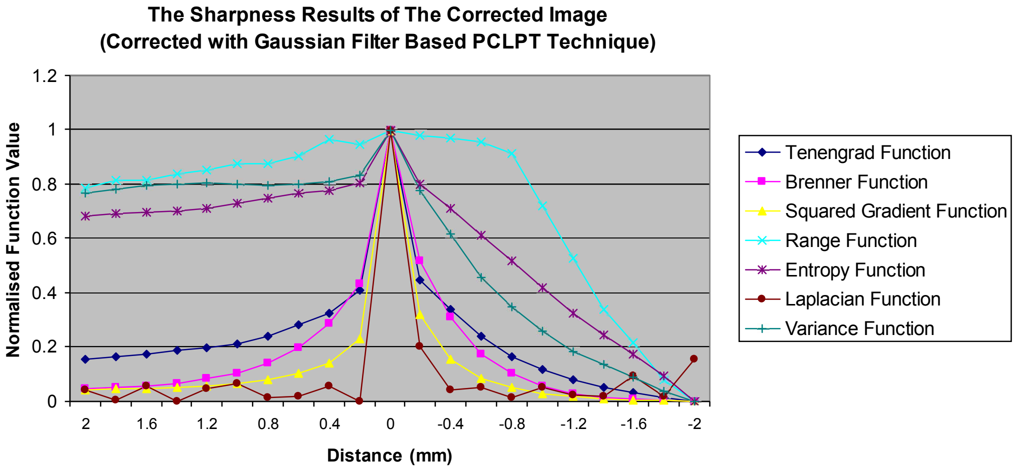


Figure 3.23 The relationship between distance and normalised function value for the corrected SEM image (carried out with Gaussian filter-based PCLPT technique).

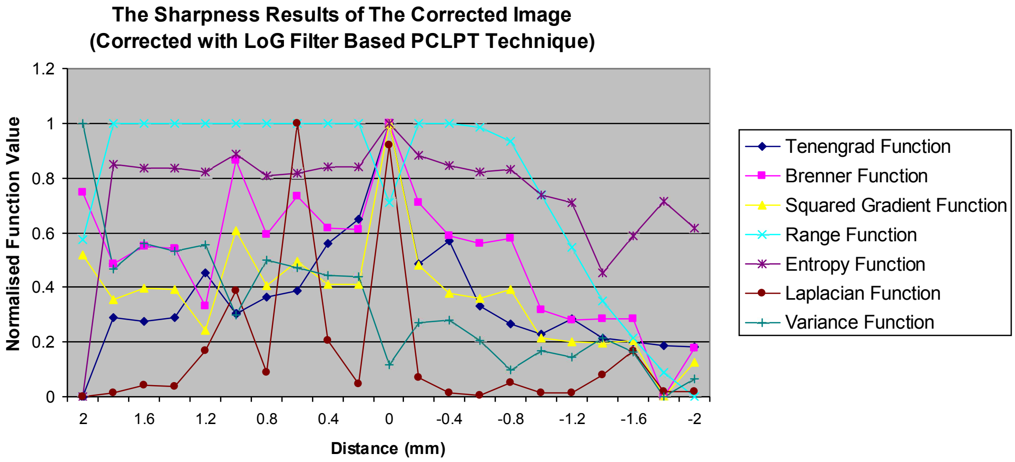


Figure 3.24 The relationship between distance and normalised function value for the corrected SEM image (carried out with LoG filter-based PCLPT technique).

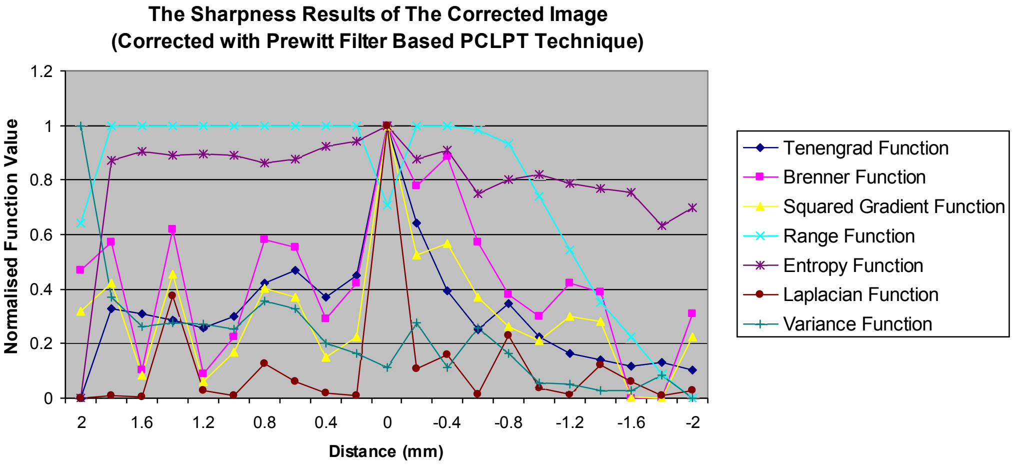


Figure 3.25 The relationship between distance and normalised function value for the corrected SEM image (carried out with Prewitt filter-based PCLPT technique).

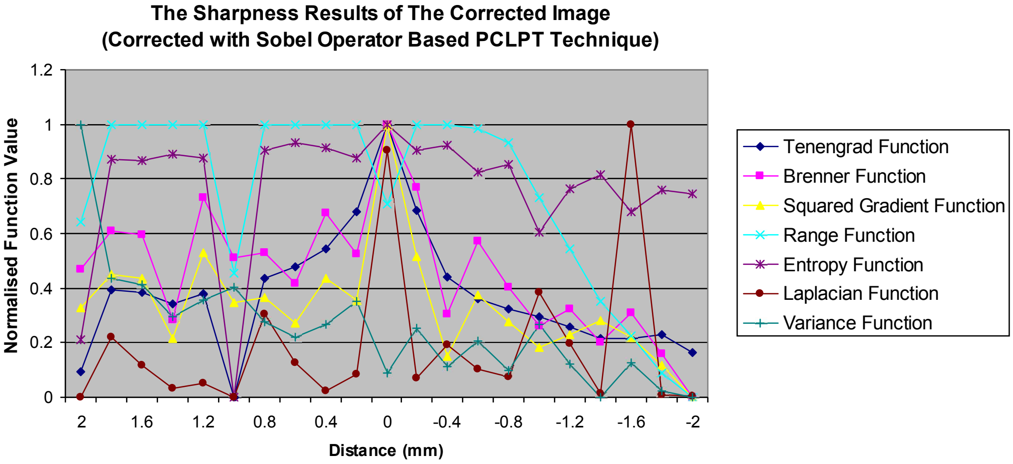


Figure 3.26 The relationship between distance and normalised function value for the corrected SEM image (carried out with Sobel operator-based PCLPT technique).

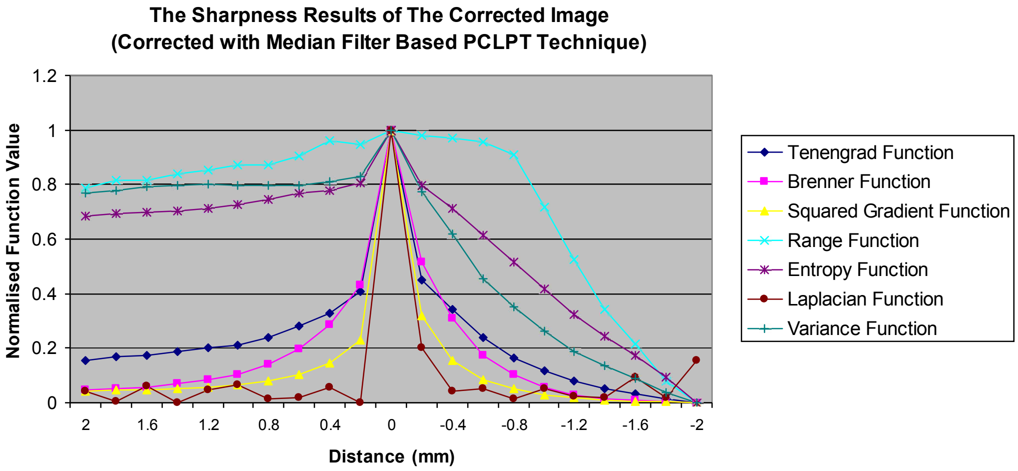


Figure 3.27 The relationship between distance and normalised function value for the corrected SEM image (carried out with Median filter-based PCLPT technique).

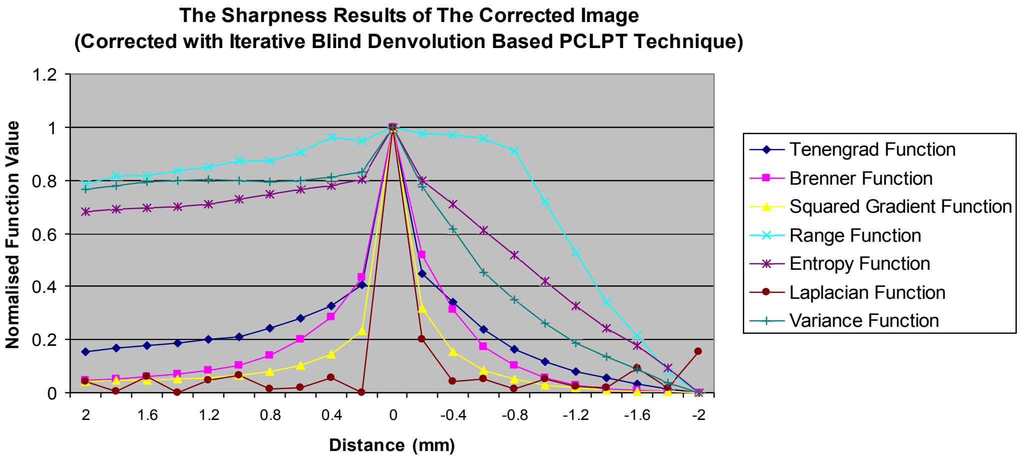


Figure 3.28 The relationship between distance and normalised function value for the corrected SEM image (carried out with iterative blind deconvolution-based PCLPT technique).

According to results presented in Figures 3.23 – 3.28 the Gaussian filter, Median filter and iterative blind deconvolution when used as pre-processors to the PCLPT perform well as an image correction process for DFAF technique. It is also clear that the Squared Gradient, Tenengrad, and Brenner functions provide the sharpest image position when used with the DFAF technique. The remainder of the sharpness functions did not show a clear relationship between sharpness and the distance moved by the sample. A robust sharpness function should define the relationship very clearly and it would be expected that the distribution of sharpness function would be symmetric with respect to the distance moved by the sample, either up or down. For example, the function value of the point 0.2 is expected to be equal to that at the point -0.2 relative to the focused image point 0. In the same way, any given point x and its symmetric value (-x) (reflection symmetry with respect to the horizontal direction) should have the same sharpness value. The matching error between symmetric distances can be used as an error metric to evaluate the performance of the given sharpness function. In this study, the sum of the absolute matching differences (SAMD) of symmetric points was utilised for the given sharpness functions and the following results were found for each function: Tenengrad function: 0.9715, Brenner function: 0.4701: Laplacian function: 0.4747, Squared Gradient: 0.3796, Range function: 3.2222, Entropy function: 3.4148 and Variance function: 5.0718. According to the SAMD results, the performance of the Squared Gradient function is best.

3.6. Summary

In this chapter, the focus was on the image correction process for distorted SEM images. The image correction process was developed for SEM images using a pre-processed PCLPT technique after which the DFAF technique was used on the corrected images to define the relationship between depth and sharpness value.

The pre-processing step was introduced utilising image filters to increase the efficiency of the PCLPT technique. In this study, six image filters were employed and the proposed technique was tested on four benchmark images: Lena, Boats, Cameraman and image of a copper sample captured with a SEM. Finally, the DFAF technique was used on the corrected images.

Seven sharpness measurement functions were combined with the DFAF technique. The performance of each technique was measured using a SAMD metric. According to the experiments, the Squared Gradient function performed well in determining the relationship between distance and sharpness function value but still sensitive for the SEM image and a better technique is needed to illustrate the distance changes. A Bees Algorithm based optimised BID technique is proposed and two different approaches will be described in Chapter 4. After refining and developing the Bees Algorithm an improved version will be used with the BID technique and this will be described in Chapter 5.

Chapter 4

IMPROVEMENTS TO THE BEES ALGORITHM

4.1. Preliminaries

This chapter proposes two different improvements to the Bees Algorithm (BA), both improvements are in the neighbourhood search and the improved BAs were tested on continuous type benchmark functions and an optimisation problem.

The first improvement is based on the adaptive neighbourhood size change and site abandonment strategy. The proposed algorithm increases the speed of the searching process and avoids getting stuck in local minima by changing the neighbourhood size dynamically.

The second improvement is to determine the direction of the most promising sites by computing the slope angle of the best sites. If the slope angle is lower than a given threshold value, then a Hill Climbing Algorithm-based neighbourhood search process is utilised to find the most promising sites.

4.2. The Bees Algorithm

The BA has both local and global search capability utilising exploitation and exploration strategies respectively. The BA uses the set of parameters given in Table 4.1. The pseudo code of the algorithm is given in Figure 4.1 and the flow chart of the algorithm is given in Figure 4.2.

Table 4.1 Basic parameters of the Bees Algorithm.

Parameter	Symbols
Number of scout bees in the selected patches	*n*
Number of best patches in the selected patches	*m*
Number of elite patches in the selected best patches	*e*
Number of recruited bees in the elite patches	*nep*
Number of recruited bees in the non-elite best patches	*nsp*
The size of neighbourhood for each patch	*ngh*
Number of iterations	*iter*
Difference between value of the first and last iterations	*diff*

The Algorithm starts with sending *n* scout bees randomly to selected sites (see Figure 4.3). The fitness values of each site are evaluated and sorted from the highest to the lowest (a maximisation problem). The local search step of the algorithm covers the best locations (sites) which are the *m* fittest locations. The *m* best sites are also classified into two sub-groups; *elite* and *non-elite best sites*, as given in Figure 4.4. The number of *elite* sites is set as "*e*" and number of the *non-*

elite best sites is “*m-e*”. The local search process starts with recruiting forager bees in the neighbourhood of the best sites. The neighbourhood size is set to “*ngh*”. The number of recruited bees in the neighbourhood of each elite site is set to “*nep*” and the number of recruited bees in the neighbourhood of the non-elite best sites is set to “*nsp*”, as given in Figure 4.5. The global search process is a random search process in the *n-m* “*non-best” s*ites, as given in Figure 4.6. Finally, the overall locations are sorted according to their fitness value and the process runs until the global optimum is found.

```
Generate the initial population size as $n$, set the best patch size as $m$, set the elite patch size as $e$, set the number of forager bees recruited to the of elite sites as $nep$, set the number of forager bees around the non-elite best patches as $nsp$, set the neighbourhood size as $ngh$, set the maximum iteration number as $MaxIter$, and set the error limit as $Error$.

i= 0

Generate initial population.
Evaluate Fitness Value of initial population.
Sort the initial population based on the fitness result.
While $i \leq MaxIter$ or $FitnessValue_i - FitnessValue_{i-1} \leq Error$
    i= i+l;
    Select the elite patches and non-elite best patches for neighbourhood search.
    Recruit the forager bees to the elite patches and non-elite best patches.
    Evaluate the fitness value of each patch.
    Sort the results based on their fitness.
    Allocate the rest of the bees for global search to the non-best locations.
    Evaluate the fitness value of non-best patches.
    Sort the overall results based on their fitness.
    Run the algorithm until termination criteria met.
End
```

Figure 4.1 Pseudo-code of the basic Bees Algorithm.

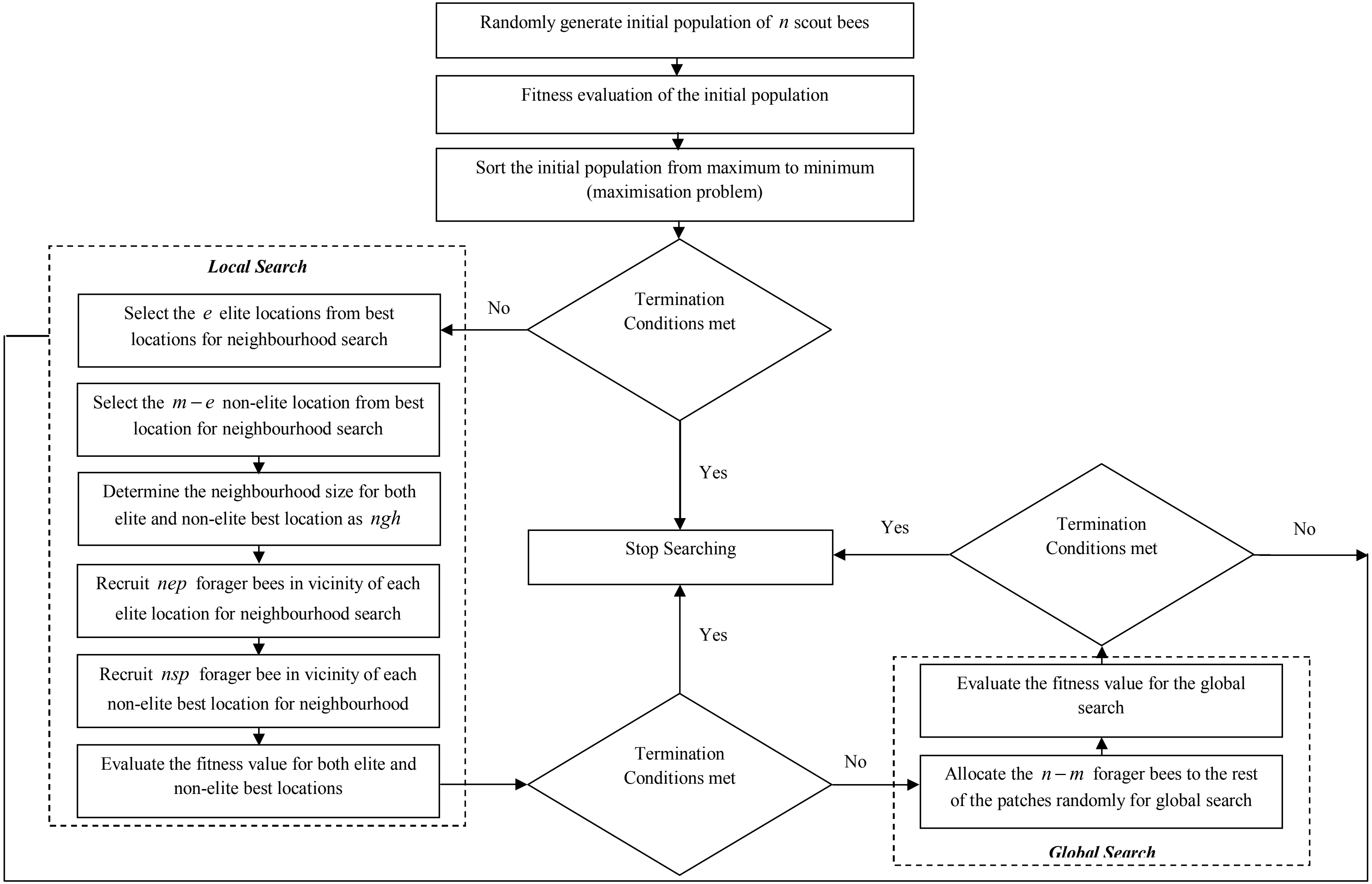


Figure 4.2 Flowchart of the basic Bees Algorithm.

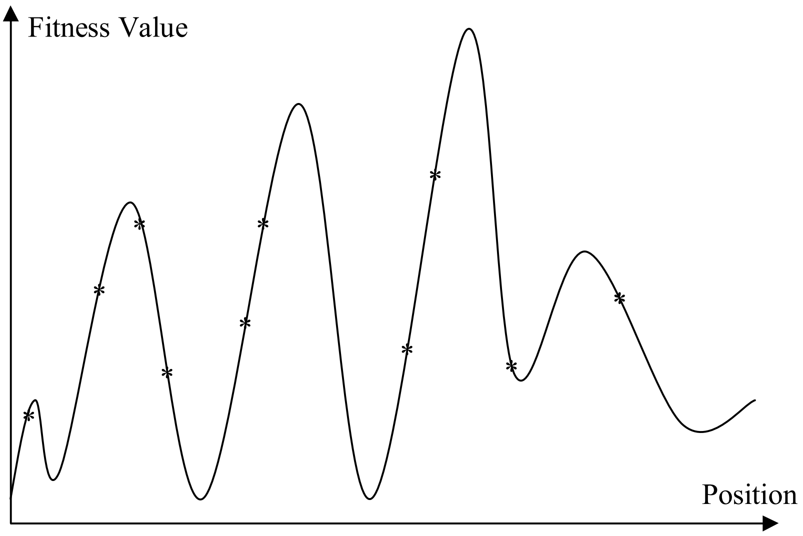


Figure 4.3 The initially selected *n* patches and their evaluated fitness values

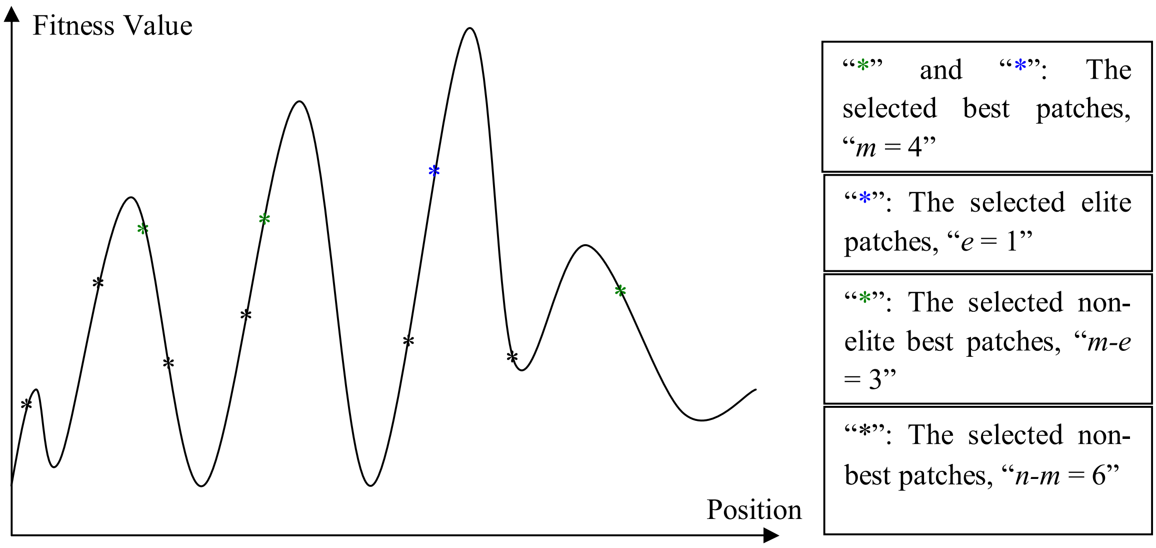


Figure 4.4 Selection of elite and non-elite best patches.

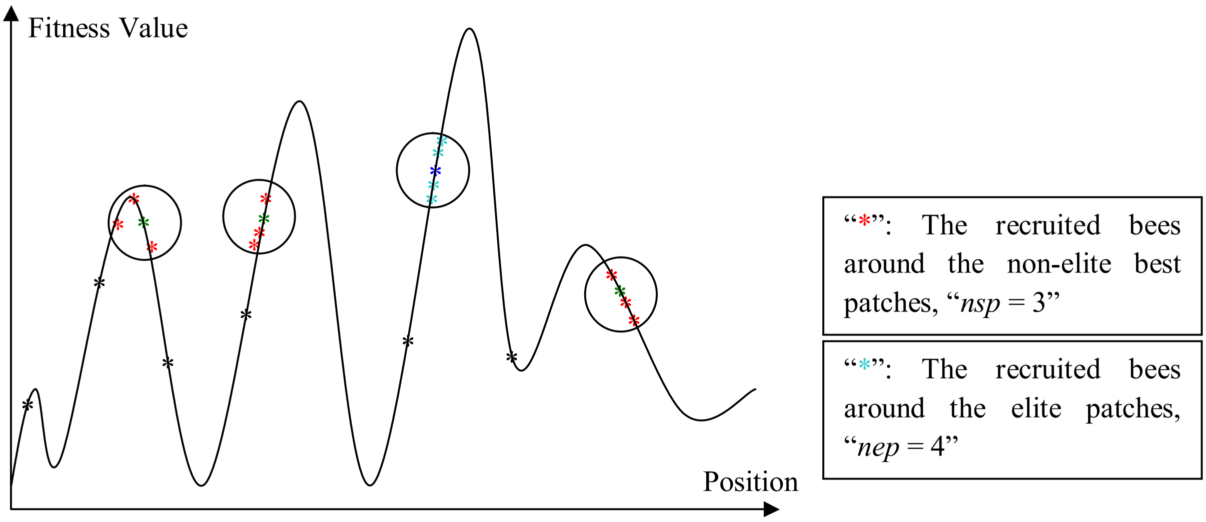


Figure 4.5 Recruitment of forager bees to the elite and non-elite best locations.

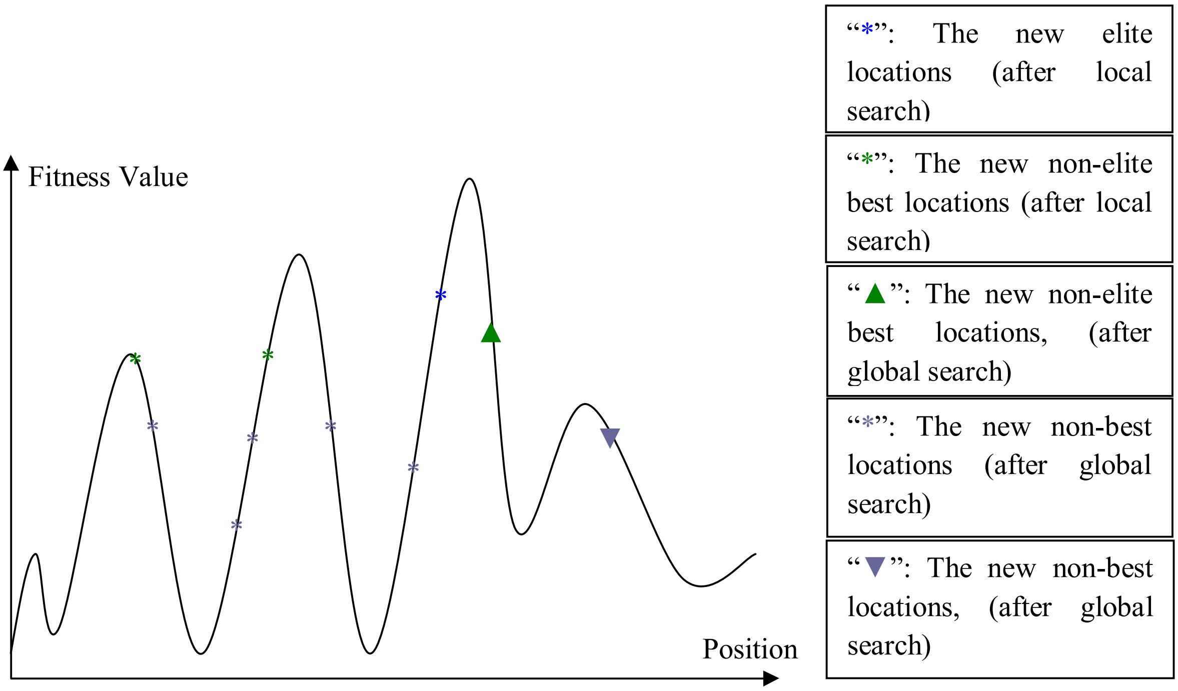


Figure 4.6 Results from basic BA after local and global search.

4.3. BA Improved by Adaptive Change in Neighbourhood Size and Site Abandonment Strategy

This section describes the proposed improvements to the BA by applying adaptive change to the neighbourhood size and site abandonment approach simultaneously. Combined neighbourhood size change and site abandonment (NSSA) strategy has been attempted on the BA by Koc (2010) who found that the convergence rate of a NSSA-based BA can be slow when the promising locations are far from the current best sites. Here an adaptive neighbourhood size change and site abandonment (ANSSA) strategy is proposed which will avoid local minima by changing the neighbourhood size adaptively. The ANSSA-based BA possesses both shrinking and enhancement strategies according to the fitness evaluation. The initial move is to implement the shrinking strategy. The strategy works on a best site after a certain number of repetitions. The strategy works until the repetition stops. If, in spite of the shrinking strategy, the number of repetitions still increases for a certain number of iterations, then an enhancement strategy is utilised. Finally, if the number of repetitions still increases for a number of iterations after the use of the enhancement strategy, then that site is abandoned and a new site will be generated.

Koc (2010) utilised the following parameter for shrinking the neighbourhood size and site abandonment strategy: neighbourhood size $= ngh$, the shrinking constant

$=sc$, the abandoned sites $=aband\_site$. In this study four more parameters are introduced. The first is the number of repetitions for each site, denoted as $keep\_point$.The *keep_point* records the number of repetitions for all the repetitive results for the best sites. The second parameter is called the "*Repetition Number for the Shrinking*", denoted as rep_nshr ; the number of shrinking is the number of repetitions for starting the shrinking strategy, as given in Equations 4.1 and 4.2.

The third parameter is called "*Repetition Number for the Enhancement*", denoted rep_nenh . This parameter defines the number of repetitions until the end of the shrinking process, and the beginning of the enhancement process as shown in Equations 4.1 and 4.3. The enhancement process works until the number of the repetitions is equal to the rep_naban , which denotes the "*Repetition Number for Abandonment Process*". So a non-productive site is abandoned and it is stored in $aband\_site$ list. At the end of the searching process, if there is no better solution than the abandoned site, it will be the final solution. The pseudo code of the proposed algorithm is given in Figure 4.7.

$$new\_ngh = \begin{cases} keep\_point \le rep\_nshr & ngh \\ rep\_nshr < keep\_point \le rep\_nenh & R1 \\ rep\_nenh < keep\_point \le re\_naban & R2 \\ rep\_naban < keep\_point & ngh \end{cases} \tag{4.1}$$

$$R1 = ngh - (ngh * \frac{100 - (keep\_point - rep\_nshr)}{100} * sc) \tag{4.2}$$

$$R2 = ngh + (ngh * \frac{100 - (keep\_point - rep\_nenh)}{100} * sc) \tag{4.3}$$

```
Generate the initial population size as n, set the best patch size as m, set the elite patch size as e, set the number of forager bees around elite sites as nep, set the number of forager bees around non-elite best patches as nsp, set the shrinking constant as sc, set the shrinking start number as rep_nshr, set the enhancement start point as rep_nenh, set the abandoning start number as rep_naban, set the neighbourhood size as ngh, set the maximum iteration number as Max_Iter, and set the error limit as Error.

$keep\_point = Zeros(1,m)$;

$aband\_site = Zeros(iter,m)$;

i= 0.Generate initial population; Evaluate Fitness Value of initial population;

Sort the initial population based on the fitness result.

While $i \le MaxIter$ or $FitnessValue_i - FitnessValue_{i-1} \le Error$

        i= i+1;

        Select the elite patches and non-elite best patches for neighbourhood search.

        Recruit the forager bees to the elite patches and non-elite best patches.

        Evaluate the fitness value of each patch; Sort the results based on their fitness.

        $For \quad k = 1:m$

        $If \quad keep\_point(i,k) > rep\_nshr \; and \; keep\_point(i,k) \le rep\_nenh$ then.

        $new\_ngh(i,k) = ngh - (ngh * sc * \frac{100-(keep\_point(i,k) - rep\_nshr)}{100})$.

        $End$

        $If \; keep\_point(i,k) > rep\_nenh \; and \; keep\_point(i,k) \le rep\_naban$

        $new\_ngh(i,k) = ngh + (ngh * sc * \frac{100-(keep\_point(i,k) - rep\_naban)}{100})$

        $End$

        $Else \quad If \quad keep\_point > naban$

        $aband\_site(i+1,k) = aban\_site(i,k) + 1$;

        $Else \; new\_ngh(i,k) = ngh$; $End$

        Allocate the rest of the bees for global search to the non-best locations; evaluate the fitness value of non-best patches; Sort the fitness values and positions; Run the algorithm until termination criteria are met.

        End
```

Figure 4.7 Pseudo code of improved BA with ANSSA strategy.

4.3.1. Experimental Results for improved BA with ANSSA Strategy

The basic BA has been tested on continuous benchmark functions by (Ghanbarzadeh, 2007; Sholedolu, 2009; Pham and Castellani, 2009; Koc, 2010). In this section, the improved BA was tested on both a continuous type optimisation problem and the benchmark functions. The results were compared with the basic BA.

The selected continuous type optimisation problem was an inverted and shifted Himmelblau function (see Equation 4.4). This function is a multimodal 2D function and the figure representing the function is shown as Figure 4.8. The Himmelblau function is a set of quartic form functions and it is not easy to find the global optimum with analytical approaches. In addition, the inverted and shifted form of function becomes more complicated, therefore it is worth to utilise the modified form of the function in optimisation algorithms to evaluate the performance of the proposed algorithms.

$$F(X_1,X_2)=\frac{1}{0.1-(X_1^2+X_2-11)^2-(X_1+X_2^2-7)^2} \tag{4.4}$$

where $-6 \le X_1, X_2 \le 6$

The global maximum for Himmelblau function:

$F(X_1,X_2)=10,\ [X_1,X_2]=[3,2]$

The parameters of the algorithm can be seen in Table 4.2.

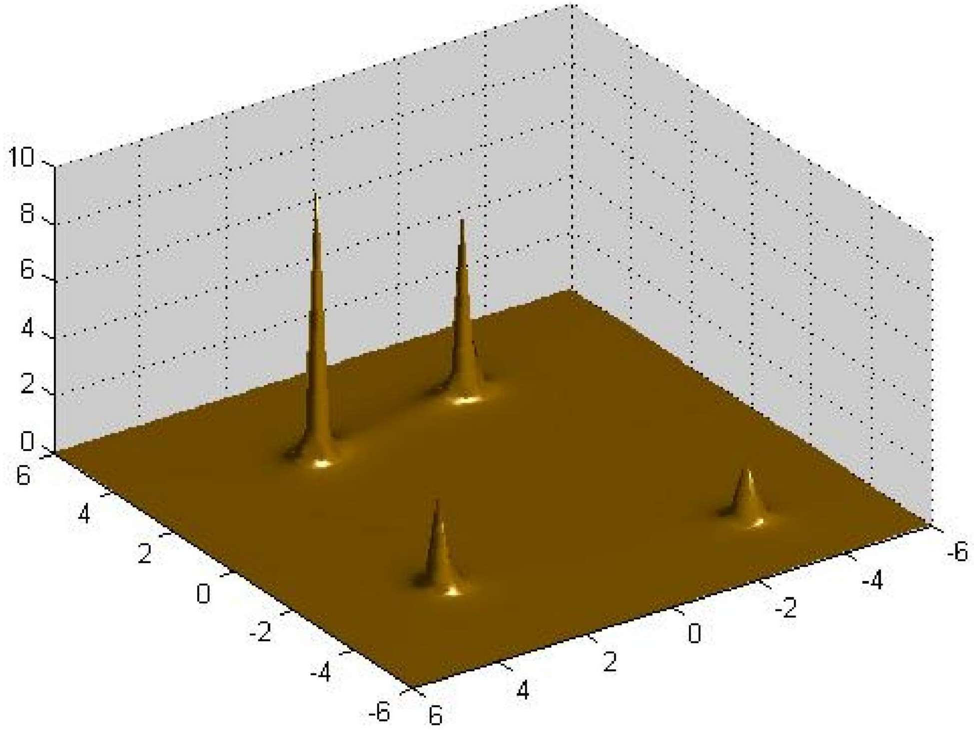


Figure 4.8 Himmelblau function.

Table 4.2 Selected parameters for the Bees Algorithm.

Parameters	Value
Number of Scout Bees in the Selected Patches (n)	50
Number of Best Patches in the Selected Patches (m)	15
Number of Elite Patches in the Selected Best Patches (e)	3
Number of Recruited Bees in the Elite Patches (nep)	12
Number of Recruited Bees in the Non-Elite Best Patches (nsp)	8
The Size of neighbourhood for Each Patches (ngh)	1
Number of Iterations ($iter$)	5000
Difference between the First Iteration Value and the Last Iteration ($diff$)	0.001
Shrinking Constant (sc)	2
Number of Repetitions for Shrinking Process (rep_nshr)	10
Number of Repetitions for Enhancement Process (rep_nenh)	25
Number of Repetitions for Site Abandonment (rep_naban)	100

The comparative results of the basic BA and the improved BA are given in Figure 4.9. Both algorithms were run 100 times and the results presented are for the mean fitness evaluation in iterations. According to the computed results for the Himmelblau function, the improved BA performed better than the basic BA. The mean value and standard deviation of the enhanced BA and the basic BA are given in Table 4.3, further the average completing time and standard deviation of the completing time are given in Table 4.4.

Table 4.3 Synopsis of best fitness results for 100 runs with basic BA and enhanced BA for Himmelblau function.

	Mean value	Standard deviation
BA	9.6327	0.3984
Enhanced BA	9.7450	0.3138

Table 4.4 Synopsis of time taken for 100 runs with basic BA and enhanced BA for Himmelblau function.

	Mean value (secs)	Standard deviation (secs)
BA	19.21	0.5941
Enhanced BA	13.05	0.4752

The statistical significance between the best fitness results of the 100 runs for the Himmelblau function was assessed using the t-test, see Appendix D1-D2. The results are in terms of number of iterations is shown in Figure 4.9 and in terms of number of runs is shown in Figure 4.10. According to the given results, the alpha value for a two tailed test was 0.0278 with a 95% confidence level which strongly indicates that the ANSSA strategy-based improved BA is significantly better than the basic BA.

The next test performed with the parameters given in Table 4.2 on fifteen benchmark functions first presented by Adorio (2005) and Pohlheim (2006) and developed by Pham and Castellani (2009), as listed in Table 4.5. The comparative results of these benchmark tests are given in Table 4.6.

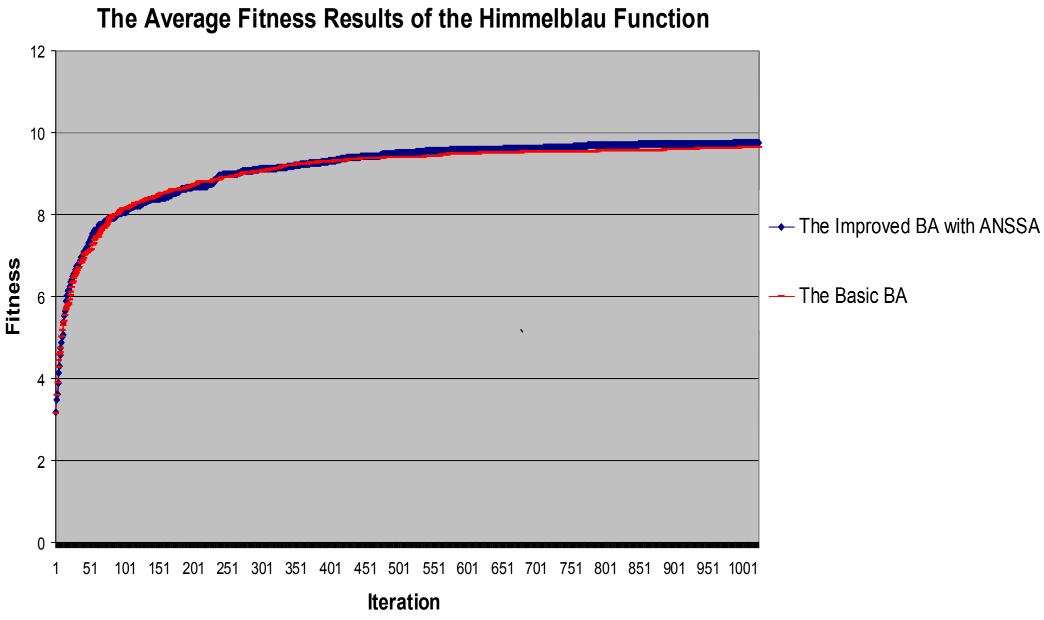


Figure 4.9 Average fitness value of the basic BA and the improved BA with ANSSA strategy (mean fitness value of 100 runs).

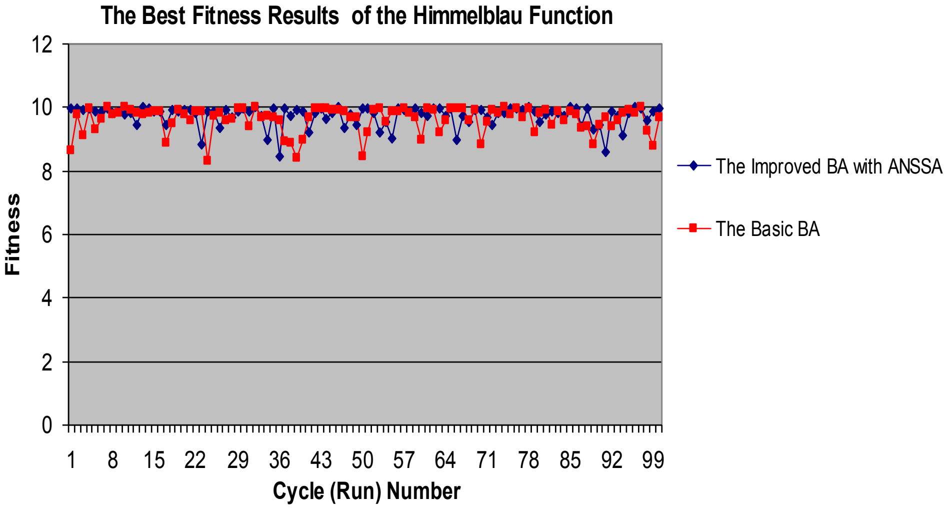


Figure 4.10 Best fitness results of the basic BA and Improved BA with ANSSA strategy (100 runs).

Table 4.5 The selected benchmark functions.

No	Function Name	Interval	Function	Global Optimum
1	De Jong (2D)	[-2.048, 2.048]	$max\ F = 3905.93 - 100(X_1^2 - X_2^2) - (1 - X_1)^2$	X = [1, 1] F(X) = 3905.93
2	Goldstein &Price (2D)	[-2, 2]	$min\ F = [1 + (X_1 + X_2 + 1)^2\ (19 - 14X_1 + 3X_1^2 - 14X_2 + 6X_1X_2 + 3X_2^2)]$ $[30 + (2X_1 - 3X_2)^2\ (18 - 32X_1 + 12X_1^2 + 48X_2 - 36X_1X_2 + 27X_2^2)]$	X = [0, -1] F (X) = 3
3	Branin RCOS (2D)	[-5, 10]	$min\ F = a(X_2 - bX_1^2 + cX_1 - d)^2 + e(1 - f)cos(X_1) + e$ $a = 1,\ b = \frac{5.1}{4\pi^2},\ c = \frac{5}{\pi},\ d = 6,\ e = 10,\ f = \frac{1}{8\pi}$	X = [-π, 12.275] X = [π, 2.275] X = [3π, 2.475] F(X) = 0.3977272
4	Martin & Gaddy (2D)	[0, 10]	$min\ F = (X_1 - X_2)^2 + (\frac{X_1 + X_2 - 10}{3})^2$	X = [5, 5] F(X) = 0
5	Rosenbrock - a (2D)	[-1.2, 1.2]	$min\ F = 100(X_1^2 - X_2)^2 + (1 - X_1)^2$	X = [1, 1] F(X) = 0

Table 4.5 The selected benchmark functions (Cont.).

6	Rosenbrock - b (4D)	[-1.2, 1.2]	$min\ F = \sum_{i=1}^{3}[100(X_i^2 - X_{i+1})^2 + (1 - X_i)^2]$	X = [1, 1, 1, 1] F(X) = 0
7	Hyper Sphere (6D)	[-5.12, 5.12]	$min\ F = \sum_{i=1}^{6} X_i^2$	X= [1, 1, 1, 1, 1, 1] F(X) = 0
8	Hyper Sphere (10D)	[-5.12, 5.12]	$min\ F = \sum_{i=1}^{10} X_i^2$	X = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0] F(X) = 0
9	Griwank (10D)	[-512, 512]	$max\ F = \frac{1}{[\sum_{i=1}^{10}\frac{X_i^2}{4000} - \prod_{i=1}^{10} cos(\frac{X_i}{\sqrt{i}})] + 0.1}$	X = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0] F(X) = 10
10	Rastrigin (10D)	[-5.12, 5.12]	$min\ F = 100 + \sum_{i=1}^{10}(X_i^2 - 10\,cos(2\pi X_i))$	X = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0] F(X) = 0

Table 4.5 The selected benchmark functions (Cont.).

11	Easom (2D)	[-100, 100]	$min\ F = -cos(X_1)cos(X_2)e^{-((X_1-\pi)^2+(X_2-\pi)^2)}$	X = [π, π] F(X) = -1
12	Schwefel (2D)	[-500 , 500]	$min\ F = \sum_{i=1}^{2}[-X_1\ sin(\sqrt{\lvert X_i \rvert})]$	X = [0, 0] F(X) = -837.658
13	Schaffer (2D)	[-100, 100]	$min\ F = 0.5 + \frac{(sin(\sqrt{X_1^2+X_2^2}))^2-0.5}{1+0.001(X_1^2+X_2^2)^2}$	X = (0, 0) F(X) = 0
14	Ackley (10D)	[-32, 32]	$min\ F = -20e^{-0.2\sqrt{\frac{\sum_{i=1}^{10}X_i^2}{10}}} - e^{\frac{\sum_{i=1}^{10}cos(2\pi X_i)}{10}} + 20 + e$	X = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0] F(X) = 0
15	Sum of Different Power (10D)	[-1, 1]	$min\ F = \sum_{i=1}^{10}\lvert X_i \rvert^{i+1}$	X = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0] F(X) = 0

Table 4.6 Comparative results of basic BA and the ANNSA strategy-based improved BA for 100 test runs of all fifteen benchmark functions.

	The ANNSA strategy based improved BA			The basic BA			Significance of difference in mean time between basic BA and enhanced BA	
Func. No	Success Rate %	Mean Iteration Number	Total Iteration Time (secs)	Success Rate %	Mean Iteration Number	Total Iteration Time (secs)	Sig. (α<0.05)	Alpha (α)
1	100	297	3’ 12”	100	342	3’ 33”	Yes	0.022
2	100	1497	24” 40	100	1497	17’ 21”	No	0.200
3	100	1497	10” 08’	100	1497	6” 09	No	0.600
4	100	293	3’ 20’	100	317	5’ 25”	No	0.110
5	100	183	3’ 51”	100	225	3’ 0”	No	0.358
6	100	184	8’ 15”	100	138	4’ 51”	No	0.957
7	100	101	6’ 32”	100	116	5’ 40”	No	0.762
8	100	150	17’ 4”	100	143	14’ 47”	No	0.433
9	100	734	83’3”	98	1117	88’ 41”	Yes	0.020
10	98	560	58’ 42”	98	700	56’ 39”	No	0.07
11	99	1497	28’ 26”	98	1497	23’ 10”	No	0.563
12	100	1497	28’ 41”	98	1497	20’ 51”	No	0.468
13	100	1497	21’ 40”	100	1497	30’ 11”	No	0.801
14	100	1402	153’ 5”	100	1226	112’ 21”	Yes	0.020
15	100	3	0’ 23”	100	5	0’ 29”	Yes	0.002

According to the results in Table 4.6, the ANSSA strategy-based improved BA performed well on functions 1, 9, 14, and 15 with a 100% success rate and the process time was also significantly better than the basic BA at better than a 5% level of confidence. The improved algorithm performed well for the rest of the benchmark functions but the process time did not significantly change compared to that of the basic BA.

4.4. BA Improved with Slope Angle Computation and Hill Climbing Algorithm

This section focuses on an improvement of the BA based on the Slope Angle Computation and Hill Climbing Algorithm (SACHCA). Even though the BA has both local and global search capability, it still has some weakness such as a high level of randomness, computational time, and blind search in the local search process. Thus local search improvement in the BA is proposed in this study.

The Hill Climbing Algorithm (HCA) is an iterative single element-based local search algorithm, also known as Gradient Ascent / Descent algorithms. The local minimum of an optimisation problem can be found by the HCA but the global optimum is not guaranteed (Grosan and Abraham, 2011).

The SACHCA-based improved BA is concerned with locating the best sites. Slope angle computation is employed to determine the inclination of the current sites. The promising location is far from the current position if the slope angle is close to 90°. The current location is near to a local optimum point if the slope angle is close to 0°. According to the angle orientation, the direction of the local optimum can be determined (see Figure 4.11). The process starts with computing the slope angle. The HC algorithm finds the related local optimum point when the slope angle of a related site is less than a certain threshold value. The slope angle computation can show the orientation of the local optimum according to the current position in terms of angular degrees, which may help to find the global optimum faster. With honey bees this directional information is conveyed by a waggle dance. The slope angle computation approach mimics this behaviour to determine the direction of promising sites. To increase the speed of an optimisation algorithm, the information about the orientation of promising locations can be utilised to boost the search process.

In this study, the slope angle is computed using the first order numerical derivation. The numerical derivation of each site is calculated from its neighbourhood. The two end points in the vicinity of the neighbourhood are used to compute the numerical derivation. The central difference method is utilised for numerical derivation (see Equation 4.5) as shown in Figure 4.11.

$$slope\_angle = F'(X) = \frac{F(X + \frac{\Delta X}{2}) - F(X - \frac{\Delta X}{2})}{\Delta X} \tag{4.5}$$

If the slope angle is very steep, then the promised location is far from the selected site but when the angle is close to zero, the promised location is very close to the selected site, as shown in Figure 4.12.

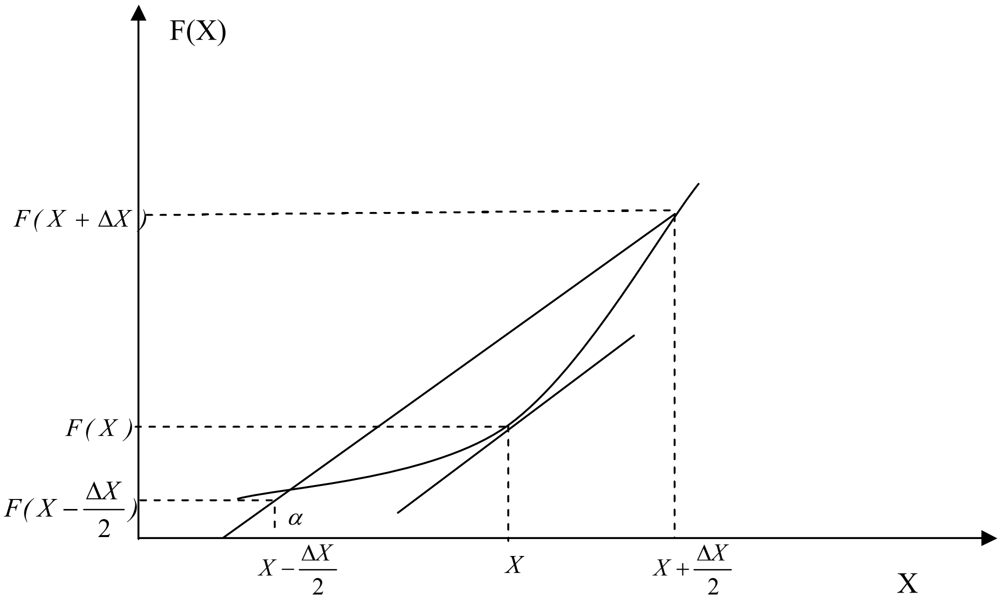


Figure 4.11 Slope angle with numerical derivation based on the central difference.

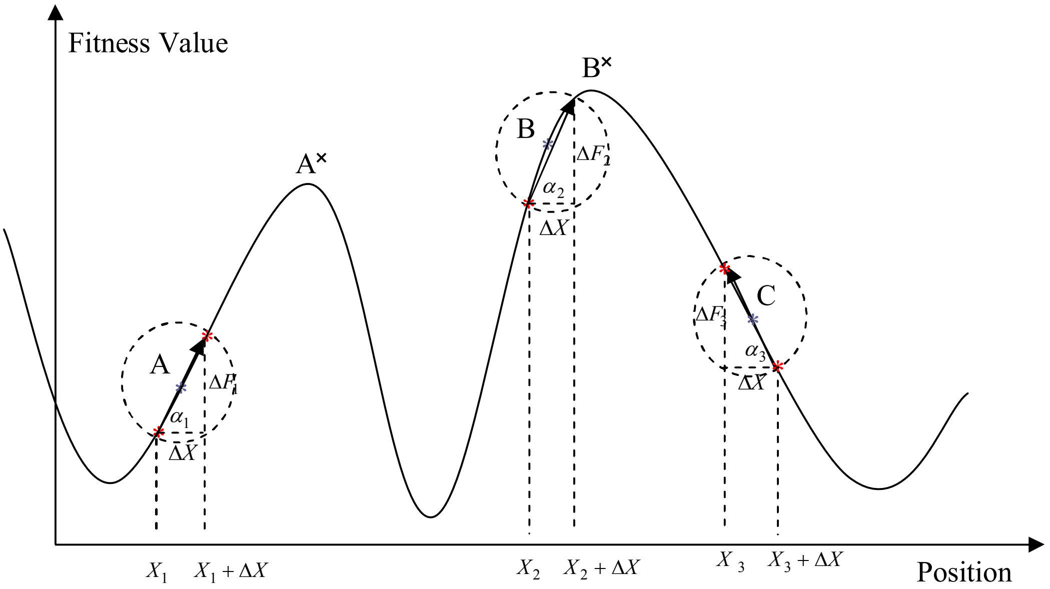


Figure 4.12 Slope angle for each of the best selected sites.

There are three selected (best) sites shown in Figure 4.12. Around site ***A***, the angle direction is towards site $\boldsymbol{A}^{\times}$, which is a local optimum. The direction of sites ***B*** and ***C*** are towards the site $\boldsymbol{B}^{\times}$, which is another local optimum. At the end of the searching process, all the local optimums will be sorted and the biggest selected as the global optimum. In the case of Figure 4.12, the site $\boldsymbol{B}^{\times}$ was selected as global optimum. The local search process was accomplished with the use of the HCA, as shown in Equation 4.6.

$$X(i+1) = X(i) + h\nabla F(X_i) \tag{4.6}$$

where i is the iteration number, $X(i)$ is the current position, $X(i+1)$ is the next position, h is the incremental size, $\nabla F(X_i)$ is the gradient of the current position.

The BA improved with SACHCA has four more additional parameters than the basic BA. The first parameter is step size, denoted as h and is used as the step size of the HCA to predict the next position.

The second parameter is the slope angle, denoted as $slope\_angle$. This parameter is used for identifying the direction of the local optimum site. Initially, the angle of steepness is set to zero.

The third parameter is called slope angle limit, denoted as $angle\_limit$. If the angle of slope is equal or less than $angle\_limit$, then the local optimum is close to the related site. This parameter is also the starting condition for the HCA. The predicted local optimum points are stored and compared with the other local optima.

The forth parameter is called the “number of waiting time for the HCA”, denoted as HC_time_limit. The HCA runs for the selected sites until the “number of waiting time for the HCA” is reached which is the termination condition.

The pseudo code of the SACHCA-based improved BA is shown in Figure 4.13

```
Generate the initial population size as $n$ , set the best patch size as $m$, set the elite patch size as $m$ , set the number of forager bees around of elite as $nep$, $m$, set the number of forager bees around of non-elite best patches as $nsp$ , set the step size for HC algorithm as $h$ , set the angle limit as $angle\_limit$, set the "number of the waiting time for HC" algorithm as $HC\_time\_limit$ , set the neighbourhood size as $ngh$, set the maximum iteration number as $MaxIter$, and set the error limit as $Error$ .

i= $0, time = 0$ ; $slope\_angle(1:m) = 0$ ; Generate initial population; Evaluate Fitness Value of initial population;

Sort the initial population based on the fitness result.

While $i \le MaxIter$ or $FitnessValue_i - FitnessValue_{i-1} \le Error$

    i= i+1;

    Select the elite patches and non-elite best patches for neighbourhood search.

    Recruit the forager bees to the elite patches and non-elite best patches.

    Evaluate the fitness value of each patch; Sort the results based on their fitness.

    $For \quad k=1:m$

    Calculate $angle(k)$

    While $slope\_angle(k) > angle\_limit$ and $time \le HC\_time\_limit$ then.

    $X(i+1,k)=X(i,k)+h\nabla F(X(i.,k))$.

    Evaluate Fitness value for each position

    $End$

    Record all found local optimum sites and sort them (end of the neighbourhood search).

    Allocate the rest of the bees for global search to the non-best locations;

    Evaluate the fitness value of non-best patches;

    Sort the fitness values and positions; Run the algorithm until termination criteria are met.

End
```

Figure 4.13 Pseudo code of the improved BA based on SACHCA.

4.4.1. Experimental Results for the BA Improvement based on SACHCA

The performance of the BA enhanced by use of the SACHCA was tested against an inverted and translated Himmelblau function and the fifteen benchmark functions described above in Section 4.3.1. The results were compared with those obtained with the basic BA and the BA improved with the ANSSA strategy. The parameters set for the proposed algorithm are given in Table 4.7.

The comparative best fitness results are given in Figure 4.14. All the results are an average of 100 runs. The mean value and standard deviation of the basic BA, ANSSA strategy based BA and SACHCA based improved BA are given in Table 4.8, further the average completing time and standard deviation of the completing time are given in Table 4.9. According to results, the BA improved with SACHCA performed better than the other two algorithms.

Table 4.7 Test parameters for BA improved with SACHCA.

Parameters	Symbols
Number of Scout Bees in the Selected Patches (n)	50
Number of Best Patches in the Selected Patches (m)	15
Number of Elite Patches in the Selected Best Patches (e)	3
Number of Recruited Bees in the Elite Patches (nep)	12
Number of Recruited Bees in the Non-Elite Best Patches (nsp)	8
The Size of neighbourhood for Each Patches (ngh)	1
Number of Iterations ($iter$)	5000
Difference between the First Iteration Value and the Last Iteration ($diff$)	0.001
Angle limit ($angle\_\lim it$)	0.5
Step size for HC algorithm (h)	0.1
Number of waiting time for HC algorithm ($HC\_time\_\lim it$)	1000

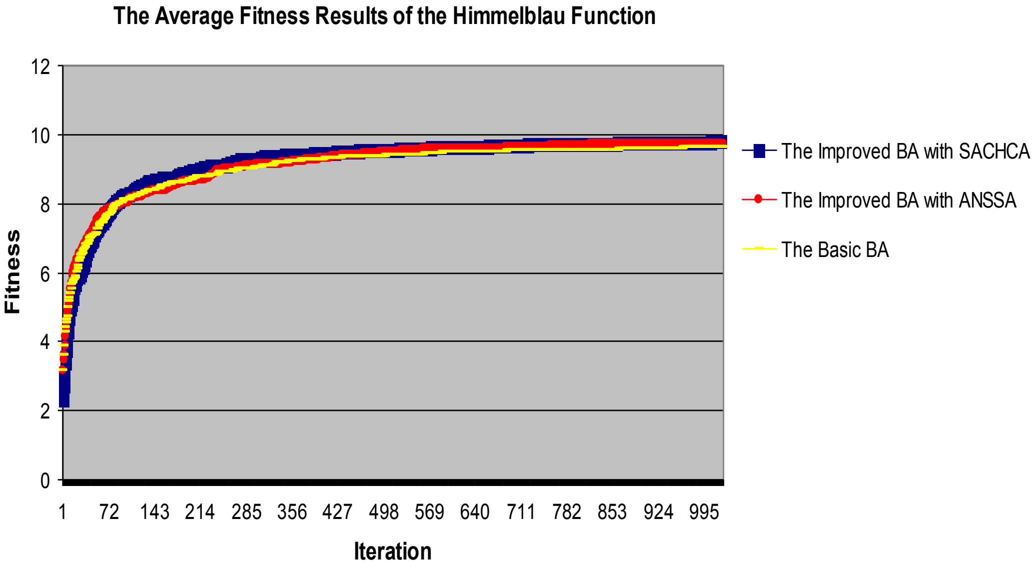


Figure 4.14 Average fitness value of the basic BA, BA improved with ANSSA strategy and BA improved with SACHCA (mean fitness value of 100 runs).

Table 4.8 Synopsis of best fitness results for 100 runs with basic BA and enhanced BA for Himmelblau function.

Algorithms	Mean value	Standard deviation
The Basic BA	9.6327	0.3984
The BA improved with ANSSA Strategy	9.7450	0.3138
The BA improved with SACHCA	9.8240	0.1807

Table 4.9 Synopsis of time taken for 100 runs with basic BA and enhanced BA for Himmelblau function.

Algorithms	Mean value	Standard deviation
The Basic BA	19.21	0.5941
The BA improved with ANSSA Strategy	13.05	0.4752
The BA improved with SACHCA	9.23	0.0197

The t-test was used to calculate the statistical significance of the difference between the best fitness values obtained from the three algorithms. The results are given in Appendix D3. The test was performed in pairs such as, the improved BA with SACHCA and one of the others. The t-test between mean best fitness values for the SACHCA based BA and basic BA gave an alpha value of $2.379x10^{-5}$ which means it is possible to say with greater than 95% confidence level that there is a significant difference between the mean values obtained with these two algorithms. Similarly mean value obtained for the BA improved with ANSSA was significantly higher than that obtained with the basic BA. The best fitness results for 100 runs of the improved BA with SACHCA and the basic BA are given in Figure 4.15.

The third comparison, between the mean best fitness values for the SACHCA based BA and the BA improved ANSSA, showed that the former is significantly greater than the latter at the 95% confidence level. The t-test gave an alpha value of 0.003. The 100 best fitness results of both algorithms are given in Figure 4.16.

Further, the proposed algorithm has been tested on fifteen benchmark functions which were given in Table 4.5 and the results were compared with the results of the basic BA and ANSSA strategy based enhanced BA, given Table 4.10.

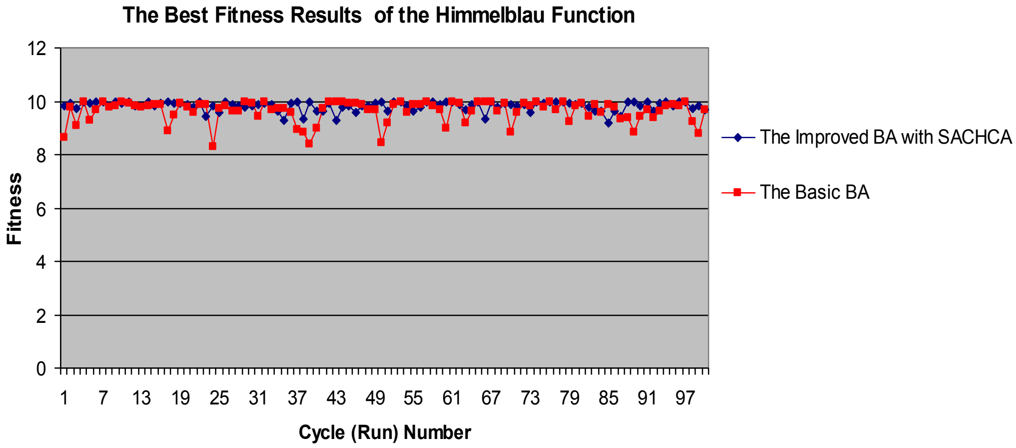


Figure 4.15 Best fitness value of the basic BA and the BA improved with SACHCA (100 runs).

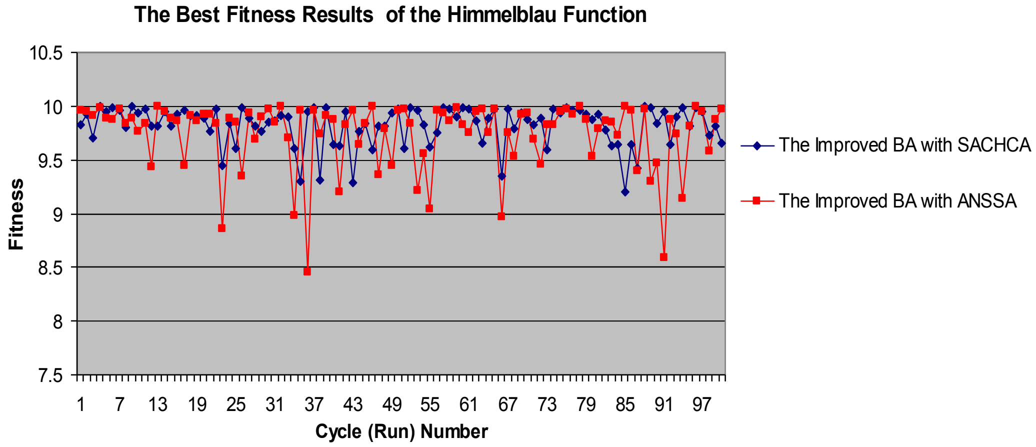


Figure 4.16 Best fitness value of BA improved with ANSSA and BA improved with SACHCA (100 runs).

Table 4.10 Comparative results of fifteen benchmark functions for the BA improved with SACHCA, BA improved with ANNSA and basic BA.

Func No	Basic BA (A1)			BA Improved with ANNSA (A2)			BA Improved with SACHCA (A3)			Significance of difference between (A3-A1)		Significance of difference between (A3-A2)	
	Success Rate %	Mean Iteration Number	Total Iteration Time (sec)	Success Rate %	Mean Iteration Number	Total Iteration Time (sec)	Success Rate %	Mean Iteration Number	Total Iteration Time (sec)	Significant (α<0.05	α	Significant (α<0.05	α
F1	100	297	3’ 12’’	100	342	3’ 33’’	100	240	2’ 43’’	Yes	0.002	Yes	0.0001
F2	100	1497	24’’ 40	100	1497	17’ 21’’	100	1497	10’ 51’’	Yes	1.58E-08	Yes	8.79E-11
F3	100	1497	10’’ 08’	100	1497	6’’ 09	100	1497	4’ 22’’	Yes	0.0003	Yes	5.68E-06
F4	100	293	3’ 20’	100	317	5’ 25’’	100	359	3’ 18’’	Yes	0.0018	Yes	0.0392
F5	100	183	3’ 51’’	100	225	3’ 0’’	100	162	3’ 20’’	No	0.1595	No	0.2908
F6	100	184	8’ 15’’	100	138	4’ 51’’	100	141	5’ 12’’	No	0.9653	No	0.9478
F7	100	101	6’ 32’’	100	116	5’ 40’’	100	133	6’ 53	No	0.4496	No	0.4426
F8	100	150	17’ 4’’	100	143	14’ 47’’	100	121	12’ 04’’	Yes	6.16E-09	Yes	3.46E-09

Table 4.10 Comparative results of fifteen benchmark function for the improved BA with SACHCA, improved BA with ANNSA and basic BA (Cont.).

F9	100	734	83’ 3’’	98	1117	88’ 41’’	100	535	51’ 39’’	No	0.9567	Yes	0.0041
F10	98	560	58’ 42’’	98	700	56’ 39’’	100	644	53’ 21’’	Yes	0.0098	Yes	5.6E-06
F11	99	1497	28’ 26’’	98	1497	23’ 10’’	100	1497	9’ 58’’	No	0.1040	Yes	0.0268
F12	100	1497	28’ 41’’	98	1497	20’ 51’’	100	1497	9’ 40’’	No	0.4553	No	0.9692
F13	100	1497	21’ 40’’	100	1497	30’ 11’’	100	1497	9’ 28’’	No	0.2599	No	0.2652
F14	100	1402	153’ 5’’	100	1226	112’ 21’’	100	1497	103’19’’	Yes	6.08E-06	Yes	0.0125
F15	100	3	0’ 23’’	100	5	0’ 29’’	100	38	2’ 34’’	Yes	3.27E-09	Yes	2.21E-17

According to Table 4.10, the BA improved with SACHCA performed well with benchmark test functions F1, F2, F3, F4, F8, F10, F14 and F15 (refer to Table 4.5). The total iteration time of the proposed algorithm is better than the others. Finally, the proposed algorithm found the global optimum value in every run.

To calculate the statistical significance between the BA improved with SACHCA and other BA versions, t-test was utilised. According to the Table 4.10, the proposed algorithm performed better than the others for the following functions: F1, F2, F3, F4, F8, F10, F14 and F15; because the alpha values were found lower than 0.05 with a 95% confidence level for both comparison between the BA improved with SACHCA – basic BA (denoted as A3-A1 in Table 4.10) and the BA improved with SACHCA – the BA improved with ANSSA (denoted as A3-A2 in Table 4.10).

The BA improved with SACHCA performed well on functions F5, F6, F7, F12, and F13, achieving a 100% success rate and the process times were faster than those of the basic BA. However, the results were not significant than the other BA versions.

Finally, the results of SACHCA-based BA achieved with functions F9 and F11 were better than other BA versions, although functions F9 and F11 are 10 dimensional and a hard group of functions (Pham and Castellani, 2009), The total iteration times of the BA improved with SACHCA for the given functions were less than those of the basic BA and the BA improved with ANNSA, which means the process time is also improved with the BA improved with SACHCA.

4.5. Summary

This chapter has presented two improvements to the BA.

The first improvement was based on changing the neighbourhood size strategy and on having a more adaptive neighbourhood size change during the search process.

The second improvement was to determine the direction of local optima by using the slope angle of a neighbourhood. The HCA was then utilised on the sites which had a moderate slope angle.

By including the second improvement, both the search speed was improved and more accurate results were obtained.

Both improvements to the BA were tested on the Himmelblau function and fifteen selected benchmark functions. The results have been presented in this chapter.

The results of the proposed algorithms are satisfactory for the given optimisation problems. The improved BA algorithms and the basic BA will therefore be used in the next chapter as a new depth calculation technique based on the BA as an optimisation tool to find the best possible level of blurriness.

Chapter 5

A NOVEL DEPTH COMPUTATION TECHNIQUE WITH A BEES ALGORITHM BASED BLIND IMAGE DECONVOLUTION

5.1. Preliminaries

This chapter presents a novel depth computation technique based on optimised Blind Image Deconvolution (BID). The optimisation process uses the basic Bees Algorithm (BA), BA enhanced by SACHCA and BA enhanced with ANNSA as described in Chapter 4. For comparison a simulated annealing (SA)-based BID is also used.

A blurred image can be caused by changes in optical system parameters such as the focal length, object position in the optical system, aperture diameters, f-number of the camera, etc. The main task of the deconvolution process is to recover the original image and Point Spread Function (PSF), see Equation 2.1.

The PSF of a defocused image can be used for many applications in machine vision such as depth calculation, image restoration and image de-blurring. In practice, the PSF of a blurred image is not known and while it is not an easy task to determine it without having some prior information a number of approaches have been proposed for computing the PSF without prior information.

The proposed techniques utilise kurtosis-based non-Gaussianity analysis and the BA to find the optimum value of the PSF blurriness parameter, sigma (σ), which is the standard deviation of the PSF. Then the determined parameter can be correlated with the distance from the surface of the sample to the objective lens of the SEM to define the relationship between sigma and depth.

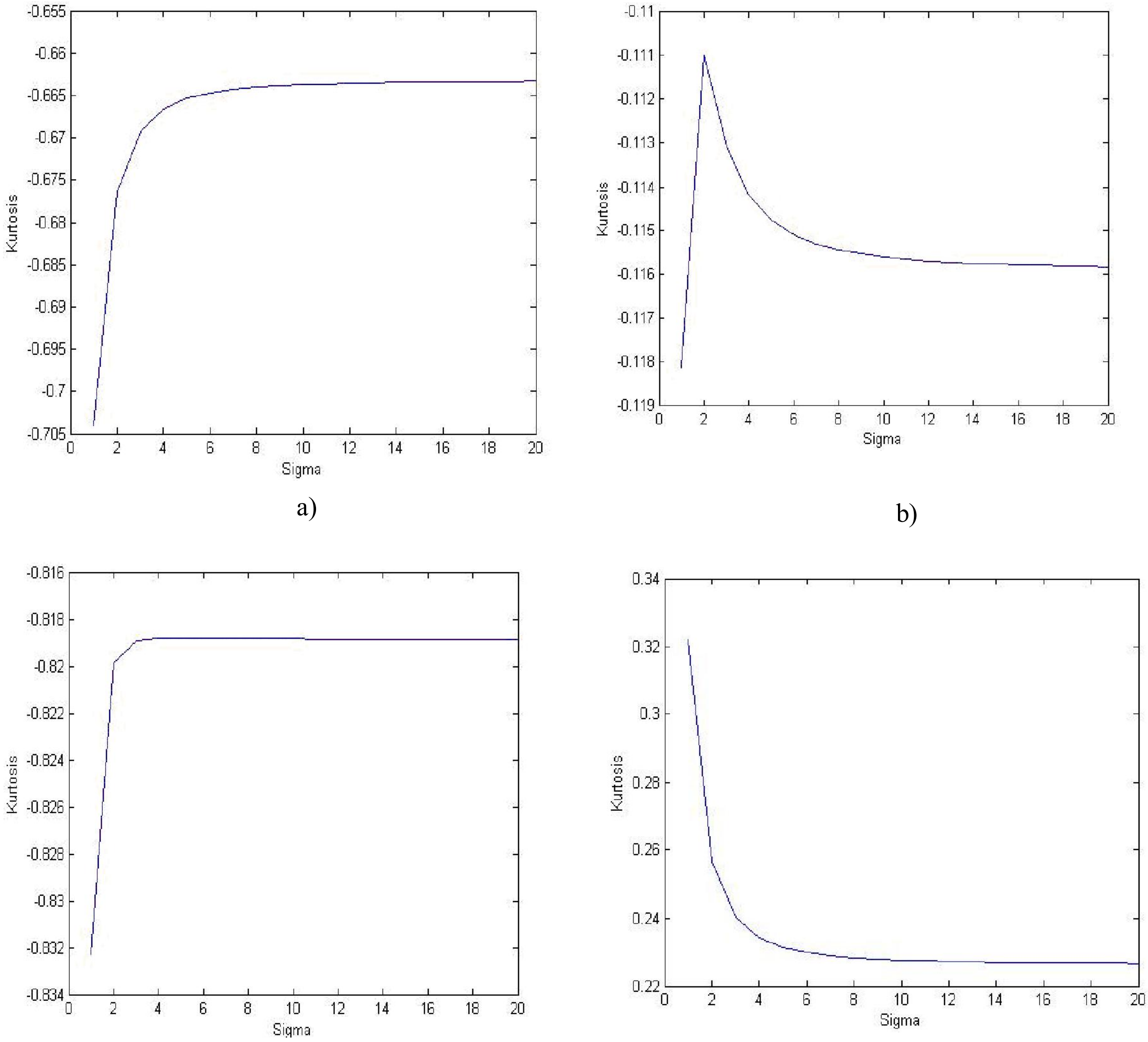
5.2. Kurtosis of Blurred Images

Different levels of blurred images have different kurtosis values which are related to the blurriness parameter. In this section, the effect of changes in σ on the kurtosis value of the image is investigated. To show the relationship between sigma and kurtosis, the four images of Lena, Boats, Cameraman and Copper sample captured with a SEM were used. The original images are given in Figure 5.1 a-d. All four images were blurred with a Gaussian filter by changing the σ value from 1 to 20 and the relationship between σ and kurtosis for the images is shown in Figure 5.2 a-d.

a) b)

c) d)

Figure 5.1 Images for kurtosis analysis a) Lena, b) Boats, c) Cameraman and d) Copper sample.



a)

b)

c)

d)

Figure 5.2 Relationship between sigma and kurtosis for the given images a) Lena, b) Boats, c) Cameraman and d) Copper sample.

According to the Central Limit Theorem (CLT) the distribution of the mean of *n* independent random variables approaches a *Normal Gaussian* distribution. In the case of a blurred image, there are two main components, the original image and the PSF that together create the blurred image. As explained in Section 2.5.2.4 the kurtosis value for a Gaussian distribution is zero so it is expected that the kurtosis value for a blurred image is closer to zero than that for a sharp image.

In Figure 5.2 it can be seen that the results are promising; as σ increases the kurtosis decreases for the images of Lena, Cameraman and the Copper sample. However, the results for the image of the Boats show that the kurtosis decreases with increase in σ only for $\sigma > 2$. The main reason for this is because of the large number of pixels in the areas of sky and sea where the intensity between a pixel and its neighbours are not so different. To overcome inconsistent changes in the kurtosis values, (Yu, 2008) proposed a whitening pre-processing stage.

5.2.1. Whitening Process

The whitening process is an essential pre-processing technique for independent component analysis ICA (Hyvarinen et al., 2001). The whitening process finds the low-correlated sub-data set from the raw data set and eliminates pixels where the number of pixels is higher than a certain threshold value. Yu (2008) proposed a threshold computational technique for the whitening process, as given in Equations 5.1-5.2:

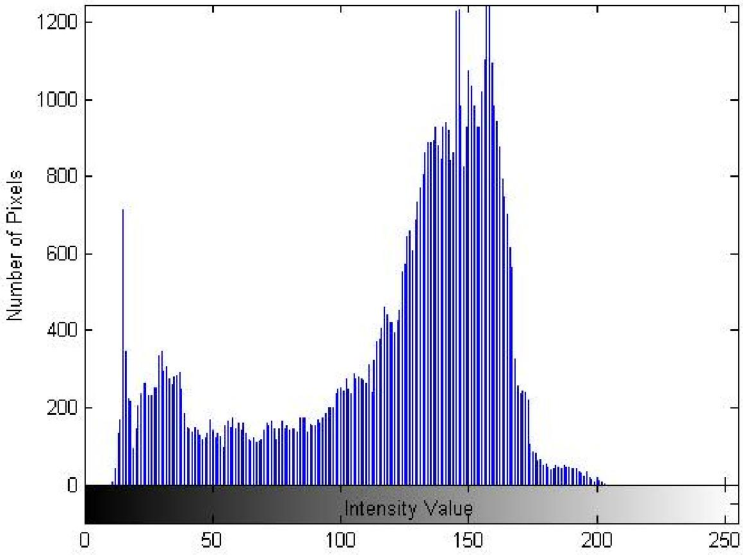
$$Threshold = Median + \alpha(Standard\_Deviation) \qquad (5.1)$$

where $Median$ is the median value of the histogram data representing pixel intensity against number of pixels, see Figure 5.3, α is a scale factor, $Standard\_Deviation$ is the standard deviation of the histogram data.

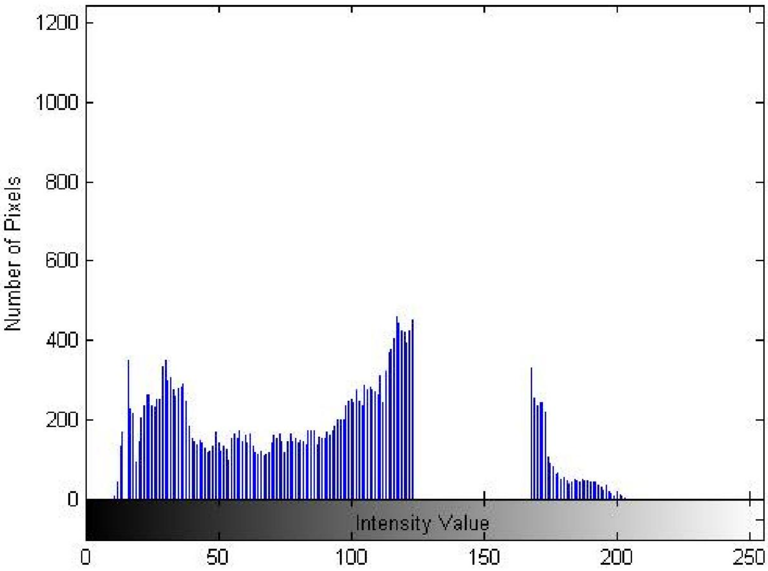
$$\alpha = \begin{cases} 2 & if \quad [(Maximum - Median)/Standard\_Deviation] < 2 \\ 1 & if \quad [(Maximum - Median)/Standard\_Deviation] \geq 2 \end{cases} \qquad (5.2)$$

where $Maximum$ is the maximum value of the histogram data.

The whitening process was applied to the histogram of the original Boats image. The histogram of the original image is given in Figure 5.3a. According to Equations 5.1 and 5.2, a threshold value was found as 861.4. Then the pixel elimination process was applied according to the computed threshold value. The final histogram is given in Figure 5.3b. After the whitening process, the image data became low-correlated. The final image of Boats is given in Figure 5.4.



a)



b)

Figure 5.3 Histogram of the image of Boats a) before the whitening process, b) after the whitening process.

Figure 5.4 Image of Boats after the whitening process.

5.3. Determining the Blurriness Parameter of the PSF with Kurtosis-Based Non-Gaussianity Analysis

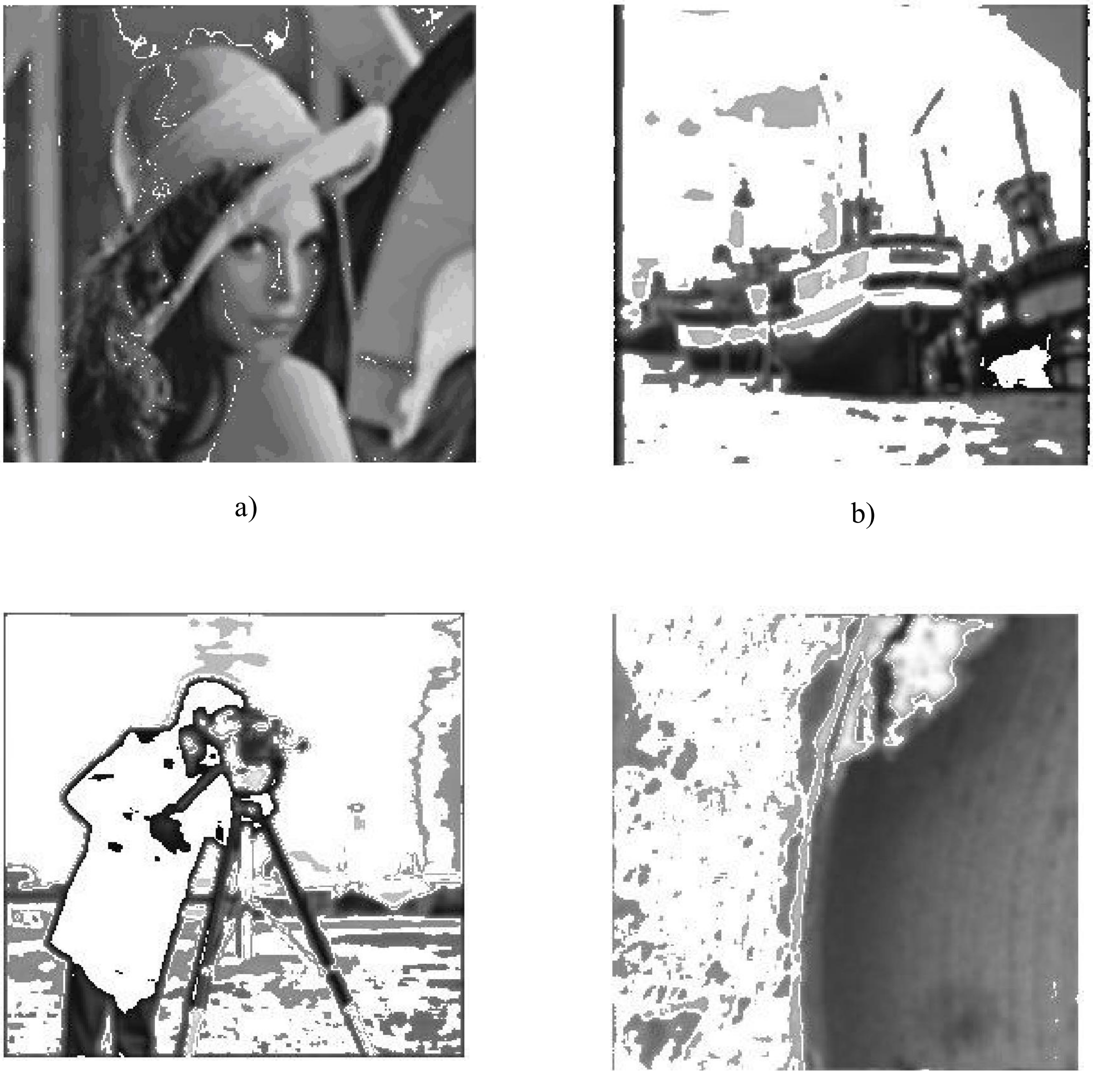
This section reports the determination of the blurriness parameter using kurtosis-based non-Gaussianity analysis. The sharper the image the higher the kurtosis value and this information can be used to determine exactly the blurriness parameter of a blurred image. In this section, the images of Lena, Boats, Cameraman and Copper sample are blurred using $\sigma = 5$, This value of σ was selected because, as shown in Figure 5.2, the corresponding kurtosis is decreasing and close to its minimum value. The blurred images are shown in Figure 5.5 a-d.

a) b)

c) d)

Figure 5.5 Images blurred using $\sigma = 5$; a) Lena, b) Boats, c) Cameraman and d) Copper sample.

All the blurred images were whitened as described in the previous section to increase consistency, see Figure 5.6 a-d. The blurriness prediction results are shown in Figure 5.7 a-d for $\sigma = 5$. The determination process worked with all four cases.



a)

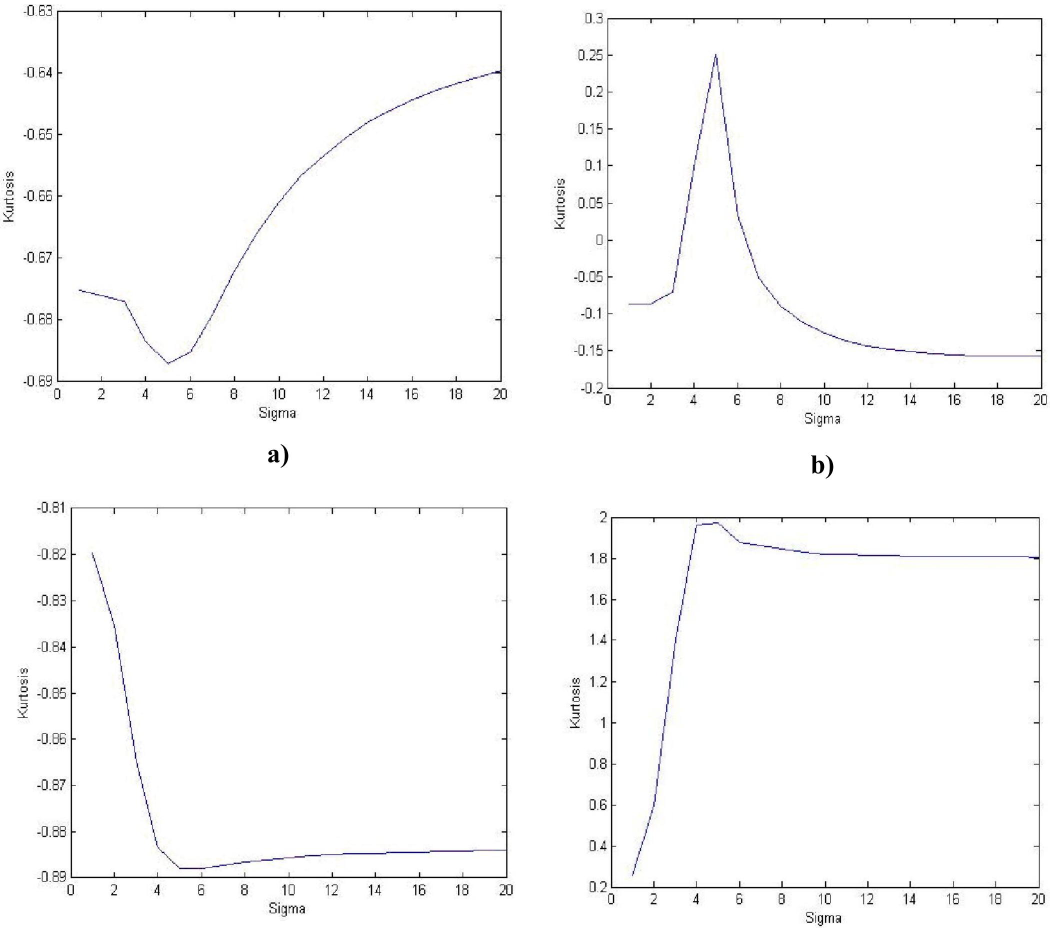
b)

c)

d)

Figure 5.6 Blurred and whitened images ($\sigma = 5$); a) Lena, b) Boats, c) Cameraman and d) Copper sample.

The next step was to find the blurriness parameters with an optimised BID process. In practice, determination of the blurriness parameter is a blind search process. It is called a blind search process because there is no information on the level of blurriness or original image. In the next section, an optimised BID technique is used to overcome this problem and find the optimum blurriness parameter.



a)

b)

c)

d)

Figure 5.7 Blurriness parameter search for the blurred ($\sigma = 5$) and whitened images a) Lena, b) Boats, c) Cameraman and d)) Copper sample.

5.4. The ICA-Based Optimised BID technique with the BA

The aim of this section is to find the optimum blurriness parameter and it focuses on the ICA-based optimised BID technique with BA. The BID technique utilises the kurtosis-based non-Gaussianity analysis and the kurtosis is employed as a fitness function in the BA. The basic BA, BA enhanced by SACHCA and BA enhanced with ANNSA as described in Chapter 4 were used here.

In this study, the PSF of the blurred image is assumed to be 2D-Gaussian function. The standard deviation, σ of the PSF was searched by the proposed BAs. The global optimum value of the standard deviation was utilised to find the PSF. Hence, the original image can also be estimated.

In the next sections, the following techniques will be explained; the ICA-based optimised BID technique with the basic BA, ICA-based optimised BID technique with the ANSSA strategy-based improved BA and ICA-based optimised BID with the SACHCA-based improved BA.

5.4.1. ICA Based Optimised BID with Basic BA

The ICA-based BID with the basic BA is based on optimisation of the blurriness parameter. The basic BA was employed to find the global maximum value of the kurtosis of the image by evaluating the blurriness parameter, σ. The pseudo code of the technique is shown in Figure 5.8.

```
Set the initial population as $n$, the best locations as $m$, the elite locations as $e$, the number of bees around elite locations as $neb$, the number of bees around non-elite best locations as $nsb$, the neighbourhood size as $ngh$, number of maximum iterations as $MaxIter$ and error limit as $Error$.

Generate initial population for PSF parameter; sigma, $\sigma$ and substitute the sigma in Gaussian PSF as $h(x,y)$, evaluate the kurtosis value as and sort absolute value of kurtosis in descending order for initial population.

Compute the Fourier transform of the observed image as $G(m.n)$ and predicted PSF as $H(x,y)$.

Then restore the estimate of the original image in Fourier domain as, $\hat{F}(m.n)$ with following equation, $\hat{F}(m,n) = \frac{G(m,n)}{\hat{H}(m,n)}$ then compute the inverse Fourier transform of the estimated original image as $\hat{f}(x,y)$. $i = 0$.

While $i \le MaxIter$ or $Kurtosis_i - Kurtosis_{i-1} \le Error$
    $i = i + 1$
    Recruit forager bees to the elite sites and non-elite best sites for neighbourhood search.
    Evaluate Ffitness Value. Sort Fitness Value.
    Allocate the rest of the forager bees for global search.
    Evaluate Fitness Value.
    Sort the final result in descending order and the maximum kurtosis value.
    Run the program until the termination conditions met.
End
```

Figure 5.8 Pseudo code of the ICA-based optimised BID technique with basic BA.

5.4.2. ICA-Based Optimised BID with ANSSA Strategy-Based Improved BA

The ANSSA strategy-based improved BA was proposed in the previous chapter and tested on continuous type optimisation problems.

In this section, the algorithm was utilised for the optimisation of the ICA-based BID technique. The main goal of the proposed technique was to find the optimum value of σ when the maximum non-Gaussianity has been found. The pseudo code of the technique is shown in Figure 5.9.

```
Set the initial population as n, the best locations as m, the elite locations as e, the number of bees around of elite locations as neb, the number of bees around non-elite best locations as nsb, set the shrinking constant as sc, set the shrinking start number as rep_nshr, set the enhancement start point as rep_nenh, set the abandoning start number as rep_naban, set the neighbourhood size as ngh, number of maximum iteration as select MaxIter and error limit as Error.

$keep\_point = Zeros(1,m)$; $aband\_site = Zeros(iter,m)$.

Generate initial population for PSF parameter; sigma, $\sigma$ and substitute the sigma in Gaussian PSF as $h(x,y)$, evaluate the kurtosis value as and sort absolute value of kurtosis as descend order for initial population.

Calculate the Fourier transform of the observed image as $G(m.n)$ and predicted PSF as $H(x,y)$.

Then restore the estimated original image in Fourier domain as, $\hat{F}(m.n)$ with following equation, $\hat{F}(m,n) = \frac{G(m,n)}{\hat{H}(m,n)}$

then compute the inverse Fourier transform of the estimated original image as $\hat{f}(x,y)$. $i = 0$;

While $i \le MaxIter$ or $Kurtosis_i - Kurtosis_{i-1} \le Error$

    $i = i + 1$; Recruit the forager bees to the elite sites and non-elite best locations for neighbourhood search.

        $keep\_point(i,k) > rep\_nshr$ and $keep\_point(i,k) \le rep\_nenh$;

        $new\_ngh(i,k) = ngh - (ngh * sc * \frac{100-(keep\_point(i,k) - rep\_nshr)}{100})$;

        End

        If $keep\_point(i,k) > rep\_nenh$ and $keep\_point(i,k) \le rep\_naban$

        $new\_ngh(i,k) = ngh + (ngh * sc * \frac{100-(keep\_point(i,k) - rep\_naban)}{100})$;

        End

        Else If $keep\_point > naban$

        $aband\_site(i+1,k) = aban\_site(i,k) + 1$;

        Else $new\_ngh(i,k) = ngh$;

        End

    Evaluate Fitness Value. Sort Fitness Value. Allocate the rest of the forager bees for global search.

    Evaluate Fitness Value. Sort the final result as descend order and the maximum kurtosis value.

    Run the program until the termination conditions met.

End
```

Figure 5.9 Pseudo code of the ICA-based optimised BID technique with the ANSSA strategy-based improved BA.

5.4.3. ICA-Based BID with the SACHCA-Based Improved BA

The SACHCA-based improved BA was proposed in the previous chapter and tested on continuous type optimisation problems.

In this section, the algorithm was used to optimise the ICA-based BID technique. The proposed technique looks for the optimum sigma value when kurtosis-based non-Gaussianity analysis has found a maximum. The pseudo code of the technique is shown in Figure 5.10.

```
Set the initial population as n, the best locations as m, the elite locations as e, the number of bees around of elite locations as neb, the number of bees around non-elite best locations as nsb, set the step size for HC algorithm as h, set the angle limit as angle_limit, set the number of the waiting time for HC algorithm as HC_time_limit, the neighbourhood size as ngh, number of maximum iteration as MaxIter and error limit as Error.

Generate initial population for PSF parameter; sigma, $\sigma$ and substitute the sigma in Gaussian PSF as $h(x,y)$, evaluate the kurtosis value and sort absolute value of kurtosis as descend order for initial population. Calculate the Fourier transform of the observed image as $G(m.n)$ and predicted PSF as $H(x,y)$. Then restore the estimated the original image in Fourier domain as, $\hat{F}(m.n)$ with following equation, $\hat{F}(m,n)=\frac{G(m,n)}{\hat{H}(m,n)}$ then compute the inverse Fourier transform of the estimated original image as $\hat{f}(x,y)$. $i=0$;

While $i \le MaxIter$ or $Kurtosis_i - Kurtosis_{i-1} \le Error$
  $i=i+1$
  Select the elite locations and non-elite locations for neighbourhood.
      $For \quad k=1:m$
      Calculate $angle(k)$
      While $angle(k) > angle\_limit$ and $time \le HC\_time\_limit$ then.
      $X(i+1,k)=X(i,k)+h\nabla F(X(i.,k))$.
      Evaluate Fitness Value for each position
      $End$
  Evaluate fitness value. Sort Fitness value.
  Allocate the rest of the forager bees for global search. Evaluate Fitness Value.
  Sort the final result as descend order and the maximum kurtosis value.
  Run the program until the termination conditions met;
End
```

Figure 5.10 Pseudo code of the ICA-based optimised BID technique with the SACHCA-based improved BA.

5.4.4. Experimental Results for the ICA-Based BID Technique with the BA

This section presents the experimental results obtained for the ICA-based BID technique with the three proposed versions of the BA (basic BA, the ANSSA strategy-based improved BA and the SACHCA-based improved BA). The blurred images of Lena, Cameraman, Boats and the Copper sample were used to measure the performance of the proposed BID techniques. All four entire images were blurred using $\sigma = 5$ (see Figure 5.5 a-d). Then the given images were pre-processed with a whitening process. Finally, each of the proposed BA versions was used with the BID technique to find the value of the optimum blurriness parameter (sigma). The results of the BA-based BID techniques were compared with a Simulated Annealing (SA)-based BID technique as given in Section 2.5.1, the model proposed by (Yu, 2008). The parameters of the BA-based BID techniques are given in Table 5.1. The simulation parameters for the SA algorithm were as follows: the highest temperature was set 20, the lowest temperature was 0, the number of iterations was selected as 100.

The average results for the blurriness parameter found by the different algorithms are shown in Figures 5.11-5.14. Each algorithm was run 100 times. The best results of each run are given in Figures 5.15-5.18. Further, the average value and standard deviation of 100 best fitness results of each algorithm for the given images are given in Table 5.2. Finally, the Root Mean Square (RMS) errors for the techniques are given in Table 5.3. The RMS error was calculated as in following Equation 5.3:

$$RMS = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - Exp)^2} \tag{5.3}$$

where Exp =5 (expected value), x_i= best value of each run, $n = 100$(number of run).

Table 5.1 Selected parameters for the BA.

Parameters	Basic BA	ANSSA Based BA	SACHCA Based BA
Number of scout bees in the selected patches (*n*)	10	10	10
Number of best patches in the selected patches (*m*)	5	5	5
Number of elite patches in the selected Best patches (*e*)	1	1	1
Number of recruited bees in the elite patches (*nep*)	5	5	5
Number of recruited bees in the non- elite best patches (*nsp*)	3	3	3
The size of neighbourhood for each patch (*ngh*)	0.01	0.01	0.01
Shrinking constant (*sc*)	-	0.1	-
Number of repetitions for shrinking process (*rep_nshr*)	-	10	-
Number of repetitions for enhancement process (*rep_nenh*)	-	30	-
Number of repetitions for site abandonment (*rep_naban*)	-	100	-
Angle limit (*angle_limit*)	-	-	0.5
Step size for HCA (h)	-	-	0.1
Number of waiting time for HCA (HC_time_limit)	-	-	100
Number of Iterations (*iter*)	1000	1000	1000
Difference between the first and last iteration value (*diff*)	0.001	0.001	0.001

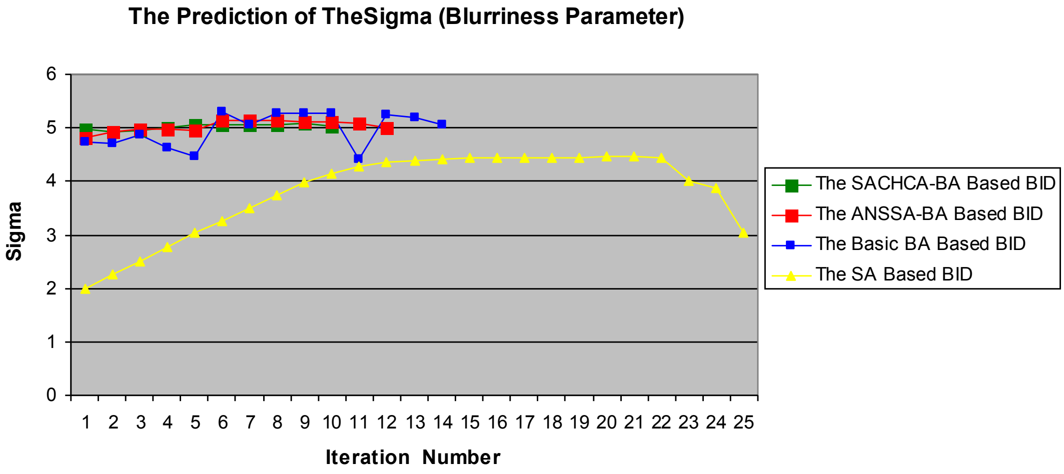


Figure 5.11 Blurriness parameter for the blurred-whitened image of Lena as determined by the SA based BID and three BA-based BID techniques (average of 100 runs).

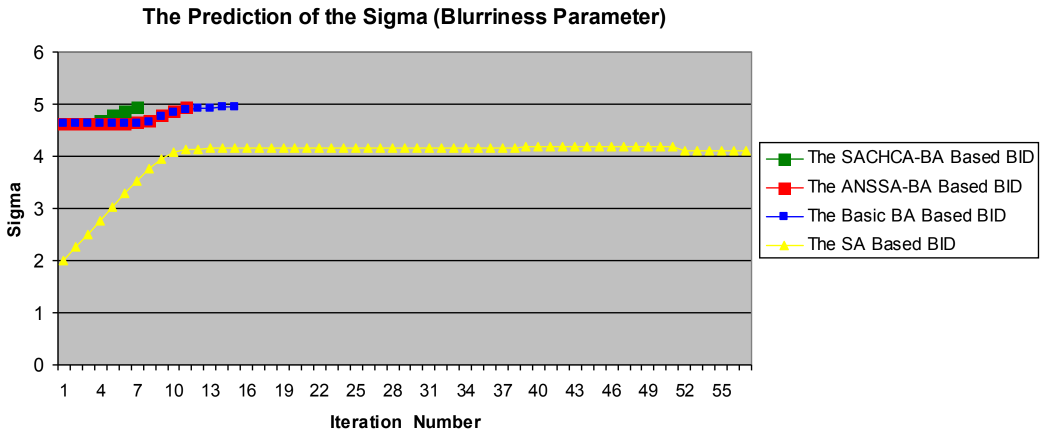


Figure 5.12 Blurriness parameter for the blurred-whitened image of Cameraman as determined by SA based BID and three BA-based BID techniques (average of 100 runs).

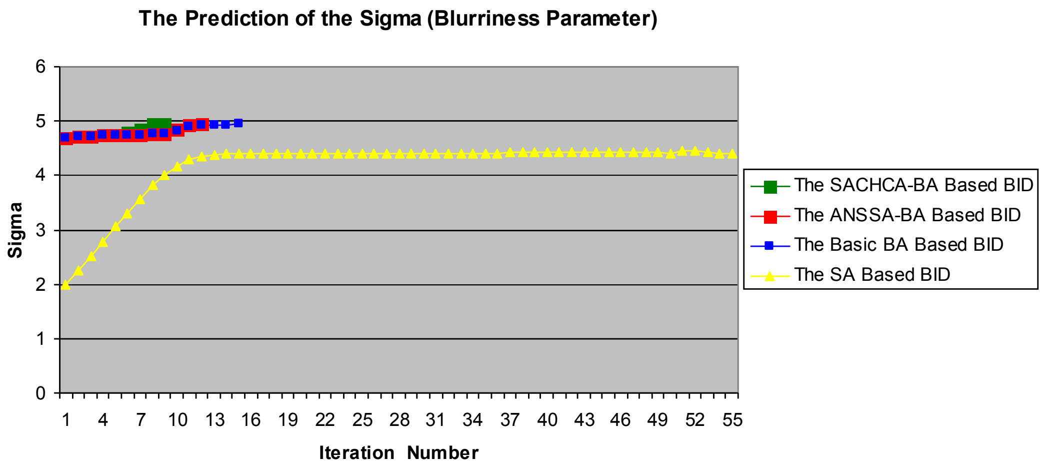


Figure 5.13 Blurriness parameter for the blurred-whitened image of Boats as determined by the SA based BID and three BA-based BID techniques (average of 100 runs).

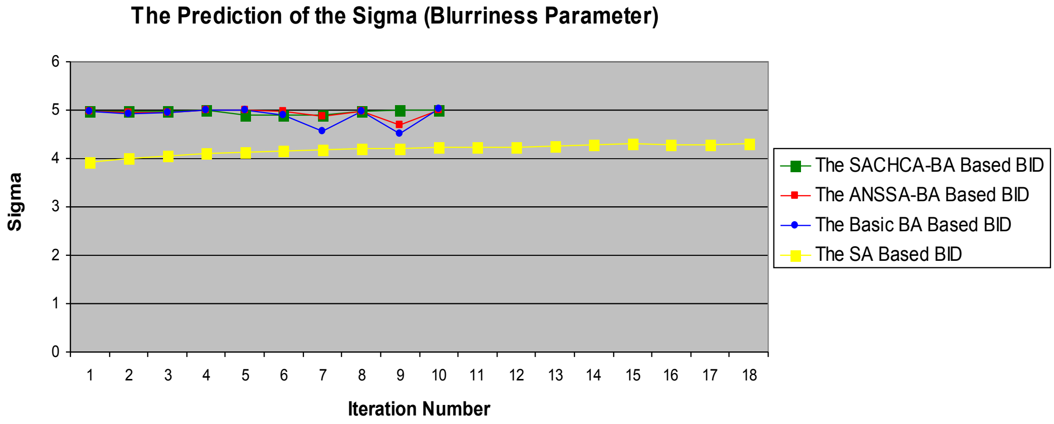


Figure 5.14 Blurriness parameter for the blurred-whitened image of Copper sample as determined by the SA based BID and three BA -based BID techniques (average of 100 runs).

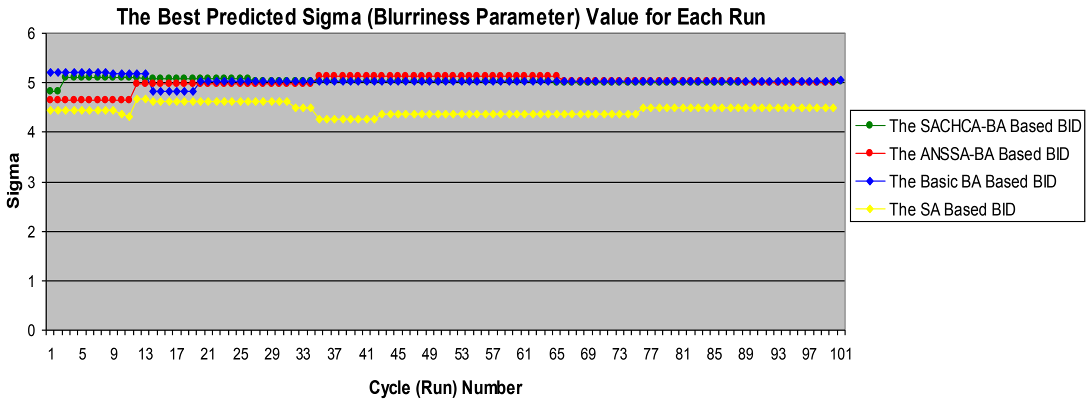


Figure 5.15 Best approximations to sigma for the blurred-whitened image of Lena as determined by the SA based BID and three BA-based BID techniques (average of 100 runs).

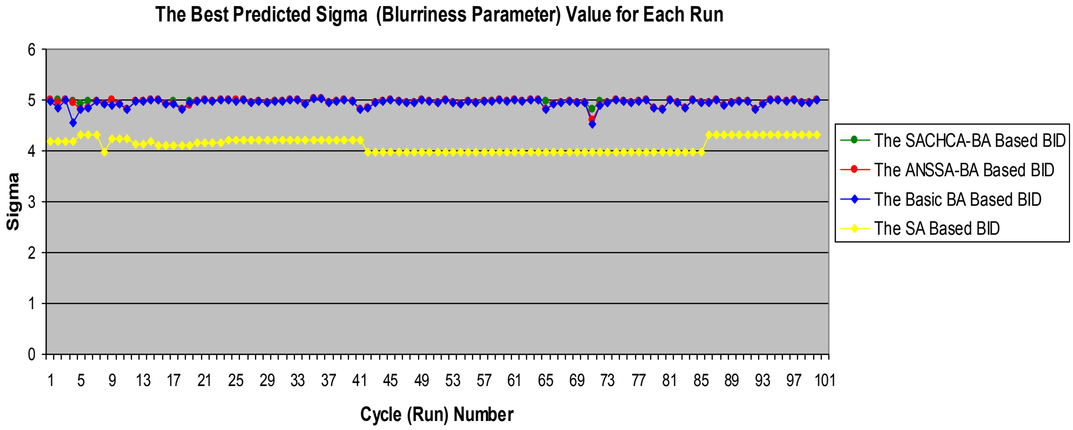


Figure 5.16 Best approximations to sigma for the blurred-whitened image of Cameraman as determined by the SA based BID and three BA-based BID techniques (average of 100 runs).

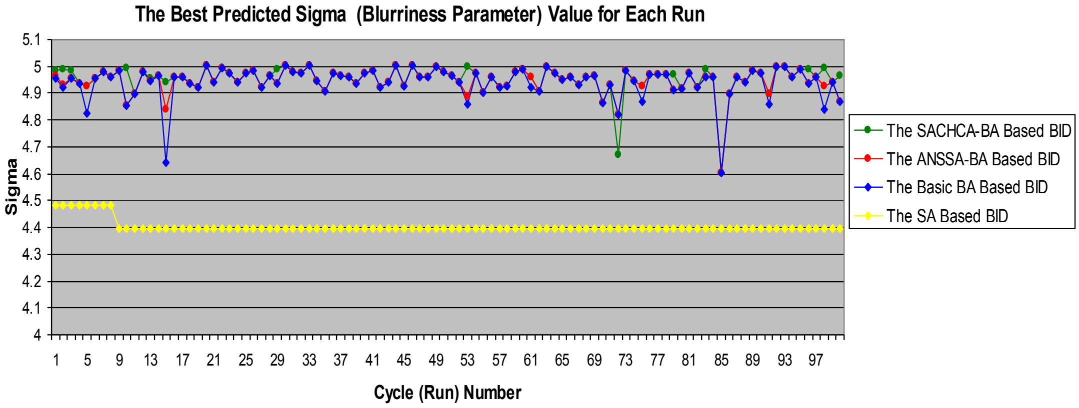


Figure 5.17 Best approximations to sigma for the blurred-whitened image of Boats as determined by the SA based BID and three BA-based BID techniques (average of 100 runs).

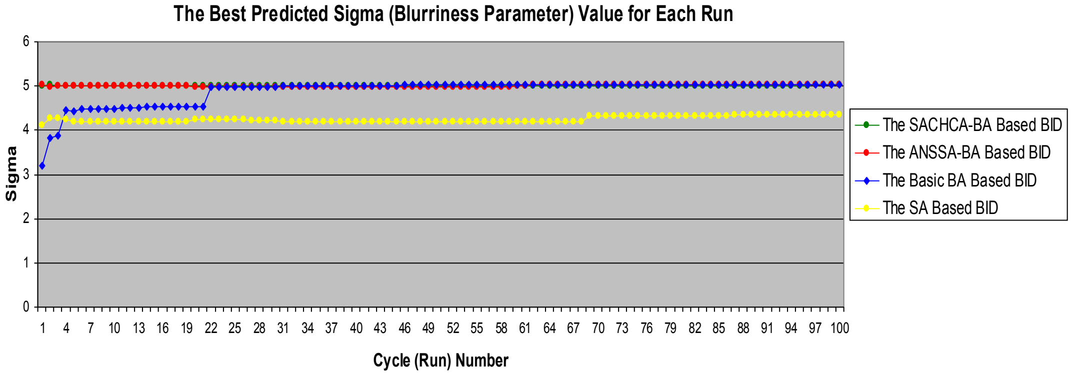


Figure 5.18 Best approximations to sigma for the blurred-whitened image of Copper sample as determined by the SA based BID and three BA-based BID techniques (average of 100 runs).

Table 5.2 Synopsis of best fitness results for 100 runs with SACHCA BA, ANSSA BA, basic BA and SA based-BID for images of Lena, Boats, Cameraman and Copper sample.

Test Images	SACHCA BA Based BID		ANSSA BA Based-BID		Basic BA Based-BID		SA Based-BID	
	Mean	Standard Deviation	Mean	Standard Deviation	Mean	Standard Deviation	Mean	Standard Deviation
Lena	5.0030	0.0465	5.0060	0.1384	5.0484	0.0777	4.4527	0.1070
Boats	4.9540	0.0453	4.9472	0.0503	4.9413	0.0614	4.404	0.0228
Cameraman	4.9572	0.0503	4.9840	0.0633	4.9408	0.0803	4.1146	0.1304
Copper Sample	5.0001	0.0084	5.0106	0.0217	4.8819	0.3027	4.2484	0.0635

Table 5.3 RMS error of each technique for the images of Lena, Cameraman, Boats and Copper sample.

Test Images	RMS Error of SACHCA Based BA-BID	RMS Error of ANSSA Based BA-BID	RMS Error of Basic Based BA-BID	RMS Error of SA Based-BID
Lena	0.0554	0.1378	0.1912	0.5575
Cameraman	0.0658	0.0817	0.0995	0.8949
Boats	0.0710	0.0728	0.0846	0.5965
Copper sample	0.0084	0.0240	0.3235	0.7543

According to Table 5.2, the BA based-BID techniques performed well on entire images compare to the SA based BID. In addition that the RMS error of the SACHCA-based improved BA-based BID technique was less than for any of the other techniques for all four images, according to the Table 5.3. The SA-based BID technique performed worst in the sense that it gave the largest RMS error of any technique for all four images. The ANSSA Based BA-BID generally outperformed the Basic Based BA-BID.

Using the best values for σ by the obtained different algorithms (Figure 5.14 to 5.17) the t-test was used to determine significant differences. The alpha values obtained from statistical significance tests are given in Table 5.4.

Table 5.4 Statistical significance of difference between best values of σ from each technique for given images.

Image	Techniques	ANSSA BA Based-BID	Basic BA Based-BID	SA Based-BID
Lena	SACHCA BA Based BID	0.0939	0.0471	0.0000
	ANSSA BA Based BID		0.0078	0.0000
	Basic BA Based BID			0.0000
	SA Based BID			
Cameraman	SACHCA BA Based BID	0.2872	0.0865	0.0000
	ANSSA BA Based BID		0.4834	0.0000
	Basic BA Based BID			0.0000
	SA Based BID			
Boats	SACHCA BA Based BID	0.3584	0.1252	0.0000
	ANSSA BA Based BID		0.4658	0.0000
	Basic BA Based BID			0.0000
	SA Based BID			
Copper sample	SACHCA BA Based BID	0.0001	0.0002	0.0000
	ANSSA BA Based BID		0.0001	0.0000
	Basic BA Based BID			0.0000
	SA Based BID			

Table 5.4 clearly shows that the best values for each of the three BA based BID algorithms are significantly different from, and more accurate than, the SA based BID algorithm at the 99% confidence level. For each of the four images the apparent ranking of the three BA based BID algorithms is consistently in the order (from best to worst) SACHCA, ANSSA and Basic BA.

However, application of the t-test shows the differences are not always significant at the 95% confidence level. For example with the image of Lena we see that the SACHCA and ANSSA are both significantly better than the basic BA based BID algorithm but not significantly different from each other.

For the image of Cameraman, Table 5.4 shows there is no significant difference in the best values between the three BA based BID algorithms.

For the image of Boats, Table 5.4 again shows there is no significant difference in the best values between the three BA based BID algorithms.

Finally for the image of the copper sample, Table 5.4 shows that the SACHCA gives significantly more accurate results than the ANSSA which, in turn, gives significantly more accuare results than the Basic BA BID technique.

It can thus conclude that the SACHCA based BA Bid technique consistently gave significantly more accurate results than the SA based BID, and where there was any significant difference with the other BA based techniques gave the best results. The BA based appears to be the best. Accordingly, it is concluded that a BA-based BID technique has a better capability for the SEM image.

5.5. A Novel Depth Calculation Using the BA-Based BID Technique for SEM

Depth computation is one of the most important tasks for SEMs. In this section, BA-based BID techniques are used to calculate the blurred parameter of a set of SEM images which were captured at different levels of depth and blurring according to the focused image plane of an SEM. The predicted results are related to depth.

The technique is based on determining the blurriness parameter of the different distance levels of the SEM images. Image construction in SEMs is given in Figure 5.19 (Nicolls, 1995). The predicted blurriness parameter is related with the distance according to Equations 5.4 and 5.5 and so a new depth metric can be defined according to the blurriness parameter. The technique is based on computing the blurriness parameter with an ICA-based optimised BID technique, and the optimisation step is carried out with the BA.

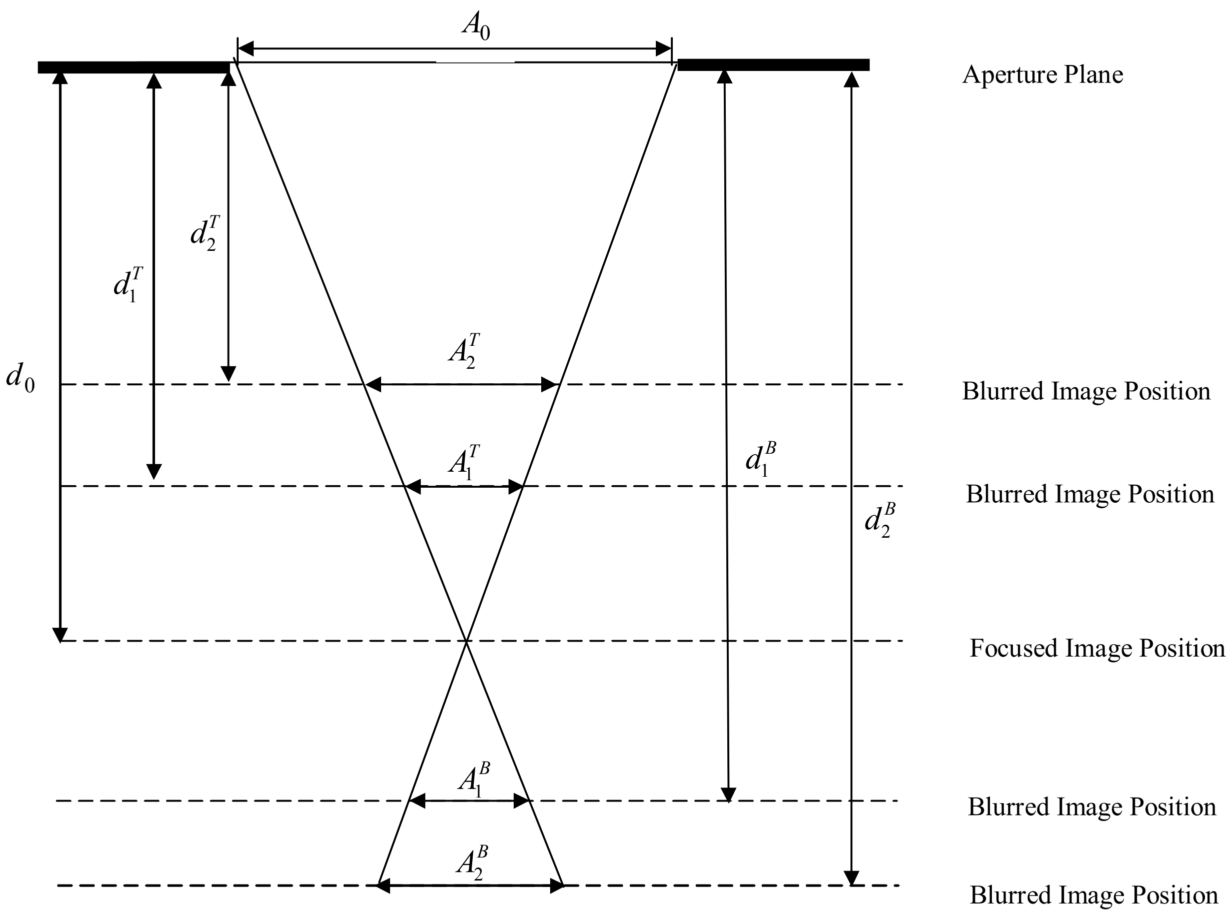


Figure 5.19 Schematic of image construction with SEM (Nicolls, 1995).

According to Figure 5.19, a blurred image is captured if the distance between aperture and specimen differs from d_0 (distance of the focused image plane). Using similar triangles, a relationship between the blurriness diameter of the image and depth is derived as in Equation 5.4.

$$\frac{d_0}{A_0} = \frac{d_0 - d_1^T}{A_1^T} = \frac{d_0 - d_2^T}{A_2^T} = \frac{d_1^B - d_0}{A_1^B} = \frac{d_2^B - d_0}{A_2^B} \tag{5.4}$$

where d_1^T, d_2^T, d_1^B and d_2^B are distances from the aperture of the SEM to the surface of the specimen; A_1^T, A_2^T, A_1^B and A_2^B are the blurriness diameters.

Aslantas (1997) proposed the blurriness parameter as proportional to the blurriness radius, as given in Equation 5.5:

$$R = k\sigma \tag{5.5}$$

where R is the blurriness radius, k is a constant and σ is the blurriness parameter.

If k is known, the predicted value of σ can be substituted into Equation 5.4 and the distance from aperture to the surface of the specimen can be measured based on Equation 5.4. The constant value can be measured using a calibration process.

According to Figure 5.19, the blurriness diameters are symmetrically placed both sides ("up" and "down") of the focused image plane. To overcome the ambiguity of which side is "up" and which is "down", the specimen can be moved to a new position and a new blurriness parameter computed. Using the new position and the new blurriness diameter, the correct side can be defined and the depth can be found.

5.6. Experimental Results for the Novel Depth Computation Technique Using the BA Based BID Techniques

This section presents the experimental results for the novel depth computation technique based on the BA-based BID technique. Initially, a set image was captured with an SEM at different depth levels and the blurriness parameters were calculated with the proposed BA-based BID technique and the results were related to the depth.

The proposed technique was implemented on a SEM model 1500XB. For the experiments, a copper sample was used, which contains a cylindrical shaped trench. All the experiments were carried out with the same SEM system parameters: Magnification = 135kX, Working Distance = 5 mm, Aperture Size = 30 µm, EHT (Voltage) = 10 kV, Noise Reduction = Line Integration, Current Mode = High Current and Imaging Mode = Secondary Electron Detector. The experiments began with setting the specimen stage to the focused image position and capturing the focused image, as shown in Figure 5.1d. Then the stage position of the focused image was moved up by 0.2 mm and the first blurred image was captured. This process was repeated 10 times in a series of incremental upward movements of 0.2 mm. When the last image was captured above the focused image plane the stage was moved back to the focused image position. Then the stage was moved down by 0.2 mm and the blurred image was captured.

This process was repeated 10 times in a series of incremental downward movements of 0.2 mm. The captured images are given in Appendices C1, and C.2.

According to Equation 5.5, the blurriness parameter is related to the blurriness diameter. The value of the constant k value was determined with a calibration process in which the stage was moved upwards by 0.2 mm, and the blurriness diameter calculated based on Equation 5.4. This was found to be 1.2 μm with 30 μm aperture size. The blurriness parameter, σ, of the PSF was computed with the proposed BA-based BID technique, and found to be 0.4105. The standard deviation results and blurriness radius were substituted into Equation 5.5 and k was found to be 1.4618. The relationship between the distance, expected blurriness radius and the expected blurriness parameter are given in Table 5.5.

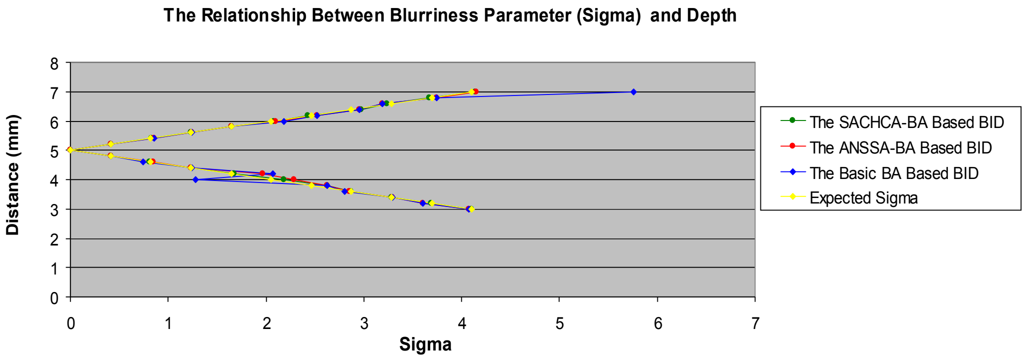
Table 5.5 The expected blurriness parameter and radius with regard to the distance between aperture and specimen.

Distance from aperture to specimen (mm)	Expected Blurriness Radius(μm)	Expected Blurriness Parameter (σ)
3.0	6.0	4.1045
3.2	5.4	3.6941
3.4	4.8	3.2836
3.6	4.2	2.8732
3.8	3.6	2.4627
4.0	3.0	2.0523
4.2	2.4	1.6418
4.4	1.8	1.2314
4.6	1.2	0.8209
4.8	0.6	0.4105
5.0	0.0	0.0000
5.2	0.6	0.4105
5.4	1.2	0.8209
5.6	1.8	1.2314
5.8	2.4	1.6418
6.0	3.0	2.0523
6.2	3.6	2.4627
6.4	4.2	2.8732
6.6	4.8	3.2836
6.8	5.4	3.6941
7.0	6.0	4.1045

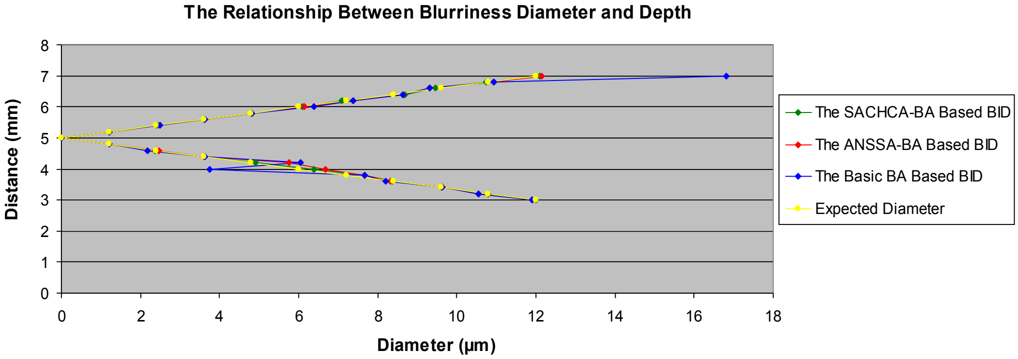
The three versions of BA based BID techniques were used for the SEM image. The relationship between the blurriness parameter and distance from focused image plane to specimen are shown in Figure 5.20a and the results of the blurriness diameter and distance from focused image plane to specimen are given in Figure 5.20b.

The accuracy of the BA-based BID techniques was measured using RMS error, see Table 5.6 which shows the performance of the SACHCA-based improved BA-based BID technique was the better than either the ANSSA-based BA BID technique or the Basic BA - BID technique. In addition that, SACHCA-based improved BA, ANSSA-based BA BID and basic BA BID matched %85.7, %66.7 and %52.3 of the given 21 depth level.

According to Table5.6 the performance of the ANSSA-based BA BID technique is not better than SACHCA-based BA BID, but better than the basic BA BID.



a)



b)

Figure 5.20 a) Relation between blurriness parameter and distance between aperture and specimen, and b) Relationship between blurriness diameter and distance between aperture and specimen.

Table 5.6 The RMS error results of the proposed techniques for the sigma and depth relationship.

Technique	Percentage(%) of Correct Estimated Distances	RMS Error
The Basic BA - BID	52.3%	0.4127
The ANSSA BA - BID	66.7%	0.1033
The SACHCA BA - BID	85.7%	0.0530

5.7. Summary

This chapter has covered prediction of the blurriness parameter using BA-based BID techniques, found a robust and effective BID technique combined with the BA and used it for depth determination.

Three different types of BA-based BID techniques were considered and assessed. The first was the basic BA; the second was an ANSSA strategy-based improved BA and the third and last was a SACHCA-based improved BA.

The three proposed BA-based and one SA-based BID techniques were tested on three benchmark images and an image of a copper sample captured with an SEM. The results obtained from the BA-based BID techniques were compared with each other and those obtained from the SA-based BID technique. The SA-based algorithm performed significantly less well than the others and was eliminated.

The three BA-based techniques were used on a set of SEM images captured at different distances from the focused image plane to predict the blurriness parameter. The predicted blurriness parameters were then used to determine distance from focused image plane to specimen in SEM. Thus a novel depth computation technique was obtained based on the BA-based optimised BID.

The relative performance of the BA-based BID techniques was assessed in this chapter and it was found that the SACHCA BA-BID algorithm was consistently the most accurate.

Chapter 6

CONCLUSION

This chapter summarises the contributions and conclusions of this study. It also provides suggestions for further research.

6.1. Contributions

First, this work investigated the depth computation techniques for SEM. The first investigation was based on a pre-processed DFAF technique. The processing stage is image correction. During the image acquisition stage with SEM, the captured images are affected by distortion, translation and rotation. In this study, captured images were corrected using the Phase Correlation Log-Polar Transformation (PCLPT) technique. Even though the PCLPT technique is a strong image correction technique, due to sub-pixel level interpolation errors, it also needs pre-processing. In this study, the pre processing stage was introduced based on image filtering for images captured with SEM. Then the DFAF technique was applied to the corrected SEM image to find the sharpness levels. Finally, the sharpness values were related to corresponding distances.

Second, this research investigated neighbourhood search in the BA and focused on improving the BA by utilising adaptive neighbourhood sizes and site abandonment (ANSSA). The proposed algorithm was tested on benchmark functions and an optimisation problem. The results of the improved BA were compared with the basic BA. The performance of the new algorithm was statistically tested with the t-test.

Third, this research investigated the Slope Angle Computation and Hill Climbing Algorithm (SACHCA)-based hybrid BA. Slope Angle Computation (SAC) is based on computing the slope of each best location. The Hill Climbing Algorithm (HCA) was utilised to find promising sites, when the slope of any patch is closer to zero with respect to a defined threshold. The proposed algorithm was tested on selected benchmark functions and an optimisation problem. The results of the proposed algorithm were compared with the results of the basic BA and the ANSSA strategy-based improved BA. The performance of the algorithm was statistically measured with the t-test.

Fourth, the blind image deconvolution (BID) technique was introduced. The technique was developed with the BA-based optimised non-gaussianity analysis. The non-gaussianity analysis was based on the kurtosis of the given images. The PSF, the blurriness parameter (sigma) of the PSF and the original image were then recovered. The proposed algorithm was tested on images of Lena, Boats, Cameraman and a copper sample captured by a SEM. Then, the proposed algorithm was utilised on a set of images captured with SEM at different depths.

The proposed algorithm was compared with a Simulated Annealing (SA)-based BID technique. The performance of the proposed algorithm was statistically tested with the t-test. Finally, the predicted blurriness parameters were utilised for the depth calculation process, where the blurriness parameter of each different depth level of the image was assigned to the related distance from the focused image plane.

6.2. Conclusions

In conclusion, all objectives stated in Chapter 1 have been met.

An image filtering-based pre-processed PCLPT technique was developed (**Objective 1**). This method gave good performance compared to the case without pre-processing.

The proposed pre-processed PCLPT technique was utilised on the image of a copper sample captured by an SEM and it corrected the given images (**Objective 2**). The performance of the pre-processed algorithm was acceptable in the case of the Gaussian, Median and iterative blind deconvolution-based pre-processing stage. In the case of the LoG, Prewitt and Sobel filtering-based pre-processing techniques, the proposed technique failed.

The DFFA technique was utilised on images corrected with the Gaussian filtering-based pre-processed PCLPT, Median filtering-based PCLPT and iterative BID technique-based pre-processed PCLPT. With respect to the sharpness function values, the sharpness results were successfully assigned to distances from the focused image plane (**Objective 3**).

The BA was developed with the ANSSA strategy and compared with the basic BA (**Objective 4**). The performance of the proposed algorithm on some benchmark functions was better than the basic BA.

The BA was improved with SACHCA and compared with the basic BA and the ANSSA strategy-based improved BA (**Objective 5**). The proposed algorithm performed better than the other two BA versions for the majority of the given benchmark functions.

The improved BAs and the basic BA were utilised in the BID technique (**Objective 6**).

Kurtosis-based non-gaussianity analysis was employed in the proposed BID technique and the optimum blurriness parameter of the image was searched by the BA with respect to the image kurtosis value. By varying the blurriness parameters of the PSF, the optimum value was found for each different depth level of the image. The results of the BA-based predicted blurriness values were compared with the results of the SA-based BID technique. Finally, the success rate of the proposed techniques was computed with the RMS error metric (**Objective 7**).

The predicted blurriness parameters were related to the different levels of distance according to the focused image (**Objective 8**).

6.3. Further research

First, this work investigated depth computation for SEM images. The first experiment was carried out with the DFAF technique on a corrected SEM image. Image correction was carried out with the pre-processed PCLPT technique. Six selected filters for the pre-processing step were tested. It is worth studying other image filters and testing their performance on the DFAF technique. Seven different sharpness functions were used. Furthermore, several functions have been proposed. It is worth utilising other sharpness functions to define the relationship between depth and sharpness value more accurately.

Second, the BA was improved with the ANSSA strategy to have a more adaptive and effective strategy during the search process. It was expected that adaptability would make the system more robust against chance. According to this, the adaptive neighbourhood size change was the start point. Further, it is worth improving the BA in respect of its adaptability on different aspects such as changing the number of bees adaptively, changing the best sites adaptively and so on.

Third, the second improvement of the BA was based on the SACHCA approach. This approach was inspired by the direction information in the waggle dance. The slope angle of the local search area was used to indicate the most promising search direction in this study. According to the slope angle, promising sites were found and searched by the HCA. If the slope angle was less than a threshold value then the HCA was utilised. In this study, the slope angle of each top-performing site was computed with the numerical first derivative of the fitness function. It is worth improving the slope angle computation with more accurate approaches. The second improvement for the BA is to focus on the other neighbourhood search techniques that can be utilised with the BA.

Fourth, the BA-based BID technique utilised kurtosis-based non-gaussianity analysis. It is worth studying other metrics with the BA-based BID technique. Furthermore, a novel depth computation technique based on the BA-based BID technique was introduced. It is worth studying the blurriness parameters to define a new depth computation technique using different imaging systems such as TEM and the interferometer.

REFERENCES

Adreasson, N., Evgrafov, A. and Patriksson, M. 2007. *An Introduction to Continuous Optimization: Foundations and Fundamental Algorithms*. Lund: Studentlitteratur.

Adorio, E. P. 2005. *MVF- Multivariate Test Functions Library in C for Unconstrained Global Optimization* [Online]. Available at: http://geocities.com/eadorio/mvf.pdf, [Accessed: 15 November 2011].

Al-Jabbouli, H. 2009. *Data Clustering Based on Bees and Trees*, PhD Thesis. Cardiff University.

Aslantas,V. 1997. *New Techniques for Determining Depth from Defocusing*. PhD Thesis, University of Wales.

Aslantas, V. and Kurban, R. 2009. A Comparison of Criterion Functions for Fusion of Multi- Focus Noisy Images. *Optics Communications*, *282*(16), pp. 3231-3242.

Ayers, G. R. and Dainty, J. C. 1988. Iterative Blind Deconvolution Method and Its Applications, *Optics Letters,* 13(7), pp. 547-549.

Back, T., Fogel, D. B. and Michalewicz, Z. 1997. *Handbook of Evolutionary Computation (Computational Intelligence Library).* New York: Taylor and Francis.

Batten, C. F. 2000. *Autofocusing and Astigmatism Correction in the Scanning Electron Microscope*. MPhil Thesis, University of Cambridge.

Belegundu, A. D. and Chandrupatla, T. R. 2011. *Optimization Concepts and Applications in Engineering.* 2nd ed. New York: Cambridge University Press.

Biggs, D. S. C. and Andrews, M. 1997. Acceleration of Iterative Image Restoration Algorithms. *Applied Optics,* 36(8), pp. 1766-1775.

Blondin, J. 2011. *Particle Swarm Optimization: A Tutorial* [Online]. Available at: http://cs.armstrong.edu/saad/csci8100/pso_tutorial.pdf, [Accessed: 28 November 2011].

Blume, M. 2007. *Multi-View Deconvolution of Medical Images*. MSc., Technical University Munich.

Blum, C. and Merkle, D. 2008. *Swarm Intelligence-Introduction and Applications.*, Berlin: Springer.

Bramier, M. F. and Banzhaf, W. 2006. *Linear Genetic Programming.*, New York: Springer.

Brenner, J. F., Dew, B. S., Horton, J. B., King, T., Neirath, P. W. and Seller, W. D. 1984. An Automated Microscope for Cytologic Research. *Journal of Histochemistry and Cytochemistry*, 24(1), pp. 100-111.

Brownlee, J. 2011. *Clever Algorithms: Nature-Inspired Programming Recipes*. [eBook version]. North Carolina: LuLu. Avaliable at: http://www.lulu.com/items/volume_71/11358000/11358156/1/print/10078410_paperback_20110125.pdf [Accessed: 09 December 2011].

Burger, W. and Burge, M. J. 2009. *Principles of Digital Image Processing*. London: Springer Verlag.

Campisi, P. and Egiazarian, K. 2007. *Blind Image Deconvolution*, Florida: CRC Press.

Chen, Y. W., Nakao, Z. and Arakaki, K. 1996. Blind Deconvolution Based on Genetic Algorithm, *ITC – CSCC'96*, pp. 71-75.

Crawley, M. J. 2005. *Statistics : An Introduction Using R*, West Sussex: Wiley.

Darwish, A. H. 2009. *Enhanced Bees Algorithm with Fuzzy Logic and Kalman Filtering.* PhD Thesis. Cardiff University.

Ding, N., Liu, Y., Jin, Y. and Zhu, M. 2010. *Image Registration on Log-Polar Transform and SIFT Features*. International Conference on Computational and Information Sciences, ICCIS 2010, Chengdu, 17-19 December 2010, pp. 749-752.

Dobes, M., Machala, L., and Furst, T. 2010. Blurred Image Restoration. *Digital Signal Processing*,20(6), pp. 1677-1686.

Dorigo, M., Maniezzo, V. and Colorni, A. 1996. Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)* 26(1), pp. 29-41.

Firestone, L., Cook, K., Culp, K., Talsania, N. and Preston K. 1991. Comparison of Autofocus Methods for Automated Microscopy. *Cytometry*, 12, pp. 195-206.

Floreano, D. and Mattiussi, C. 2008. *Bio-Inspired Artificial Intelligence-Theories, Methods, and Technologies.* London: MIT Press.

Foroosh, H., Zerubia, J. B. and Berthod, M. 2002. Extension of Phase Correlation to Sub-Pixel Registration. *IEEE Transaction on Image Processing*, 11(3), pp. 188-200.

Ghanbarzadeh, A. 2007. *The Bees algorithm. A novel optimisation tool.* PhD Thesis. Cardiff University.

Gendreau, M. and Potvin, J. Y. 2010. *Handbook of Metaheuristic.* 2nd ed. New York: Springer.

Gould, J. L. and Gould, C. G. 1988. *The Honey Bee*. New York: Scientific American Library.

Gonzalez, R. C., Woods, R. E. and Eddins, S. 2004. *Digital Image Processing Using MATLAB*. New Jersey: Pearson Prentice Hall.

Groen, F. C. A., Young, I. T. and Ligthart, G. 1985. A Comparison of Different Focus Functions for Use in Autofocus Algorithms. *Cytometry*, 6, pp. 81-91.

Grosan, C. and Abraham, A. 2011. *Intelligent Systems : A Modern Approach.*, Berlin: Springer.

Holmes, T. J., Bhattacharyya, S., Cooper, J. A., Hanzel, D., Krishnamurthi, V., Lin, W., Roysam, B., Szarowski, D. H. and Turner, J. N. 1995. Light Microscopic Images Reconstructed by Maximum Likelihood Deconvolution. In Pawley, J. B. ed. *Handbook of Biological Confocal Microscopy.* New York: Plenum, pp. 389-402.

Hong, G. and Zhang, Y. 2007.*Combination of Feature-Based and Area-Based Image Registration Technique for High Resolution Remote Sensing Image*. IEEE Transaction on Geoscience and Remote Sensing Symposium, IGARSS 2007, Barcelona, 23-28 July 2007, pp. 377-380.

Huang, Z. 2008. *Chapter 6. Behavioral Communications: 6.2. The Waggle Dance.* Available at: http http://photo.bees.net/biology/ch6/dance2.html [Accessed: 11th February 2012].

Hyvarinen, A. 1998. New Approximations of Differential Entropy for Independent Component Analysis and Projection Pursuit. in *Advances in Neural Information Processing Systems*, 10, pp. 273-279.

Hyvarinen, A. 1999. Survey on Independent Component Analysis. *Neural Computing Surveys,* 2, pp.94-128.

Hyvarinen A., Karhunen, J. and Oja, E. 2001. *Independent Component Analysis*, New York: John Wiley and Sons.

Iba, H., Paul, T. K. and Hasegawa, Y. 2010. *Applied Genetic Programming and Machine Learning.* Florida: Taylor and Francis.

Jarvis, R. A. 1976. Focus Optimisation Criteria for Computer Image Processing. *Microscope*, 24(2), pp. 163-180.

Jayaraman, S., Esakkirajan, S. and Veerakumar, T. 2009, *Digital Image Processing,* New Delhi: Tata McGraw Hill.

Kennedy, J. and Eberhart, R. 1995. Particle Swarm Optimization. *Proceedings of IEEE International Conference on Neural Networks IV* pp: 1942-1948.

Koc, E. 2010. *The Bees Algorithm Theory, Improvements and Applications*. PhD Thesis. Cardiff University.

Koppen, M., Abraham, A. and Schaefer, G. 2011. *Intelligent Computational Optimization in Engineering Techniques and Applications.* Berlin: Springer.

Koza, J. R. 1998. *Genetic Programming on the Programming of Computers by Means of Natural Selection.* 6th ed. London: The MIT Press.

Koziel, S. and Yang, X. S. 2011. *Computational Optimization, Methods and Algorithms.* Berlin: Springer.

Kristoffersen, T. K. 2007. *Stochastic programming with applications to power systems.* PhD Thesis, University of Aarhus.

Krotkov, E. P. 1987. Focusing. *International Journal of Computer Vision*, 1(3), pp. 223 – 237.

Kundur, D. and Hatzinakos, D. 1996. Blind Image Deconvolution. *IEEE Signal Processing Magazine.* 13(6), pp. 43-64.

Lam, E. Y. and Goodman, J. W. 2000. Iterative Statistical Approach to Blind Image Deconvolution, *J. Opt. Soc. Am. A,* 17(7), pp. 1177-1184.

Lamm, E. and Unger, R. 2011. *Biological Computation*. Florida: CRC Press.

Lane, R. G. and Bates, R. H. 1987. Automatic multidimensional deconvolution. *Journal of the Optical Society of America*, A(4), pp. 180-188.

Law, N. F. and Lane, R. G. 1996. Blind Deconvolution Using Least Squares Minimisation. *Optics Communications*, 128(4 – 6), pp. 341 – 352.

Lee, J. Y. 2010. *Multi-Objective Optimisation Using the Bees Algorithm.* PhD Thesis. Cardiff University.

Levente, K. 2003. *5. Mellekletek (keretrendszer, tesztkepek)* [Online]. Available at: http://vision.vein.hu/~kla/kepfeld/mellekletek.html, [Accessed: 12 August 2011].

Li, L. and Liu, F. 2011. *Group Search Optimisation for Application in Structural Design.* Berlin: Springer.

Li, J. and Rhinehart, R. R. 1998. Heuristic Random Optimisation. *Computers and Chemical Engineering* 22(3), pp. 427-444.

Liu, X., Li, M., Zhang, H. and Wang, D. 2003. Bayesian Motion Blur Identification Using Blur Priori. *International Conference on Image Processing (ICIP) 2003 Proceedings.* 3(2), pp. 957-960.

Liu, J. G. and Mason, P. J. 2009. *Essential Image Processing and GIS for Remote Sensing.* West Sussex: Wiley-Blackwell.

Madden, F. N., Godfrey, K. R., Chappell, M. J., Hovorka, R. and Bates, R. A. 1996. A comparison of Six Deconvolution Technique. *Journal of Pharmacokinetics and Biopharmaceutics.* 24(3), pp. 283-299.

Matungka, R., Zheng, Y. F. and Ewing, R. L. 2009. Image Registration Using Adaptive Polar Transform. *IEEE Transaction on Image Processing*, 18(10), pp. 2340-2354.

McCallum, B. C. 1990. Blind Deconvolution by Simulated Annealing, *Optics Communications,* 75(2), pp. 101-105.

Mendelsohn, M. L. and Mayall, B. H. 1972. Computer-Oriented Analysis of Human Chromosomes. *Computers in Biology and Medicine*, 2(2), pp.137-150.

Merchant, F. A. 2008. Three-Dimensional Imaging. In Wu, Q., Merchant, F. and Kenneth, R. C. ed. *Microscope Image Processing.* London : Academic Press, pp. 329-398.

Muller, R. A. and Buffington, A. 1974. Real-Time Correction of Atmospherically Degraded Telescope Images Through Image Sharpening. *Journal of Optical Society of America*, 64(9), pp. 1200-1203.

Nayar, S. K. and Nakagawa, Y. 1990. *Shape from Focus: An Effective Approach for Rough Surfaces.* Proceedings of the 1990 IEEE International Conference Robotics and Automation, Cincinnati, OH, USA, 13-18 May 1990, pp. 218-225.

Nicolls, F. 1995. *The Development of a Predictive Autofocus Algorithm using General Image Formation Model.* MSc Thesis. University of Cape Town.

Otri, S. 2011. *Improving the Bees Algorithm for Complex Optimisation Problems.* PhD Thesis. Cardiff University.

Oztireli, A. C. and Basdogan, C. 2008. A New Feature Based Method for Robust And Efficient Rigid – Body Registration of Overlapping Point Clouds. *The Visual Computer: International Journal of Computer Graphics,* 24(7), pp. 679-688.

Pan, W., Qin, K. and Chen, Y. 2009. An Adaptable-Multilayer Fractional Fourier Transform Approach for Image Registration, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31 (3), pp. 400-413.

Panigrahi, B. K., Lim, M. H. and Shi, Y. 2011. *Handbook of Swarm Intelligence-Concepts, Principles and Applications.* Berlin: Springer.

Pentland, A. P. 1987. A New Sense for Depth of Field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9 (4), pp. 523-531.

Pham, D. T. and Karaboga, D. 2000. *Intelligent optimisation techniques: Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks*. London: Springer-Verlag.

Pham, D. T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S. and Zaidi, M. 2005. *The Bees Algorithm*. Technical Report: MEC 0501. Cardiff: Manufacturing Engineering Centre, Cardiff University.

Pham, D. T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S. and Zaidi, M. 2006a. The Bees Algorithm: A Novel Tool for Complex Optimisation Problems. *Proc. 2nd Int. Virtual. Conf. on Intelligent Production Machines and Systems (IPROMS 2006)*. Oxford: Elsevier, pp: 454-459.

Pham, D. T., Otri, S., Ghanbarzadeh, A. and Koc, E. 2006b. Application of the Bees Algorithm to the Training of Learning Vector Quantisation Networks for Control Chart Pattern Recognition. *2nd IEEE International Conference on Information and Communication Technologies: From Theory to Applications. Damascus*, Syria, 2006, (1), pp. 1624-1629.

Pham, D. T., Koc, E., Ghanbarzadeh, A. and Otri, S. 2006c. Optimisation of the Weights of Multi-Layered Perceptrons Using the Bees Algorithm. *5th International Symposium on Intelligent Manufacturing Systems,* Sakarya, Turkey, 2006.

Pham, D. T., Ghanbarzadeh, A., Koc, E. and Otri. S., 2006d. Application of the Bees Algorithm to the training of radial basis function networks for control chart pattern recognition.*5th CIRP International Seminar on Intelligent Computation in Manufacturing Engineering, CIRP ICME,* 2006, Ischia, Italy.

Pham, D. T., Afify, A. A. and Koç, E. 2007a. Manufacturing Cell Formation Using the Bees Algorithm. *3rd International Virtual Conference on Intelligent Production Machines and Systems (IPROMS 2007)*, Whittles, Dunbeath, Scotland, 2007, pp. 523-528.

Pham, D. T., Koc, E., Lee, J. Y. and Phrueksanant, J. 2007b. Using the Bees Algorithm to Schedule Jobs for a Machine. *LAMDAMAP, 8th International Conference on Laser Metrology, CMM and Machine Tool Performance*, Cardiff, Euspen, UK, 2007, pp. 430-439.

Pham, D. T., Otri, S., Afify, A., Mahmuddin, M. and Al-Jabbouli, H. 2007c. Data Clustering Using the Bees Algorithm. *40th CIRP International Manufacturing Systems Seminar,* Liverpool, UK, 2007.

Pham, D. T., Castellani, M. and Fahmy, A. A. 2008. Learning the Inverse Kinematics of a Robot Manipulator Using the Bees Algorithm. *Proceedings of INDIN*, 2008, pp. 493-498.

Pham, D. T. and Castellani, M. 2009a. The Bees Algorithm: Modelling Foraging Behaviour to Solve Continuous Optimization Problems. *Proc. IMechE, Part C,* 223(12): 2919-2938.

Pham, D. T., Darwish, A. H. and Eldukhri, E. E. 2009b. Optimisation of a Fuzzy Logic Controller Using the Bees Algorithm. *International Journal of Computer Aided Engineering and Technology,* 1, pp. 250 – 264.

Pham, D.T., Packianther M.S. and Imaguliyev, A. Yuce, B. 2012a. *Early Neighbourhood Search and Efficiency Based Improved The Bees Algorithm*. IMS 2012 Conference 26-29 September 2012, Sakarya University,Turkey.

Pham, D.T., Packianther M.S. and Yuce, B. 2012b. *Image Registration and Translation Correction with Filtered Phase Correlation Method for Scanning Electron Microscope Images*. IMS 2012 Conference 26-29 September 2012, Sakarya University, Turkey.

Pohlheim, H. 2006. *GEATbx: Genetic and Evolutionary Algorithm Toolbox for use with MATLAB Documentation* [Online]. Available at: http://www.geatbx.com/docu/index.html, [Accessed: 15 November 2011].

Rao, S. S. 1996. *Engineering Optimization: Theory and Practice.* New York: John Wiley and Sons.

Raywaord-Smith, V. J., Osman, J. H., Reevees, C. R. and Smith, G. D. 1996. *Modern Heuristic Search Methods.* London: John Wiley and Sons.

Reddy, B. S. and Chatterji, B. N. 1996. An FFT-Based Technique for Translation, Rotation and Scale-Invariant Image. *IEEE Transaction on Image Processing*, 5(8), pp. 1266-1271.

Reeves, S. J. and Mersereau, R. M. 1992. Blur Identification by the Method of Generilised Cross Validation. *IEEE Transactions on Image Processing.* 1(3), pp. 301-311.

Rothlauf, F. 2011. *Design of Modern Heuristic Principles and Application*. Berlin: Springer.

Rosete-Suarez, A., Ochoa-Rodríguez, A. and Sebag, M. 1999. Automatic Graph Drawing and Stochastic Hill Climbing. *Proceedings of the Genetic and Evolutionary Computation Conference,* Florida, 13-17 July 1999. Morgan Kaufmann, pp. 1699-1706.

Rudnaya, M., Mattheij, R. M. M. and Maubach, J. M. L. 2010. Evaluating Sharpness Functions for Automated Scanning Electron Microscope. *Journal of Microscopy*, 240(1), pp. 38-49.

Rudnaya, M., Mattheij, R. M. M., Maubach, J. M. L. and Morsche, H. T. 2011. *Gradient-Based Sharpness Function*. Proceedings of the World Congress on Engineering, WCE 2011, London, UK, 6-8 July 2011, 1, pp. 301-306.

Rudnaya, M. and Ochshorn, R. 2011. Sharpness Functions for Computational Aestetics and Image Sublimation, *IAENG International Journal of Computer Science*, 38(4), pp. 359-367.

Rutkowski, L. 2008. *Computational Intelligence: Methods and Techniques.* Berlin: Springer.

Santos, A., Ortiz De Solorzano, C., Vaquero, J. J., Pena, J. M., Malpica, N. and Del Pozo, F. 1997. Evaluation of autofocus functions in molecular cytogenetic analysis, *Journal of Microscopy*, 188, pp. 264-272.

Schlag, J. F., Sanderson, A. C., Neuman C. P. and Wimberly, F. C. 1983. *Implementation of Automatic Focusing Algorithms for a Computer Vision System* *with Camera Control*. CMU-RI-TR-83-14, Robotics Institution, Carnegie Mellon University.

Schmidhuber, J. and Zhao, J. 1999. *Direct Policy Search and Uncertain Policy Evaluation.* Technical Report: SS-99-07. California: AAAI.

Seeley, T. D. 1995. *The Social Physiology of Honey Bee Colonies: The Wisdom of the Hive.* London: Harvard University Press.

Semmlow, J. L. 2004. *Biosignal and Biomedical Image Processing: MATLAB-Based Applications.* New York: Marcel Decker.

Sholedolu, M. O. 2009. *Nature-inspired Optimisation: Improvements to the Particle Swarm Optimisation Algorithm and the Bees Algorithm.* PhD Thesis. Cardiff University.

Snella, M. T. 2010. *Drift Correction for Scanning Electron Microscopy.* MEng Thesis, Massachusetts Institute of Technology.

Stone, J. V. 2004. *Independent Component Analysis: A Tutorial Introduction.* London: MIT Press.

Subbarao, M. and Surya, G. 1994. Depth from Defocus: A Spatial Domain Approach. *International Journal of Computer Vision,* 13(3), pp.271-294.

Subbarao, M. and Tyan, J. K. 1998. Selecting the Optimal Focus Measure for Autofocusing and Depth from Focus. *IEEE Transaction on Pattern Analysis and Machine Intelligence,* 20(8), pp. 864-870.

Sumathi, S. and Surekha, P. 2010. *Computational Intelligence Paradigms Theory and Applications Using MATLAB.* Florida: CRC Press.

Sun, Y. R. 2010. *Financial Integration of Nafta: Measurement and Analysis of the North American Financial Markets Convergence*. Master Thesis, University of Lethbridge.

Tenenbaum, J. M. 1970. *Accomodation in Computer Vision.* PhD Thesis, Stanford University.

Tyan, J. K. 1997. *Analysis and Application of Autofocusing and Three Dimensional Shape Recovery Techniques Based on Image Focus and Defocus.* PhD Thesis, SUNY.

Von Frisch, K. 1955. *The dancing bees: An account of the life and senses of the honey bee.* New York: Harcourt.

Wang, B., Kuruoglu, E. E. and Zhang, J. 2009. ICA by Maximising Non-stability. *Lecture Notes in Computer Science*, 5441(2009), pp. 179-186.

Weihong, W., Liegang, X., Jiancheng, L. and Xiaodong, H. 2012. Automatic and Precise Image Registration Based on Phase-Correlation Combined with Surface Fitting. *Applied Mathematics & Information Sciences*, 6(1), pp. 25-29.

Weise, T. 2009. *Global Optimization Algorithms: Theory and Application.* Available at: http://www.it-weise.de/projects/book.pdf, [Accessed: 28 November 2011].

Wolpert, D. H. and Macready, W. G. 1997. No Free Lunch Theorems for Optimization. *IEEE Transactions on Evolutionary Computation* 1(1), pp. 67-82.

Xiong, Y. and Shafer, S. A. 1993. *Depth from Focusing and Defocusing.* Technical Report: CMU-RI-TR-93-07. Pennsylvania: The Robotic Institute, Carnegie Mellon University.

Yap, K. H., Guan, L. and Liu, W. 2003. A Recursive Soft-decision Approach to Blind Image Deconvolution. *IEEE transaction on Signal Processing.* 51(2), pp. 515-526.

Yin, H. 2006. Blind Source Separation and Genetic Algorithm for Image Restoration, *International Conference on Advances in Space Technologies*, pp. 167-172.

Yousefi, S., Rahman, M., Kehtarnavaz, N. and Garnadia, M. 2011. A New Auto-Focus Sharpness Function for Digital Smart-Phone Cameras. *IEEE Transactions on Consumer Electronics,* 57(3), pp.475-476.

Yu. C. R. 2008. *A Study on Blind Image Restoration Based on Independent Component Analysis.* MSc Thesis, National Cheng Kung University.

Zelinka, I., Celikovsky, S., Richter, H. and Chen, G. 2010. *Evolutionary Algorithms and Chaotic Systems.* Berlin: Springer.

Zhigljavsky, A. and Zilinskas, A. 2008. *Stochastic Global Optimization.* New York: Springer.

Zitova, B. and Flusser, J. 2003. Image Registration Methods: A Survey. *Image and Vision Computing*, 21(2003), pp. 977-1000.

Zokai, S. and Wolberg, G. 2005. Image Registration Using Log-Polar Mappings for Recovery of Large-Scale Similarity and Projective Transformations. *IEEE Transcations on Image Processing,* 14(10), pp. 1422-1434.

Appendix A

No	Function Name	Explanation	Function	Parameters / Operators
1	Tenengrad Function	The function computes the gradient of the image, the sharpest image has max value (Schlag, et al., 1983 and Aslantas 1997)	$F = max\left\{ \sum_{x=1}^{N} \sum_{y=1}^{N} I(x,y)^2 \right\}$	$I(x,y) = \sqrt{I_x(x,y)^2 + I_y(x,y)^2}$; where $I(x,y)^2 > T$ $I_x(x,y) = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, I_y(x,y) = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$ (Sobel Operator is available to compute the derivations).
2	Brenner Function	Brenner function computes the sum of squared differences between pixels and two unit away pixels, the sharpest image has max value (Brenner, et. al, 1984 and Aslantas 1997).	$F = max\left\{ \sum_{x=1}^{N} \sum_{y=1}^{N} \left(I(x,y+2) - I(x,y)\right)^2 \right\}$	where $I(x,y)^2 > T$
3	Laplacian Function	Laplacian computes the sum squared Laplacian of the image and sharpest image has maximum value (Muller et al., 1974 and Krotkov, 1987).	$F =$ $max\left\{ \sum_{x=1}^{N} \sum_{x=1}^{N} \left(I(x,y+1) - 2I(x,y) - I(x,y-1)\right)^2 \right\}$	where $I(x,y)^2 > T$ The following operators are available. $F = \{ \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}, \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}, \begin{bmatrix} -1 & -4 & -1 \\ -4 & 20 & -4 \\ -1 & -4 & -1 \end{bmatrix} \}$
4	Sum Modified Laplacian Function	The function measures the Laplacian of the image; the sharpest image has max value (Nayar, and Nakagawa., 1990, and Aslantas 1997).	$F = max\left\{ \sum_{x=1}^{N} \sum_{y=1}^{N} \left(\left\lvert \frac{\partial^2 I(x,y)}{\partial x^2} \right\rvert + \left\lvert \frac{\partial^2 I(x,y)}{\partial y^2} \right\rvert \right) \right\}$	where $\left(\left\lvert \frac{\partial^2 I(x,y)}{\partial x^2} \right\rvert + \left\lvert \frac{\partial^2 I(x,y)}{\partial y^2} \right\rvert \right) \geq T$

Appendix A (Cont.)

5	Squared Gradient Function	The function computes the sum squared differences of the each pixel and a unit away neighbour pixels, the sharpest image has max value (Santos, et al., 1997).	$F = max\left\{ \sum_{x=1}^{N} \sum_{y=1}^{N} \left\| I(x,y+1) - I(x,y) \right\|^2 \right\}$	where $\left\| I(x,y+1) - I(x,y) \right\| \geq T$
6	Thresholded Absolute Gradient Function	The function is one of the basic gradient based sharpness function, the sharpest image has max function value (Groen et, al., 1985).	$F = max\left\{ \sum_{x=1}^{N} \sum_{y=1}^{N} \left\| I(x,y+1) - I(x,y) \right\| \right\}$	where $\left\| I(x,y+1) - I(x,y) \right\| \geq T$
7	Sum Modulus - Difference Function	The function computes the gradient of the region of interest which is the first order intensity differences between adjacent pixels (Jarvis, 1976 and Aslantas 1997).	$F = max\left\{ \sum_{x=1}^{N} \sum_{y=1}^{N} \left\| I(x,y) - I(x,y-1) \right\| \right\}$	
8	Range Function	The function is based on the searching for the maximum value of the difference between the maximum grey level and minimum grey level of the image Firestone et al., 1991)	$F = max\{ I_{max} - I_{min} \}$	where I_{max} and I_{min} are the maximum and minimum grey level, respectively.
9	Mendelsohn's and Mayall's Function	The function maximises the sum of difference between a selected threshold value and the pixels grey level (Mendelsohn and Mayall, 1972 and Aslantas, 1997).	$F = max\left\{ \sum_{I>T} Num(I) \right\}$	where $Num(I)$ is the number of the grey level of I, and T is the threshold and defined by the user, which may be selected as the mean value (Aslantas, 1997).

Appendix A (Cont.).

10	Mason’s and Green’s Function	This function maximises the sum of the difference between defined threshold and pixel grey level. The difference between the Mendelsohn’s and Mayall’s function and this function is the definition of the threshold (Aslantas, 1997).	$F = max\left\{ \sum_{I>T} Num(I)(I-T) \right\}$	where $T = \frac{\sum_{x=1}^{N}\sum_{y=1}^{N} I(x,y)\Delta^2_{xy}}{\sum_{x=1}^{N}\sum_{y=1}^{N} \Delta^2_{xy}}$ where Δ_{xy} is defined as the gradient of point (x,y).
11	Histogram Entropy Function	The function minimises the entropy of the histogram diagram (Aslantas, 1997; Santos, et al., 1997 and Aslantas and Kurban, 2009).	$F = min\left\{ -\sum_{I} P(I)\, ln(P(I)) \right\}$	where $P(I)$ is the probability of the grey level of I in histogram diagram. and $P(I) \neq 0$
12	Histogram of Local Variance Function	The function computes the local variance of the image histogram and the sharpest image has the minimum value (Aslantas, 1997).		
13	Variance Function	The variance function computes variance of images and the sharpest image has the maximum value (Aslantas, 1997 and Yousefi, et al., 2011).	$F = max\left\{ \frac{1}{N^2} \sum_{x=1}^{N}\sum_{y=1}^{N} \left[I(x,y) - \bar{I}\right]^2 \right\}$	
14	Normalised Variance Function	The function utilises a normalised variance based sharpness function. But (Aslantas, 1997) proposed the function as a standard deviation based function. The sharpest image has maximum value.	$F = max\left\{ \left(\frac{1}{\bar{I}^2\left(N^2-1\right)} \sum_{x=1}^{N}\sum_{y=1}^{N} \left[I(x,y) - \bar{I}\right]^2 \right)^{1/2} \right\}$	

Appendix A (Cont.).

15	Absolute Variation Function	The function is defined as a difference between pixel value and mean pixel value (Groen, et al., 1985). The sharpest image has maximum value.	$F = max\left\{ \frac{1}{N^2} \sum_{x=1}^{N} \sum_{y=1}^{N} \left\| I(x,y) - \bar{I} \right\| \right\}$	
16	Normalised Absolute Variation Function	The function measures the normalised difference between each pixel value and mean value (Groen, et al., 1985). The sharpest image has maximum function value.	$F = max\left\{ \frac{1}{\left(N^2 \bar{I}\right)} \sum_{x=1}^{N} \sum_{y=1}^{N} \left\| I(x,y) - \bar{I} \right\| \right\}$	
17	Thresholded Video Signal Content Function	The function computes the sum of the difference between a defined threshold and the pixel values (Aslantas, 1997).	$F = max\left\{ \sum_{x=1}^{N} \sum_{y=1}^{N} (I(x,y) \pm T) \right\}$	
18	Thresholded Video Signal Pixel Count	The function counts number of pixel, which has bigger pixel value then selected threshold value. The sharpest image has maximum value (Aslantas, 1997).	$F = max\left\{ \sum_{x}^{N} \sum_{y}^{N} (I(x,y) - T) \right\}$	$(I(x,y) - T) = \begin{cases} 1 & if\ (I(x,y) - T) \geq 0 \\ 0 & otherwise \end{cases}$
19	Signal Power Function	The function computes the sum of the squared grey level of the images (Aslantas, 1997). The sharpest image has maximum value.	$F = max\left\{ \sum_{x=1}^{N} \sum_{y=1}^{N} I(x,y)^2 \right\}$	

Appendix B

B.1. Gaussian Filter

The Gaussian Filter is one of the most widely used low pass filter (Gonzalez et al., 2004 and Semmlow, 2004). The filter removes the high-frequency components in the image with Gaussian. The Gaussian filter makes the image smoother (Merchant, 2008). The Gaussian function is given in Equation B.1:

$$h(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad \text{(B.1)}$$

where σ is the standard deviation.

B.2. Laplacian of Gaussian Filter

The Laplacian of a Gaussian-based filter (LoG) is another smoothening type filter, which utilises the Gaussian function. The LoG-based filter uses the second derivation of the Gaussian function, given in Equation B.2 (Gonzalez et al., 2004):

$$\nabla^2 h = h_{xx} + h_{yy} \qquad \Rightarrow \qquad \nabla^2 h = \frac{1}{2\pi\sigma^4}\left(\frac{x^2+y^2}{2} - 1\right) e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad \text{(B.2)}$$

where h_{xx} and h_{yy} are the second order partial derivative of h n x and y directions respectively.

In practice, the matrix form of the LoG is utilised instead of the continuous function form. The 5x5 matrix form of the LoG filter is given in Equation B.3:

$$\nabla^2 h = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 \\ -1 & -2 & 16 & -2 & -1 \\ 0 & -1 & -2 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix} \quad \text{(B.3)}$$

B.3. Prewitt Filter

The Prewitt filter is a linear spatial type filter which utilises the first order derivative. The Prewitt filter increases the image sharpness while enhancing the edges (Gonzalez et al., 2004). Thus it is called the edge detector. A first order gradient-based filter can be defined as a magnitude of gradient or simply the gradient of the image, given in Equation B.4:

$$\nabla h = \left(|h_x|^2 + |h_y|^2 \right)^{1/2}$$

(B.4)

where h_x and h_y are the first order partial derivatives of h in the direction x and y respectively.

The Prewitt filter utilises a 3x3 kernel matrix for the first order derivative in the x-direction and y-direction. The kernel matrixes are given in Equation B.5:

$$h_y = \frac{1}{3}\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}, \qquad h_x = \frac{1}{3}\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \quad \text{(B.5)}$$

For simplicity the gradient magnitude can be calculated approximately as in Equation B.6:

$$\nabla h \approx |h_x| + |h_y| \quad \text{(B.6)}$$

The Prewitt filter focuses on highlighting the edges.

B.4. Sobel Operator

The Sobel operator is also a first order derivative-based linear spatial filter. This filter utilises edge detection and highlights the edges (Burger and Burge, 2009). The Sobel operator utilises 3x3 kernel matrix to determine first order derivatives in the x and y directions. The kernel matrix is given in Equation B.7:

$$h_x = \frac{1}{4}\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \qquad h_y = \frac{1}{4}\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

(B.7)

Then the gradient magnitude ∇h, of the each pixel is calculated as in Equation B.6. The Sobel operator has the same features as the Prewitt filter. The performance of the Sobel operator-based pre-processed PCLPT technique was therefore expected to be the same as that of the Prewitt-based technique.

B.5. Median Filter

The Median filter is a non-linear type smoothing filter which is utilised for reducing the noise in the image (Gonzalez et al., 2004). The computation process is based on the replacing of the pixel intensity value with the median of the intensity value in the corresponding neighbourhood. This filter is effective when the noise pattern contains spike like components.

B.6. Iterative-Based Blind Deconvolution

The blind deconvolution algorithms have been improved to recover the original image and PSF without having information about any of them. There are several type of Blind image model have been developed. One type of the blind deconvolution is Iterative Blind deconvolution which has been developed for the MATLAB. The technique utilises Maximum Likelihood Estimation (MLE) to recover the original image and it's PSF. The algorithm was developed based on the study of (Holmes et al., 1995 and Biggs and Andrews, 1997). The algorithm steps are given below:

1. Initialise the PSF h_0 and template image $\hat{I}_0(x,y)$,
2. Predict the $h_k(x,y)$ as in Equation B.8:

$$h_{k+1}(x,y)=\left\{\left[\frac{g(x,y)}{h_k(x,y)*\hat{I}_k(x,y)}\right]*\hat{I}_k(-x,-y)\right\}h_k(x,y) \tag{B.8}$$

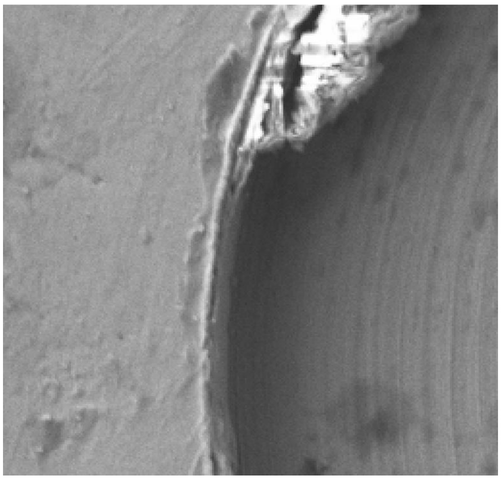
3. Predict the $\hat{I}_{k+1}(x,y)$ as in Equation B.9:

$$\hat{I}_{k+1}(x,y)=\left\{\left[\frac{g(x,y)}{\hat{I}_k(x,y)*h_k(x,y)}\right]*h_k(-x,-y)\right\}\hat{I}_k(x,y) \tag{B.9}$$

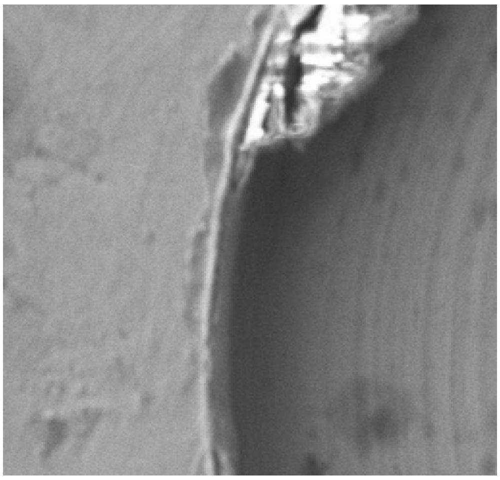
4. Result of the estimated image $\hat{I}_{k+1}(x,y)$ is compared to that of initial image,
5. The algorithm works until the termination condition is met.

Appendix C

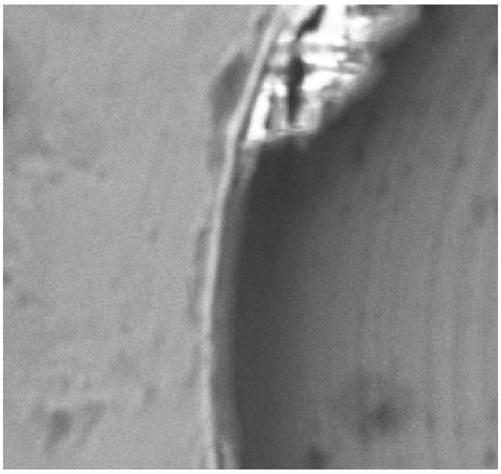
Figure C.1 a-j shows the images of a copper sample captured with SEM from the upper side of the focused plane by 0.2 incremental sizes. Figure C.2 a-j shows the images of a copper sample captured with SEM from the under side of the focused plane by 0.2 incremental sizes.



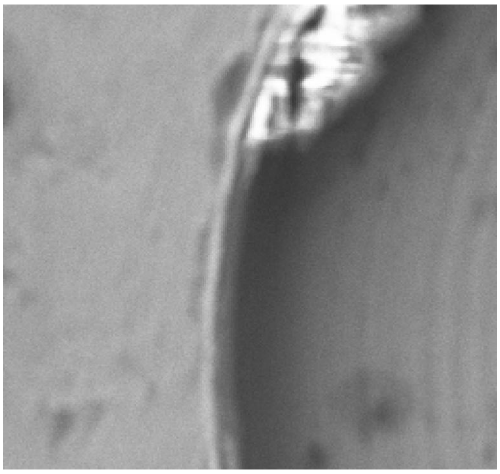
a) The image is captured 0.2mm above from the focused image.



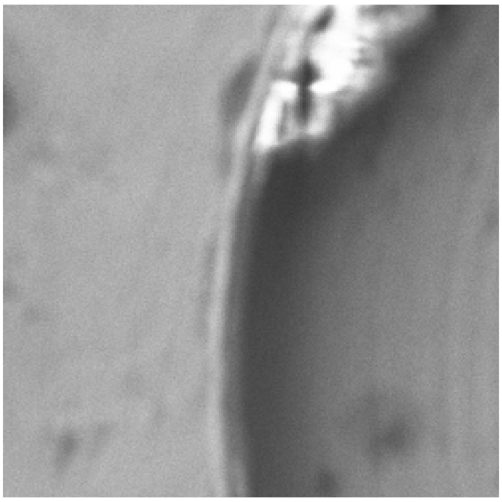
b) The image is captured 0.4mm above from the focused image.



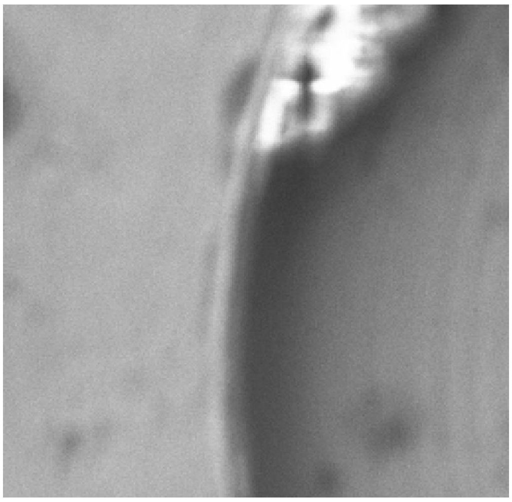
c) The image is captured 0.6mm above from the focused image.



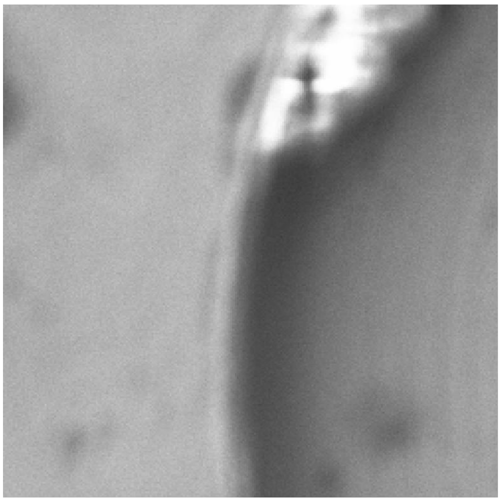
d) The image is captured 0.8mm above from the focused image.



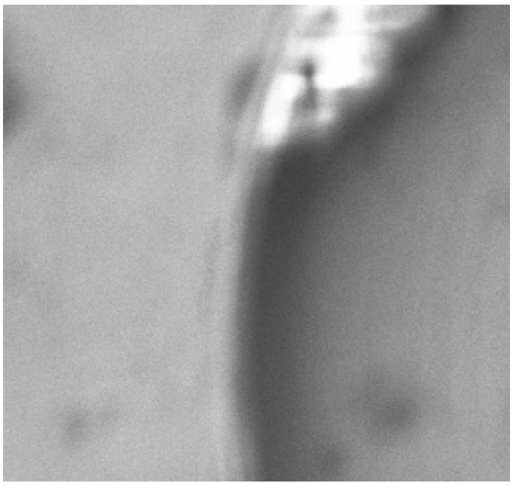
e) The image is captured 1mm above from the focused image.



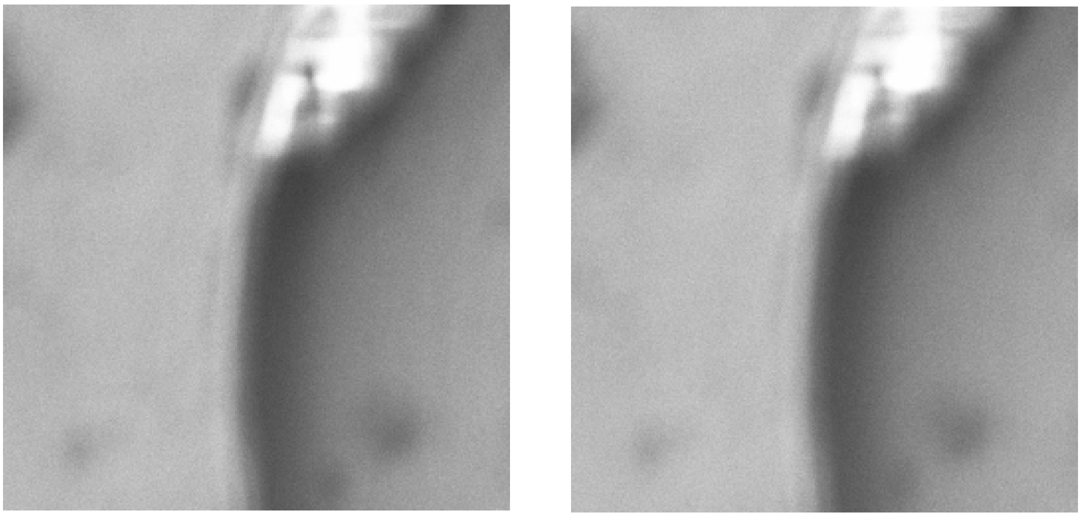
f) The image is captured 1.2mm above from the focused image.



g) The image is captured 1.4mm above from the focused image.



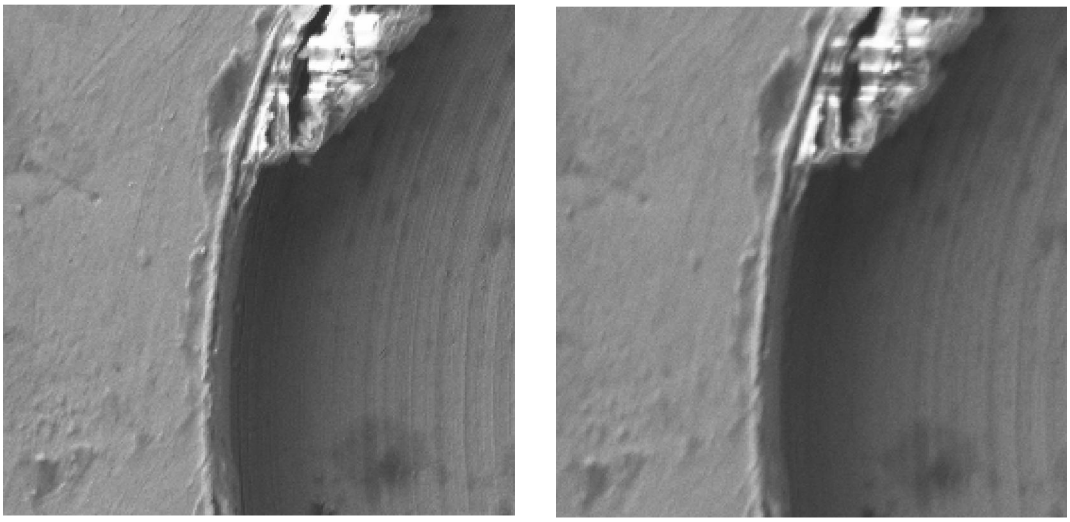
h) The image is captured 1.6mm above from the focused image.



i) The image is captured 1.8mm above from the focused image.

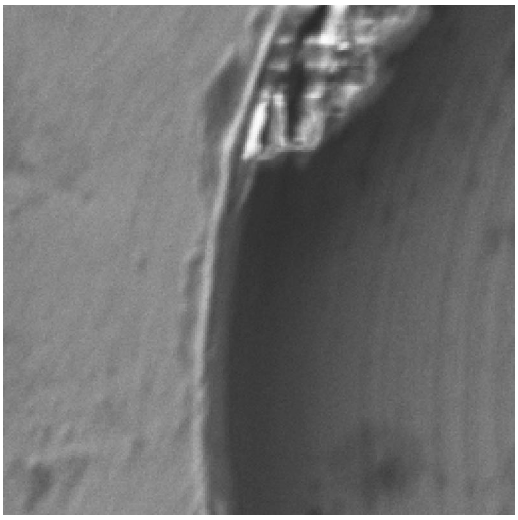
j) The image is captured 2mm above from the focused image.

Figure C.1 a-j The images were captured with SEM by 0.2 incremental size from the upper direction of the focused plane.

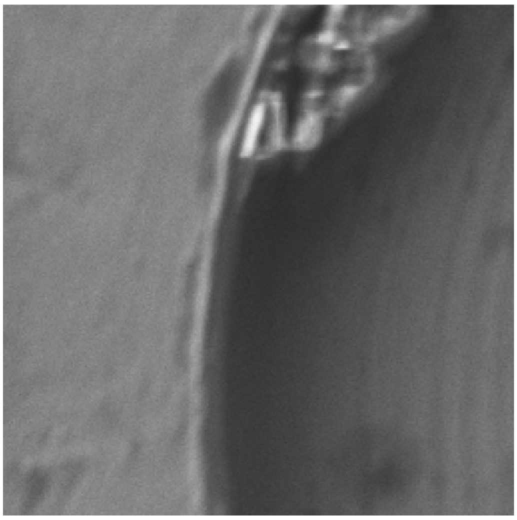


a) The image is captured 0.2mm below from the focused image.

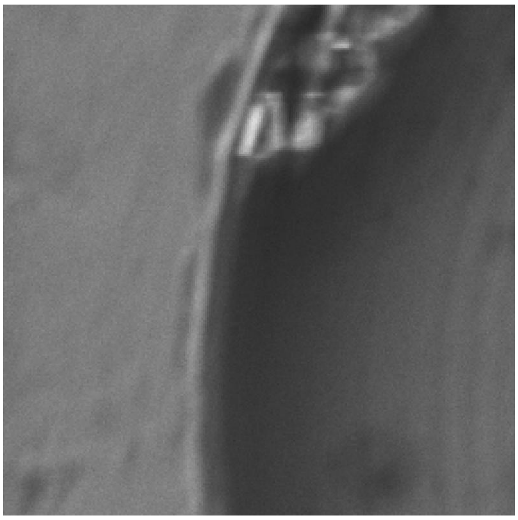
b) The image is captured 0.4mm below from the focused image.



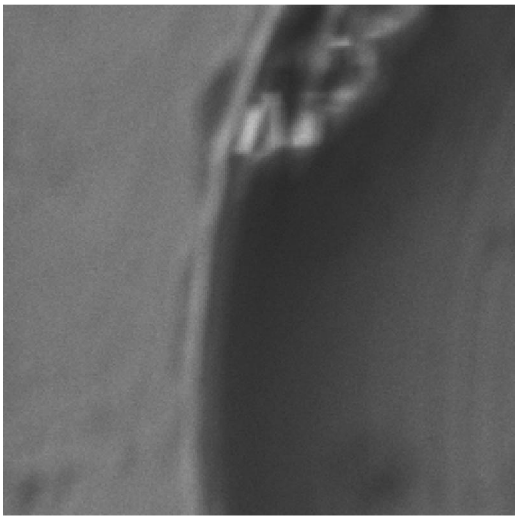
c) The image is captured 0.6mm below from the focused image.



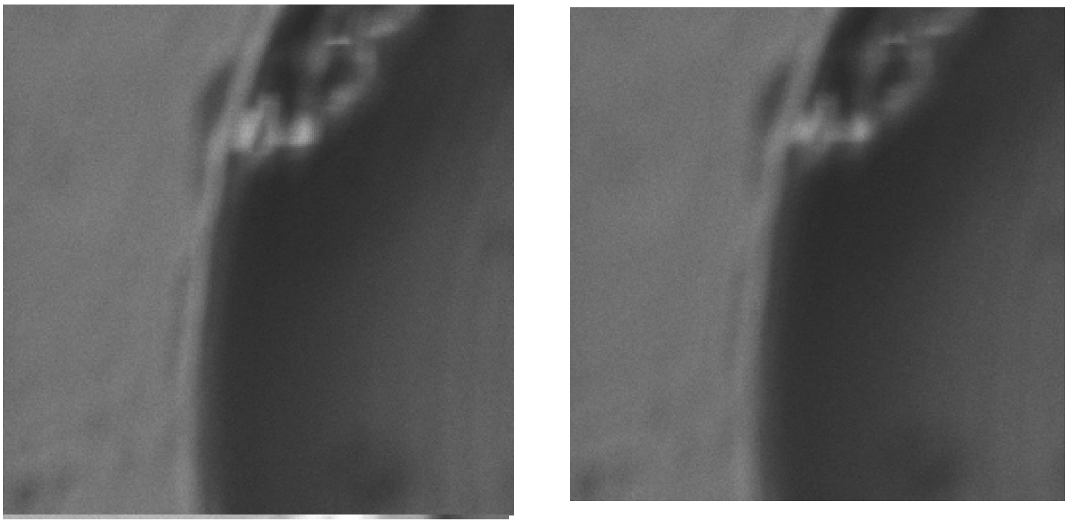
d) The image is captured 0.8mm below from the focused image.



e) The image is captured 1mm below from the focused image.

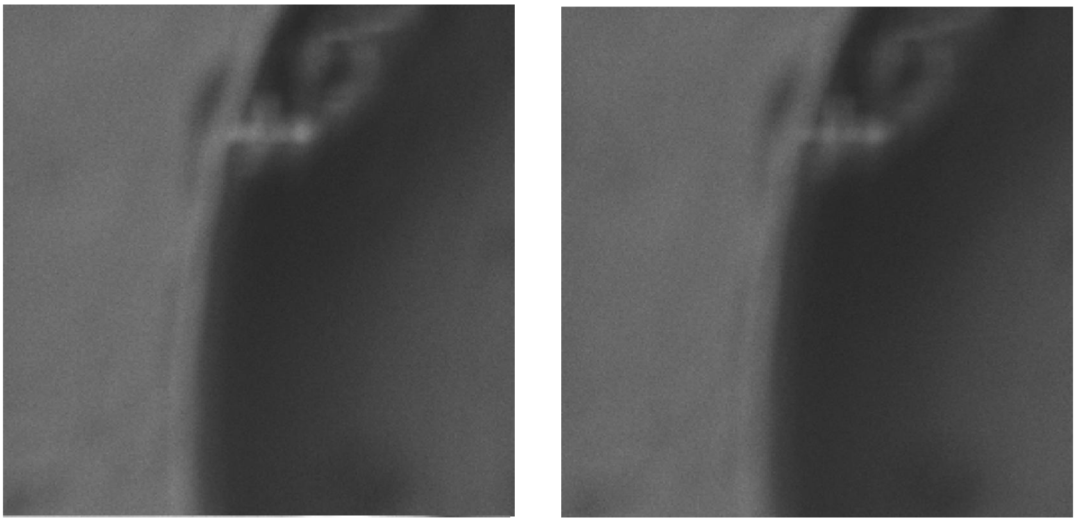


f) The image is captured 1.2mm below from the focused image.



g) The image is captured 1.4mm below than

h) The image is captured 1.6mm below from



i) The image is captured 1.8mm below than the focused image.

j) The image is captured 2mm below than the focused image.

Figure C.2 a-j The image was captured with SEM by 0.2 incremental size from the lower direction of the focused plane.

Appendix D

The Table D.1, D.2 and D.3 show the best fitness value of 100 runs for the Himmelblau function with the basic BA, the ANNSA strategy based improved BA and the improved BA based on the slope angle computation and HC algorithm respectively.

Table D.1 Best fitness values for 100 runs for Himmelblau function with the basic BA.

8.6417	9.7738	9.0988	9.9581	9.3068
9.6553	9.9978	9.7641	9.8115	9.9963
9.9288	9.8090	9.7891	9.8047	9.8952
9.8971	8.9007	9.4824	9.9425	9.7692
9.6014	9.8777	9.8833	8.3195	9.7340
9.8365	9.6062	9.6178	9.9781	9.9470
9.4210	9.9999	9.6766	9.7417	9.7086
9.5828	8.9268	8.8594	8.4171	8.9914
9.7052	9.9773	9.9887	9.9919	9.9269
9.9155	9.8787	9.6898	9.6790	8.4495
9.1936	9.9008	9.9696	9.5634	9.8942
9.8867	9.9912	9.8167	9.6941	8.9851
9.97512	9.9198	9.1940	9.6137	9.9893
9.9826	9.9806	9.6071	9.9201	8.8415
9.5572	9.9305	9.8272	9.9967	9.7755
8.8403	9.4309	9.7021	9.3806	9.6106
9.8307	9.9976	9.8193	9.9008	9.2591
9.3815	9.3515	9.7870	9.8807	9.5933
9.8609	9.4403	9.9212	9.8133	9.2199
9.9694	9.6647	9.9780	8.7745	9.6993

Table D.2 Best fitness value of 100 runs for Himmelblau function with improved BA with ANSSA strategy.

9.9596	9.9449	9.9087	9.9869	9.8899
9.8812	9.9758	9.8371	9.8836	9.7611
9.8387	9.4372	9.9988	9.9536	9.8876
9.85978	9.4467	9.9133	9.8675	9.9253
9.9302	9.8340	8.8576	9.8875	9.8567
9.3483	9.9320	9.6926	9.8973	9.9699
9.8568	9.9967	9.7089	8.9794	9.9609
8.4607	9.9641	9.7427	9.9070	9.8764
9.1988	9.8261	9.9566	9.6428	9.8377
9.9996	9.3646	9.7883	9.4486	9.9647
9.9727	9.8365	9.2090	9.5547	9.0461
9.9647	9.9406	9.8642	9.9867	9.8298
9.7550	9.9524	9.9722	9.7492	9.9751
8.9670	9.75043	9.5288	9.9270	9.9374
9.6950	9.4610	9.8316	9.8281	9.9457
9.9683	9.92248	9.9975	9.8813	9.5335
9.7910	9.8667	9.8501	9.7236	9.9931
9.9612	9.3949	9.9678	9.3060	9.4702
8.5931	9.8749	9.7437	9.1395	9.8132
9.9967	9.9472	9.5871	9.8765	9.9780

Table D.3 The best fitness value of 100 runs for Himmelblau function with the improved BA based on the slope angle computation and HC algorithm.

9.8275	9.9238	9.7072	9.9920	9.9429
9.9843	9.9664	9.8078	9.9930	9.9395
9.9775	9.8143	9.8159	9.9528	9.8186
9.9301	9.9609	9.9176	9.9063	9.8908
9.7649	9.9740	9.4463	9.8346	9.6003
9.9865	9.8860	9.8139	9.7630	9.8458
9.8637	9.9110	9.9008	9.6055	9.3061
9.9477	9.9833	9.3118	9.9829	9.6478
9.6268	9.9478	9.2858	9.7682	9.8418
9.5974	9.8097	9.8093	9.9328	9.9727
9.6071	9.9850	9.9613	9.8263	9.6180
9.7532	9.9908	9.9761	9.8947	9.9832
9.9776	9.8620	9.6576	9.8937	9.9779
9.3526	9.9724	9.7950	9.9427	9.8781
9.8207	9.8937	9.5935	9.9693	9.9378
9.9917	9.9613	9.9641	9.9239	9.8790
9.9234	9.7773	9.6330	9.6408	9.2060
9.6489	9.4254	9.9997	9.9902	9.8402
9.9529	9.6489	9.9053	9.9906	9.8110
9.9840	9.9539	9.7250	9.8121	9.6569