
Masters Theses

Student Theses and Dissertations

Spring 2014

Processing and feature analysis of atomic force microscopy images

Xiao Pan

Follow this and additional works at: https://scholarsmine.mst.edu/masters_theses



Part of the [Electrical and Computer Engineering Commons](#)

Department:

Recommended Citation

Pan, Xiao, "Processing and feature analysis of atomic force microscopy images" (2014). *Masters Theses*. 7268.

https://scholarsmine.mst.edu/masters_theses/7268

This thesis is brought to you by Scholars' Mine, a service of the Missouri S&T Library and Learning Resources. This work is protected by U. S. Copyright Law. Unauthorized use including reproduction for redistribution requires the permission of the copyright holder. For more information, please contact scholarsmine@mst.edu.

PROCESSING AND FEATURE ANALYSIS OF ATOMIC FORCE MICROSCOPY

IMAGES

by

XIAO PAN

A THESIS

Presented to the Faculty of the Graduate School of the
MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE IN
ELECTRICAL ENGINEERING

2014

Approved by

R. Joe Stanley, Advisor
Randy H. Moss
William V. Stoecker

© 2014

Xiao Pan

All Rights Reserved

ABSTRACT

Atomic force microscopy (AFM) is a versatile and powerful tool for imaging and measuring small-scale objects such as nanoparticles, single molecules, semiconductor devices and living cells. The basic operation of an AFM can be to utilize a sharp cantilever tip that interacts with the sample surface and senses the local force between the tip and sample surface. Based on the physical interaction between the AFM and the small-scale object for image acquisition, there can be a number of artifacts, including curvature distortion (bowing effects), high-frequency or low-frequency noise, which may not be easily recognized by users accustomed to conventional microscopy.

In this research, different image processing functions are designed to visualize AFM data, address different types of AFM artifacts problems and analyze features. Algorithms according to AFM image processing functions are presented. Analysis of AFM images acquired from silicon chips, which are provided by the Mechanical Engineering Department at Missouri University of Science and Technology, is displayed.

ACKNOWLEDGMENTS

I would like to appreciate my advisor, Dr. R. Joe Stanley, for all of his help, support and encouragement during my graduate studies. He has given me the opportunity to study and work in the Digital Image Processing field and be a research assistant in the AFM program. He has been a supportive advisor and mentor in my studies and research. His technical guidance is vital not only to the completion of my research projects but also to my real life. I would also like to express my thanks and gratitude to Dr. Randy H. Moss and Dr. William V. Stoecker for their time and efforts on my committee members.

I would also like to express appreciation to all my friends and colleagues that I have met and worked with, in particular Koyel Banerjee, Peng Guo and Lu Cheng. Thanks all for you spending time to share ideas with me. I would like to thank BeiBei Cheng for recommending me to Dr. Stanley, her advisor.

Most importantly, my heartfelt appreciation goes to my family members, for their ongoing support and encouragement in all of my endeavors. I am deeply grateful for my parents who always give me the best wishes. Their unconditional trust and infinite love help me to focus on the research which I am doing and accomplish what I have achieved so far.

TABLE OF CONTENTS

| | Page |
|---|------|
| ABSTRACT | iii |
| ACKNOWLEDGMENTS | iv |
| LIST OF ILLUSTRATIONS | vi |
| SECTION | |
| 1. INTRODUCTION | 1 |
| 2. METHODOLOGY | 7 |
| 2.1. BROWSE AND NORMALIZING IMAGE | 7 |
| 2.2. FLATTENING IMAGE | 9 |
| 2.3. ERASE LINE | 15 |
| 2.4. LINEAR INTERPOLATION | 18 |
| 2.5. EDGE DETECTION | 19 |
| 2.5.1. Roberts Operator | 20 |
| 2.5.2. Prewitt Operator | 20 |
| 2.5.3. Sobel Operator | 21 |
| 2.5.4. Canny Operator | 21 |
| 2.6. LOW-PASS FILTER & HIGH-PASS FILTER | 21 |
| 2.7. 3-DIMENSIONS | 22 |
| 2.8. PLOT INTENSITY | 23 |
| 2.9. COLORIZE | 24 |
| 3. RESULT AND DISCUSSION | 26 |
| 3.1. SILICON SURFACE 1 | 26 |
| 3.2. SILICON SURFACE 2 | 32 |
| 4. CONCLUSION | 42 |
| BIBLIOGRAPHY | 43 |
| VITA | 45 |

LIST OF ILLUSTRATIONS

| | Page |
|---|------|
| Figure 1.1. AFM block diagram. | 2 |
| Figure 1.2. Example of non-contact AFM mode and contact AFM mode. | 3 |
| Figure 1.3. Example of an AFM image with bowing effect. | 4 |
| Figure 1.4. Initial Interface of GUI for AFM application. | 5 |
| Figure 2.1. File selecting window of GUI for AFM application. | 8 |
| Figure 2.2. Example of an image displayed in the GUI application. | 9 |
| Figure 2.3. Signal $si(x)$, polynomial $pi(x)$ and raw data $yi(x)$ | 11 |
| Figure 2.4. Example showing how artificial streaks look like around the cells. | 12 |
| Figure 2.5 Flow chart for flattening image algorithm. | 14 |
| Figure 2.6. Comparison of horizontal leveling and vertical leveling. | 15 |
| Figure 2.7. Example of an AFM image processed by the median filter. | 16 |
| Figure 2.8. Histogram of an AFM row distribution. | 17 |
| Figure 2.9. Edge detection examples. | 19 |
| Figure 2.10. Viewpoints with Azimuth and Elevation. | 23 |
| Figure 2.11. Example of plotting intensity. | 24 |
| Figure 3.1. Illustration of raw AFM image and normalized image of silicon surface 1... 27 | 27 |
| Figure 3.2. Flattened image. | 28 |
| Figure 3.3. Comparison of line profile between no leveling image and leveling image. . 28 | 28 |
| Figure 3.4. AFM image by erase line and linear interpolation operation. | 29 |
| Figure 3.5. Silicon surface 1 AFM image results of low-pass filter and high-pass filter. 30 | 30 |
| Figure 3.6. Examples of a 3-dimensional AFM image with different viewpoints. | 31 |
| Figure 3.7. Example of silicon surface 1 AFM image in different color shapes. | 32 |
| Figure 3.8. Illustration of raw AFM image and normalized image for silicon surface 2. 33 | 33 |
| Figure 3.9. Flattened AFM image for silicon surface 2. | 33 |
| Figure 3.10. Comparison of line profile between uneven image and flattened image. | 34 |
| Figure 3.11. AFM images processed by different edge detection operators | 35 |
| Figure 3.12. Silicon surface 2 AFM image results of low-pass and high-pass filter. | 36 |
| Figure 3.13. Example of 3-D AFM image. | 37 |

| | |
|---|----|
| Figure 3.14. Example of AFM images in different colors..... | 38 |
| Figure 3.15. Demonstration of the result window. | 39 |
| Figure 3.16. Demonstration of ‘Save Image’ | 40 |
| Figure 3.17. Demonstration of ‘EXIT’ window. | 41 |

1. INTRODUCTION

Atomic force microscopy (AFM) is one of the foremost and powerful techniques to scan, image, measure and analyze surface structure at the nanoscale. An atomic force microscope is capable to acquire images with the arrangement of individual atoms or see the structure of individual molecules. Prior to AFMs, the scanning tunneling microscope, developed by Binnig and Rohrer in the early 1980s, was utilized for analyzing structures [1]. In 1986, the first atomic force microscope was invented by Binnig, Quate and Gerber [2]. AFMs became commercially available in 1989. Now AFMs have been utilized for structural analysis in various fields, including chemistry, biology, physics, materials science, nanotechnology and medicine. Three advantages facilitate AFM technology to be developed quickly for different problem domains, including: 1) producing images with 3-dimensional information; 2) scanning structures does not need to be done under vacuum or some other extreme environment (rather, AFMs can be operated in both air and liquid) [3]; 3) AFM does not need to be carried out under some certain conditions as mentioned in [2]; therefore, no special sample treatments are required which can result to alteration and destruction of AFM samples.

The main components of the AFM technique are the microscope stage, control electronics and a computer [4]. The microscope stage consists of the scanner (in AFM, also known as piezoelectric transducer); piezoelectric holder and a force sensor (also known as a force transducer) to hold and monitor the AFM tip. Usually, the sample is placed on the piezoelectric hold and the sensor is bonded with an extremely flexible cantilever which carries a very fine point. Meanwhile, an optical system of detection is utilized in order to measure the vertical deflections of the cantilever. Basically, the scanner moves the tip over the sample surface; the force transducer senses the force between the tip and the sample surface and then the feedback control information produced by the beam deflection from the force transducer will be transformed back to the piezoelectric scanner to maintain a fixed force between the tip and the sample. In theory, the AFM technique is a relatively simple instrument but a considerable amount of sophisticated engineering is needed to construct an integrated AFM with nanometer-scale

resolution. Figure 1.1 provides a sample of Atomic Force Microscope using optical system of detection.

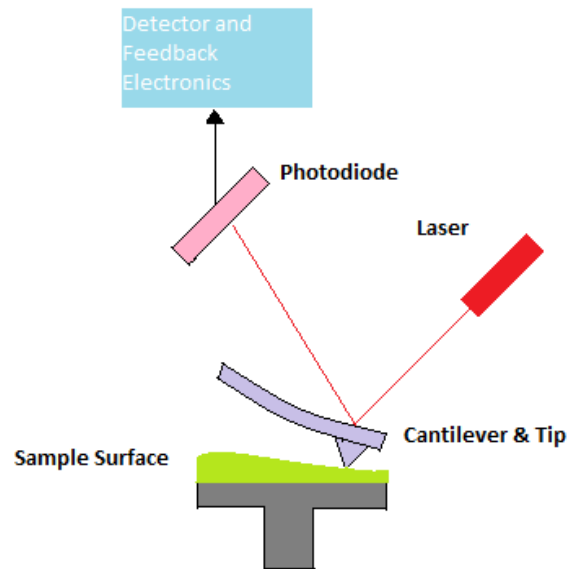


Figure 1.1. AFM block diagram.

AFM technique has three types of operation modes: contact mode, non-contact mode and tapping mode [5-6]. Non-contact AFM mode scans the sample by moving the sharp probe to its surface at a certain distance and the tip is oscillated at the resonance frequency [7]. Contact AFM mode gathers information by touching the sample's surface with a sharp mechanical probe and build up a map containing the height of the sample's surface. The tapping mode is between the contact and non-contact mode which keeps the tip close enough to the sample for a short-range force. Figure 1.2 below shows an example of non-contact AFM mode and contact AFM mode. In our research, we use the contact AFM operation mode to scan and get the information.

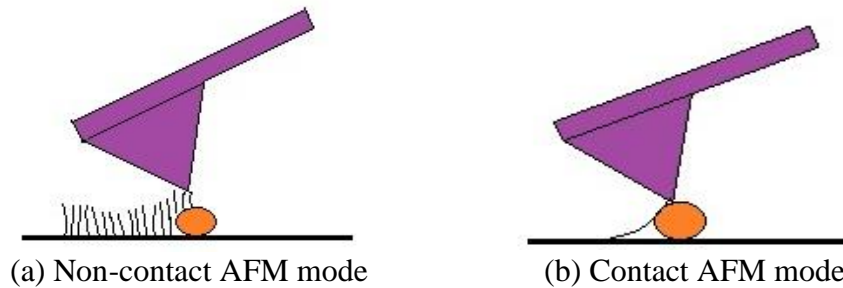


Figure 1.2. Example of non-contact AFM mode and contact AFM mode.

All measurement techniques and instruments used by scientists and engineers for AFM research development and quality control generate consequences that are prone to have artifacts. Different instruments and experimental situations can cause different types of artifacts, including unwanted high-frequency or low-frequency noise or influence of uneven background. Figure 1.3 provides an example of an AFM image with uneven background. Sometimes these artifacts are easily spotted, while sometimes so difficult. In the case that users know exactly what to search for and the source of the artifact, some artifacts can probably be detected and avoided. Even though a few artifacts are unavoidable, realizing their existence in an AFM image will also serve to prevent misinterpreting them as genuine image features, which means recognizing AFM image artifacts occupies a significant position for the AFM technique. Four primary sources of artifact in image measured with atomic force microscopes result to inaccurate AFM feature analysis, including [8]:

- Probes: AFM images are always affected by the geometrical shape of the tip. If the probe is small enough compared to the features, the probe-generated artifacts will be the minimum. The proper way to fix this problem is achieved by using the optimal probe for the application.
- Scanners: Scanners which move the probe in various directions are typically made by piezoelectric ceramic. Technically, scanners move the probe in very tiny distances. However, if a linear voltage is applied to piezoelectric ceramics, artifacts can be introduced. The best way to avoid this problem is to calibrate the scanner periodically following the appropriate manufacturer's instructions.

- Vibrations: Environment vibrations in the room, such as acoustic vibrations or floor vibrations, easily cause the probe in the AFM to vibrate and produce artifacts in an image. Typically, these vibrations will turn to oscillations in the AFM image.
- Image processing: AFM image processing is an indispensable part to process and display AFM data before viewing or analyzing an AFM image. The aim of all AFM image processing operations is to clarify the data obtained during the measurement. In other words, the core objective is to measure and observe AFM features that have been recorded with high accuracy.

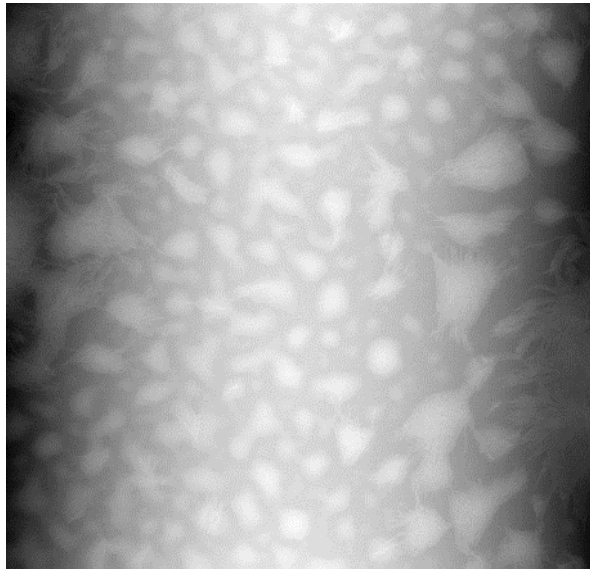


Figure 1.3. Example of an AFM image with bowing effect.

The main purpose of this thesis is to illustrate image processing methods or algorithms which can remove artifacts introduced into AFM images and analyze the processed AFM data; therefore, we developed an AFM image processing tool in MATLAB. In detail, a GUI (also known as graphical user interfaces or UI) interface is developed for this study using diverse functional buttons and slider-bars. The reason to apply a GUI is that it is made for users without any programming knowledge. Figure 1.4 shows the initial interface for the GUI in AFM application.

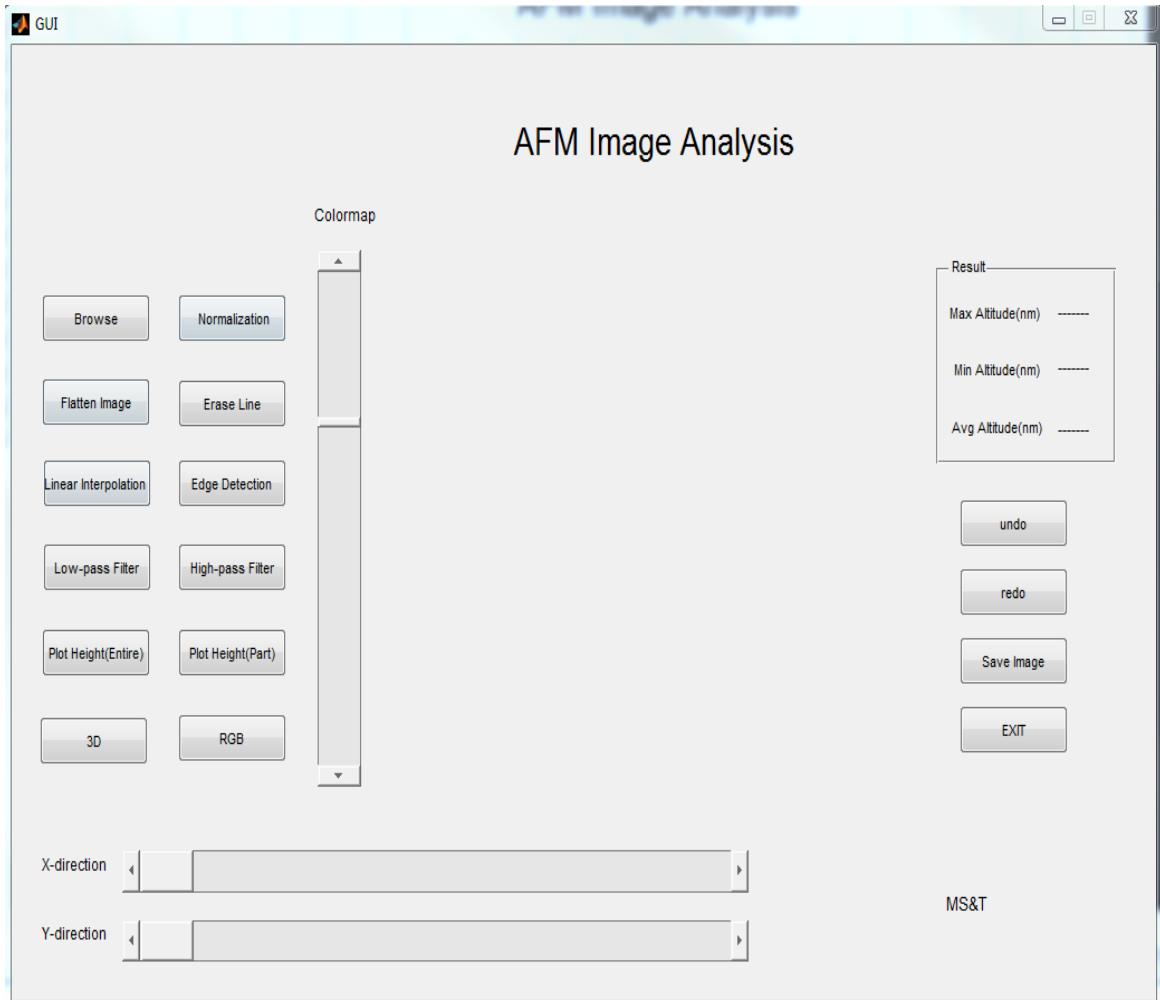


Figure 1.4. Initial Interface of GUI for AFM application.

This research was conducted with the Mechanical Engineering (Dr. Doug Bristow and his students Muthukumaran Loganathan, Alireza Toghraee, etc.) department, who designed the AFM instruments and provided experimental data from the surface of two silicon chips, which require further AFM image processing for enhancing image features and to address image artifacts.

The remainder of this thesis is organized jointly by follows: Section 2 introduces every button function in AFM GUI interface and algorithms executed on these buttons. Section 3 presents results of acquired AFM data which are operated by every function

button. Section 4 gives the conclusion of this thesis and provides the suggestion of future sphere.

2. METHODOLOGY

The goal of this research is to set up a series of image processing functions to facilitate the analysis of the shape and structure representative of nanoscale objects scanned using Atomic Force Microscopes (AFMs). A brief overview of how AFMs work to scan nanoscale (small) objects has been provided in Section 1 as background to highlight some of the artifacts that are commonly encountered and need to be taken into account in image analysis applications of those objects. Details on how to display, process and analyze these data are elaborated in the following sections.

In order to make an efficient and convenient integrated environment, a GUI (also known as graphical user interfaces or UIs) imaging application has been designed for this research. GUIs in MATLAB provide point-and-click control of software applications, eliminating the need to learn a language or type commands with the purpose to run the application. In this project, a GUI is accomplished by different functional buttons and slider-bars which can be observed in Figure 1.4. The GUI designed here contains the following options: Browse, Image Normalization, Flatten Image, Erase Line, Linear Interpolation, Edge Detection, Low-pass Filter, High-pass Filter, Plot Intensity (Entire), Plot Intensity (Part), 3D and RGB buttons; Colorize, x-direction as well as y-direction slide bars and result window. The following part of this section is followed by a description of every characteristic function on GUI and a detailed explanation of how these corresponding functions are realized.

2.1. BROWSE AND NORMALIZING IMAGE

Data files collected for objects from AFM systems vary widely in format from one document to another. Text format(-.txt) is used for the data files in this research. The .txt files contain floating point voltage values at each position to characterize an object. To display nanometer scale topographical information for the object, the initial step is to convert all data information in text format to an image format, whose intensity is exactly equal to the raw data from the text format file. The GUI developed in this research has a special self-contained MATLAB program which reveals a window for users to select .txt file from a specified path. The file selection window described above is depicted in Figure 2.1 below.

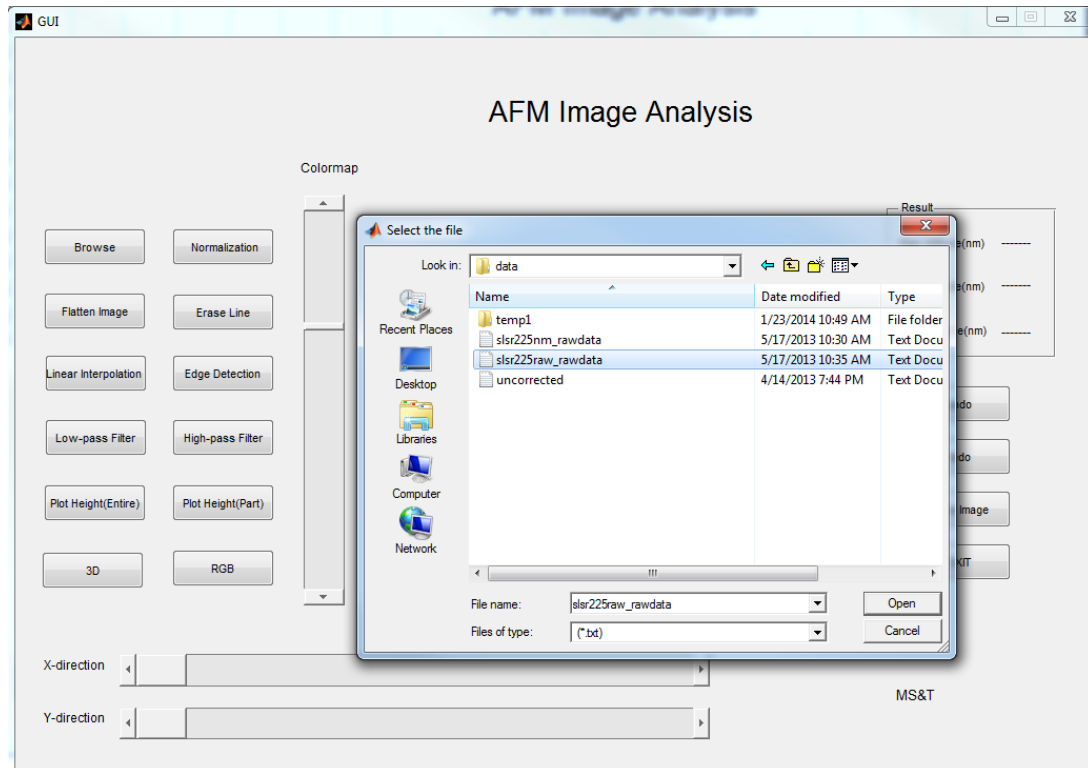


Figure 2.1. File selecting window of GUI for AFM application.

Once the original object data file is opened, the data file is transformed into a matrix format to provide an image representation of the object. In converting to the matrix format, the values at each matrix position (pixel value) are normalized using the following approach for visualization and analysis as an image. In our research, images are displayed using 8 bits, which provide image intensity values to be in the range from 0 to 255 [9]. Given an image f , the normalization approach that provides the range of values between 0 and some designated maximum value (such as 255) if as follows:

- Create an image, f_m , whose minimum value is 0;

$$f_m = f - \min(f) \quad (1)$$

- Create a scaled image, f_s , whose values are in the range $[0, K]$;

$$f_s = K \left[\frac{f_m}{\max(f_m)} \right] \quad (2)$$

As mentioned earlier, working with 8-bit images as well as setting $K=255$, we can get a normalized image whose intensities span the full 8-bit scale from 0 to 255. The above window shows that users can click open button if they have found the desired file. An image will be shown in the center of the GUI interface. Figure 2.2 is an example of an image displayed in the center.

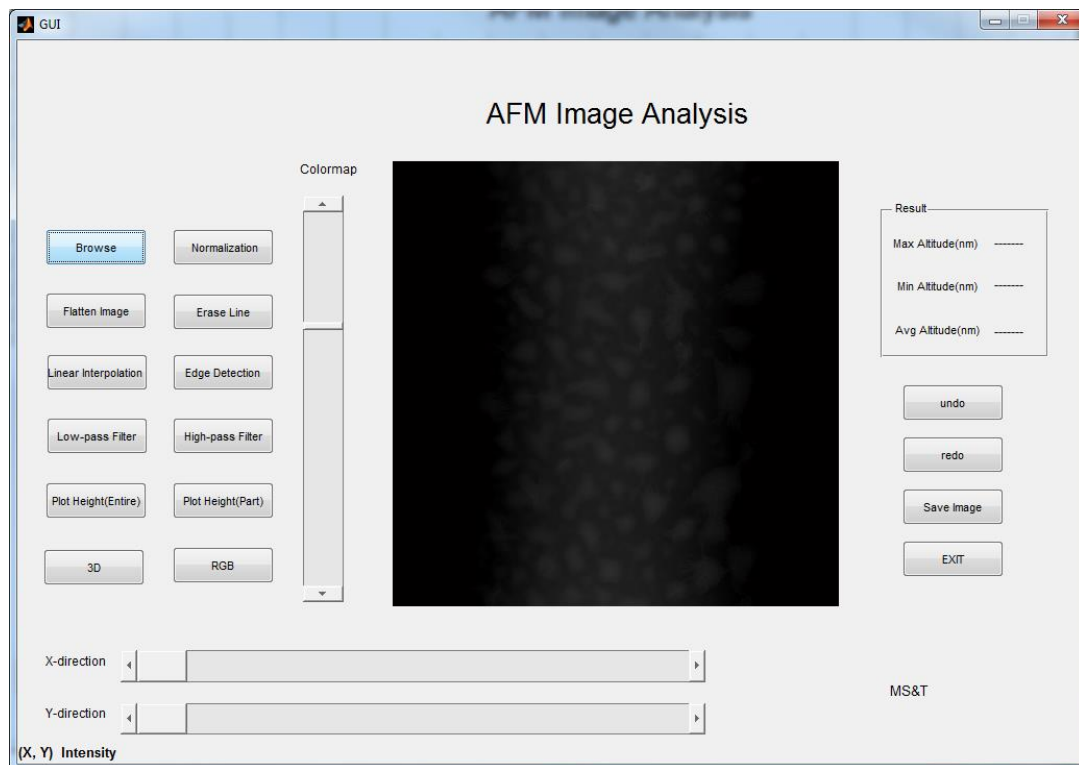


Figure 2.2. Example of an image displayed in the GUI application.

Users can also click cancel button when the object file is not available, and the interface will turn back to the beginning interface again as showed in Figure 1.4.

2.2. FLATTENING IMAGE

Due to the mechanics of the Atomic Force Microscope (AFM), a curvature distortion also referred to as bowing effect, can be observed in gained AFM object scans. An example of an AFM image with bowing effect has been presented in figure 1.3. If the

background in the AFM object sample scan has considerable tilt in it, the change in height of the sample will be affected by the change in height of background [10]. Some types of objects even with very slight height tilt can cause noticeable bowing effects. The intensity or value distribution varies according to different AFM images. Sometimes intensities are relatively higher in the middle of the image while lower at both sides. However, in some other cases, intensities in the middle are lower than the two sides. Hence, erasing the curvature distortion of the background can be used to address this distortion/artifact in acquired AFM data. In other research, AFM image systems commonly address this bowing effect using line flattening or plane fitting which requires human to segment object data from the background. Unfortunately, this manually labeling to generate an exclusive mask is very time consuming and inaccurate [11]. A method of automatic line flattening (also known as image leveling) is an iterative technique applied to each row of the image. In this routine, a planar leveling algorithm was investigated and developed to automatically exclude object points in each row of the recorded AFM image. Typically, the data from each row in the AFM image are fitted by a polynomial, which is reduced by raw data values of that row. Compared to the former manual detection, the new method presents the following several strengths:

- (i) It is desirably automatic and labor intensive.
 - (ii) It increases the accuracy of labeling the objects in the image to a large extent.
- Since for the manual labeling, objects in AFM images are not easily distinguishable in original images.

The dimension of the normalized image we obtain in this research is 512 x 512 and i line from the normalized image is modeled as:

$$y_i(x) = s_i(x) + p_i(x) \quad (3)$$

where $s_i(x)$ represents normalized signal information without the bowing effect, $p_i(x)$ is the distortion of the image due to the bowing effect and $y_i(x)$ represents the data from the original normalized AFM image; x is the horizontal coordinate, i is based on the line number in the image.

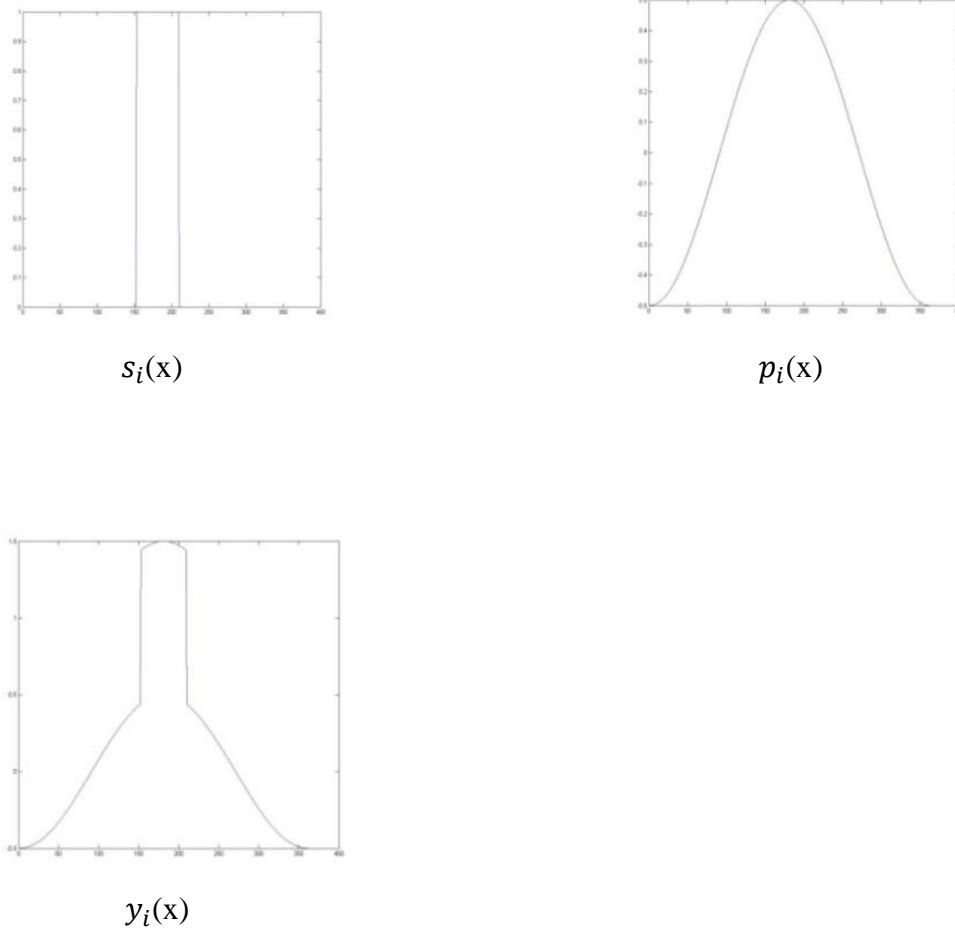
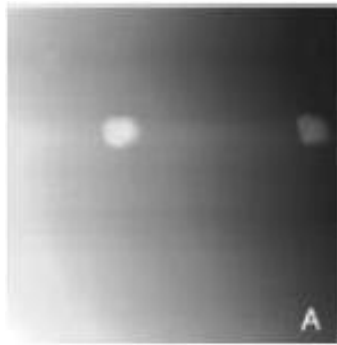


Figure 2.3. Signal $s_i(x)$, polynomial $p_i(x)$ and raw data $y_i(x)$.

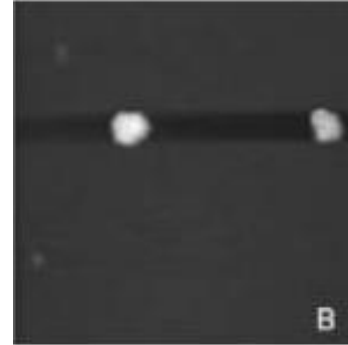
Consequently, the main purpose is to remove the bowing effect which is $p_i(x)$ in this example and recover the object information accurately. The approach to deduce the bowing effecting equation is to calculate a convex polynomial fitting on the AFM image background. In this routine, the value of each row in the image is fit to a polynomial equation. Erasing this polynomial from the recorded scan line would exclude the convexity effects completely. Each line in an AFM image is flattened iteratively in the following equation:

$$s_i(x) = y_i(x) - p_i(x) \quad (4)$$

If the AFM image has a few isolated features, direct polynomial subtracting will lead such images to show shadows around features on a flattened surface. Examples are presented below in Figure 2.4 [12].



(a) Original AFM image



(b) AFM image with polynomial subtraction directly

Figure 2.4. Example showing how artificial streaks look like around the cells.

The reason is that polynomial fitting may include the features. In fact, where the large features, whose intensity values have apparent difference with background, appear on the image, the substrate becomes artificially lowered which means the larger the features, the worse the first polynomial fit [13]. To overcome this problem, dividing the AFM normalized images into two clusters, which are the background and objects respectively, and picking up the background cluster to do the further operation become necessary [14]. Specific steps are given below:

- (1) Fit a 6th polynomial p_i and erase it from the normalized line,

$$s_i = y_i - p_i \quad (5)$$

where y is the normalized pixel values of AFM images and i represents the row number. In MATLAB, the function 'polyfit' was firstly utilized to find the coefficients of the polynomial p_i that fits the data p_i to y_i . Then another function 'polyval' can apply these coefficients and return the value of the polynomial p_i evaluated at y_i .

(2) Determine whether it is required to subtract background and objects. If it is required, continue, otherwise go to step 5 directly.

The core factor in this step is the intensity value (data value) distribution in the current row of the image, denoted as the function s_i . It is observed from the experimental image data distribution that the intensity values on the rows with large features usually concentrate on certain specific tonal variations whereas other rows which are without large features usually have larger tonal variation. To find these large variations, we need to follow certain procedures.

The procedures are as follows. First, a histogram of the current line in the whole image s is made. Then, the total number of positions with histogram value of 0 is counted. Finally, this number is divided by 50 (determined empirically from the experimental data). If the quotient is greater than 1.5, then relatively large features exist on the current scan line; therefore, this scan line needs to be clustered.

(3) If s_i contains objects, use the K-means algorithm to cluster the AFM image into two observations, which are object and background, respectively. In defining the background, a new cluster marked as background is defined as a function z over the whole image and every row in function z is denoted as z_i , where i is the row number. By convention, the background observations in z_i have relatively lower intensity than the objects present.

K-means clustering is a method of cluster analysis which aims to partition n observations into k clusters in which each observation belongs to the clusters with the nearest mean; therefore, all the points are clustered only by their values, not their positions. In order to segment the background and object, s_i is clustered into two clusters ($k = 2$), and the cluster with the lower centroid values is marked as background, and the cluster with the larger values is labeled as object. The background function z is determined by minimizing the sum \bar{z}_i of squared errors,

$$\bar{z}_i = \sum_{c_{ij} \in z_i} (c_{ij} - u_k)^2 \quad (6)$$

where c_{ij} refers to the background labeled pixels from the columns for matrix s_i and u_k is the centroid of cluster z_i .

(4) Use the same way as Step 1 to fit a new polynomial q_i evaluated by the new function z_i and erase it from the scan line.

$$s_i = y_i - q_i \quad (7)$$

(5) Output s_i as the final flattened image.

The flow chart of this algorithm is given below:

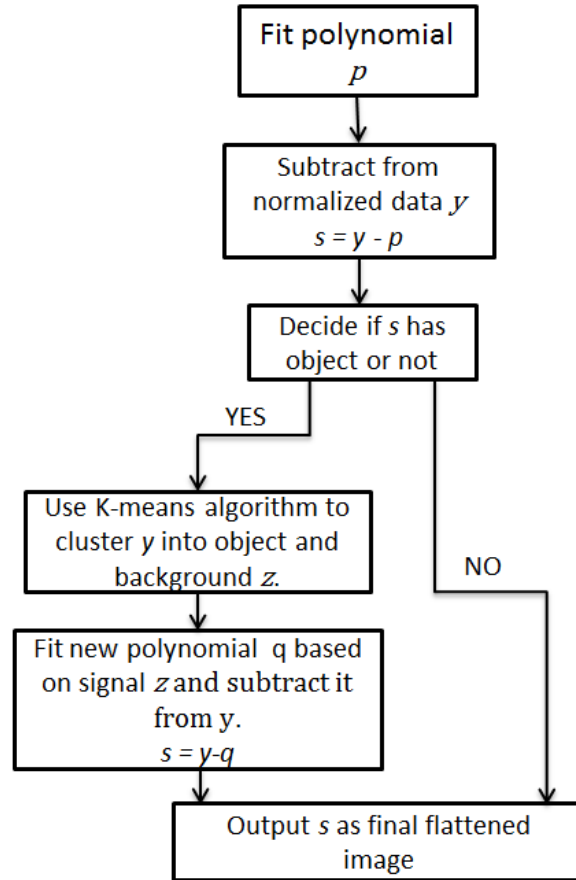
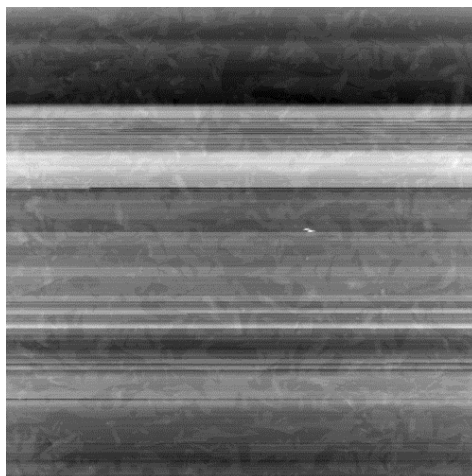
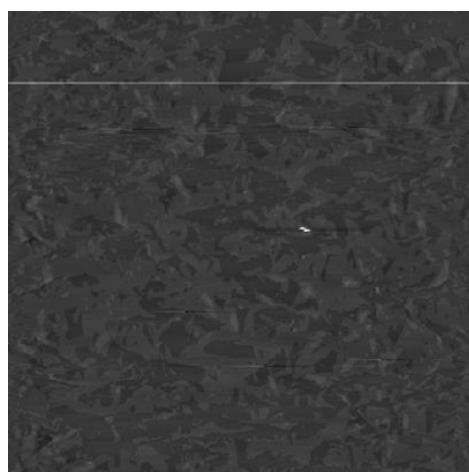


Figure 2.5. Flow chart for flattening image algorithm.

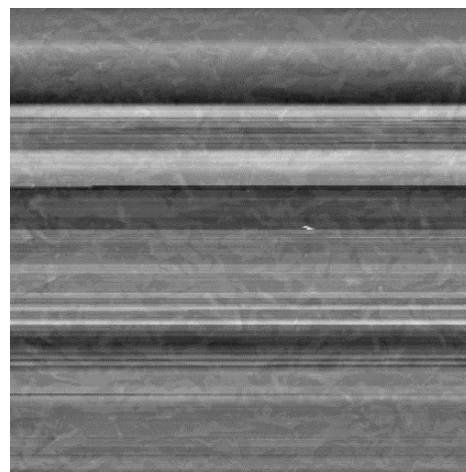
In this study, each horizontal line of the image is processed in this way, but the approach can be applied based on each vertical line in the images. The reason why the procedure is carried out in the horizontal line is that the horizontal axis is usually the fast scan axis for AFM data acquisition. In more detail, horizontal discontinuities in the AFM image are led by change in imaging conditions which will be accounted for by a horizontal line-by-line leveling. Figure 2.6 provides the comparison of an AFM image managed by the horizontal leveling and vertical leveling respectively.



(a) Original Image



(b) Horizontal leveling



(c) Vertical leveling

Figure 2.6. Comparison of horizontal leveling and vertical leveling.

2.3. ERASE LINE

Stripes or strokes which appear in some parts of the AFM images are very common in AFM recorded images. This striping or stroking effect present in AFM data commonly occurs for several reasons. In the process of scanning, a cantilever may suddenly encounter an unexpected high topography. The probe would probably be

damaged by some changes of external environmental conditions like the thermal excitation or noise shaking influence [15]. Distorted AFM data will result and propagate for the row scan for several sample positions until the AFM scan returns to the proper tracking of the sample topography which is easy to miss some area's scanning. An effective procedure is a need to fill up these stripes or strokes. One of the common methods is to remove the entire line where these streaks locate and replace them with an average or a median value of the neighboring scan lines. In this research, a median filter is applied to obtain the median value of the neighboring scan lines so that streaks are repainted. Figure 2.7 shows an example of steaks on an AFM image and its corresponding image processed by the median filter.

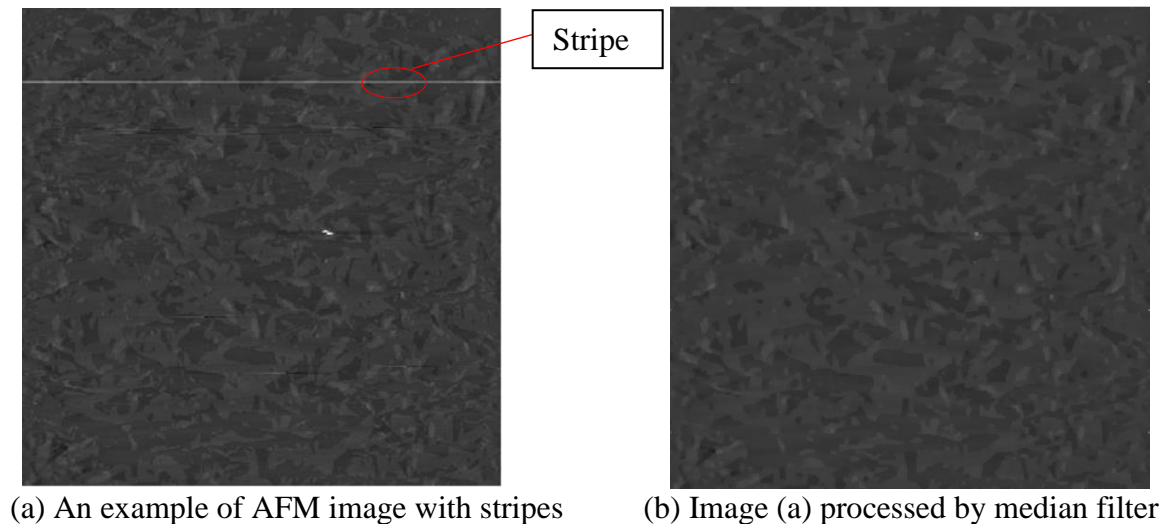


Figure 2.7. Example of an AFM image processed by the median filter.

The critical step in this algorithm is to utilize a unique algorithm to locate stripes position. Through the analysis of intensity in each row, we find the average intensity of every row is nearly equal except the row which contains streaks or strokes. When a row whose average intensity is totally different with others is found, it is the row where streaks or stripes locate. The size of each AFM image we obtained in this research is

512x512, so we can get 512 average row values and pick up the value which is distinct to others among these 512 values. The approach to pick up the distorted rows is illustrated below. Figure 2.8 demonstrates an example of how the average intensity in each row distributes corresponding to Figure 2.7(a).

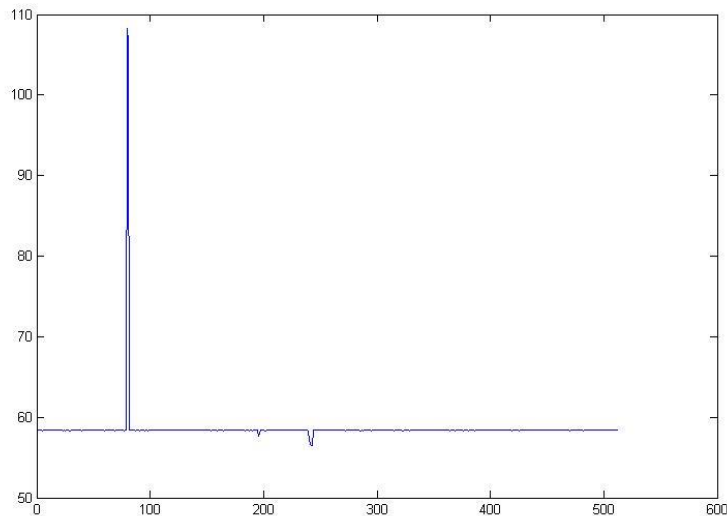


Figure 2.8. Histogram of an AFM row distribution.

In more details the steps are:

- (1) Process the normalized AFM image by the median filter with the size of 3x3.
- (2) Acquire average value from every row and save every row's average value $[a_1, a_2, a_3, \dots, a_{512}]$ in matrix A.
- (3) Pick up the median value a_m from matrix A.
- (4) Round every row's average value $[a_1, a_2, a_3, \dots, a_{512}]$ to their nearest integers and save these values $[int_1, int_2, int_3, \dots, int_{512}]$ in matrix A_i . At the same time, round median value a_m to its nearest integer int_m .
- (5) Calculate the difference between each average value in matrix A_i and int_m respectively and save absolute value of these difference $[b_1, b_2, b_3, \dots, b_{512}]$ in matrix B.

$$b_i = \text{abs}(int_i - int_m), i=1, 2, 3, \dots, 512 \quad (8)$$

(6) Mark the rows in matrix B whose value is not zero and these marked rows are ones where streaks or stripes locate.

(7) Remove the marked rows information in the normalized AFM image and be replaced by the same rows information from the AFM image processed by median filter. This approach can successfully fill up the streaks or stripes and remain the AFM data integrity.

2.4. LINEAR INTERPOLATION

Erasing lines is an efficient and key way to detect and fill up the horizontal stripes in the recorded AFM images. In fact, there are still some light areas or spots on AFM images which cannot be dealt with only by erasing the line operation. For this reason, linear interpolation is applied to find and shrink the light areas automatically.

Linear interpolation is the act of finding a point within two other points so it uses a straight line to find a point between two others. The formula for linear interpolation is:

$$\frac{n-y_0}{m-x_0} = \frac{y_1-y_0}{x_1-x_0} \quad (9)$$

where points (x_0, y_0) and (x_1, y_1) are two given coordinates, point (m, n) is a point within these two given coordinates. Linear interpolation method in image processing averages values of neighboring pixels to calculate values at intermediate points. In our research, 10 pixels (this number is determined empirically from the experimental data) from both sides of the light spot are selected to evaluate and recover blank area data. Each light area in an AFM image is detected and erased in two processes: (1) threshold images to binary images which contain all objects and background equal to '0' but the scars or light areas to '1', followed by (2) using neighborhood lines to 'fill-in' the gaps. The whole procedures are illustrated as follows:

(1) Detect the light areas by thresholding the entire normalized original images and mark the areas with the value '1' which represent the light areas and all other pixels with the value 0. 'im2bw' function in MATLAB is used to threshold the image. It is a function to convert the grayscale image to a binary image. The output image replaces all pixels in the input image with luminance greater than 0.9(determined empirically from the experimental data) with the value 1.

- (2) Pick up 10 pixels from left and right sides of the light area which is done with respect to rows respectively.
- (3) Use MATLAB 'interp1' function to do interpolating algorithm to conjecture a value 'x1' which can represent that certain light spot value.
- (4) Use the x1 as the final replacing data to fill up the light spot.

Erase line function and linear interpolation function both have their corresponding advantages to recover missing data; therefore, a combination of these two methods is required in data repairing section.

2.5. EDGE DETECTION

Distinguishing and analyzing the characterization of particles in AFM is another important task. However, distinguishing the particles in the agglomerates is a big problem to strict nanoparticles characterization with a high precision. Consequently, edge detection which can detect and localize the boundaries of objects in the AFM images is an efficient way to tackle this problem. Edge detection occupies a significant position in surface property for technological application. An edge can be described as a sudden change of intensity in an image. Especially, in binary images, edge relies on abrupt change in intensity level to 1 from 0 and vice versa. Conventional methods of edge detection include Roberts, Prewitt, Sobel and Canny method. The example below displayed in Figure 2.9 shows how the edge detection methods work on the AFM image [16].

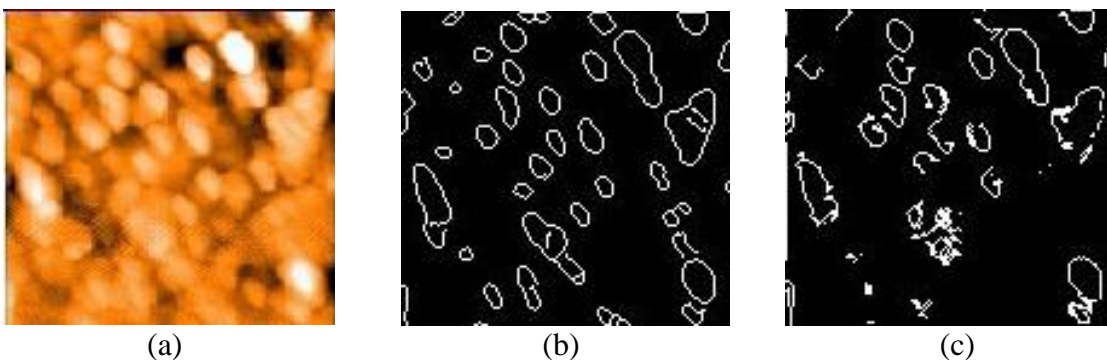


Figure 2.9. Edge detection examples.

(a) Original AFM image (b) Final edge detected image using the Canny method (c) Final edge detected image using the Prewitt method

The tool to find edge direction at location (h,v) of an image, f, is the gradient, defined as the vector:

$$\begin{bmatrix} g_h \\ g_v \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial h} \\ \frac{\partial f}{\partial v} \end{bmatrix} \quad (10)$$

Considering a 3x3 region as an example, details on each edge operator are elaborated below [17].

$$\begin{bmatrix} z_1 & z_2 & z_3 \\ z_4 & z_5 & z_6 \\ z_7 & z_8 & z_9 \end{bmatrix}$$

2.5.1. Roberts Operator. The Roberts operator is based on the diagonal differences. These derivatives can be implemented by filtering an image with the mask as follows:

$$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

The partial derivatives using above mask are given by:

$$g_h = \frac{\partial f}{\partial h} = (z_9 - z_5) \quad ; \quad (11)$$

$$g_v = \frac{\partial f}{\partial v} = (z_8 - z_6) \quad (12)$$

2.5.2. Prewitt Operator. The size of mask for Prewitt operator is 3x3. So these masks take into consideration the nature of the data on opposite sides of the center point and carry more information regarding the direction of an edge. The Prewitt mask is given below:

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

The partial derivatives using Prewitt masks are given by:

$$g_h = \frac{\partial f}{\partial h} = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3) \quad (13)$$

$$g_v = \frac{\partial f}{\partial v} = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7) \quad (14)$$

2.5.3. Sobel Operator. Similar to the Prewitt operator, Sobel operator also uses a 3x3 mask to implement. While the difference is that Sobel operator uses a weight of 2 in the center coefficient to smooth images. The mask to implement in Sobel operator shows below:

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

The partial derivatives using Sobel masks are given by:

$$g_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \quad (15)$$

and

$$g_y = \frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7) \quad (16)$$

2.5.4. Canny Operator. The Canny operator is applicable to search for local maxima of the gradient of image. The derivative of a Gaussian filter serves to calculate the gradient. The method uses two thresholds, to detect strong and weak edges. This method is more likely to detect genuine weak edges.

In this research, scalars of all edge operators' code in MATLAB are implemented by their default parameters.

2.6. LOW-PASS FILTER & HIGH-PASS FILTER

Unwanted high or low-frequency noise often appears in the AFM image, so it's required to use a filter to remove this noise. In our research, two types of matrix filters, low-pass filter and high-pass filter, are applied. Matrix filters belong to the linear spatial filter which is based on average adjacent points in the image to erase certain frequency [18].

Low-pass filter, also known as average filter, only allows low-frequency components of the AFM image to pass. Low-pass filter has a smoothing effect because it prevents all high-frequency components from AFM images. This filter is simply the average of the pixels contained in the neighborhood of the filter mask. A 3x3 filter mask is chosen in the research and the example is given below:

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

On the other hand, high-pass filter allows all high-frequency elements to pass while decreases low-frequency parts. Consequently, this filter is always used to sharp image or enhance edges in AFM images. A 3x3 filter mask is still selected and the example is provided as follows:

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

2.7. 3-DIMENSIONS

Essentially, AFM height data provide three-dimensional information about the objects within the image. However, the method mentioned above of describing AFM data provides a two-dimensional image, using intensity or color scale to represent height information. It is difficult to compare features such as object shape at different height within the same image. Converting 2-dimensional images to 3-dimensional images is an advanced approach to overcome this barrier. Producing 3-dimensional images is a simple and fast application in our research. Three-dimensional rendering is beneficial for viewers to understand height information and shape. Typically, this is done by the method that maintains the intensity of AFM recorded images. Such techniques really enhance the interpretation of the height information to a great extent.

In order to visualize the shape of AFM images perfectly and analyze features from AFM three-dimensional images, different orientation vision observation is required in the research. The '3-D' button in the GUI defines a default viewpoint as (45°, 45°), where left value is the azimuth angle and right one for the elevation angle. Hence, two slide bars which control 3-dimensional images' azimuth and elevation viewpoint respectively. The azimuth is a polar angle in the x-y plane, with positive angles indication counterclockwise rotation of viewpoint. Elevation is the angle above or below the x-y plane. The diagram below makes an interpolation of azimuth and elevation in detail.

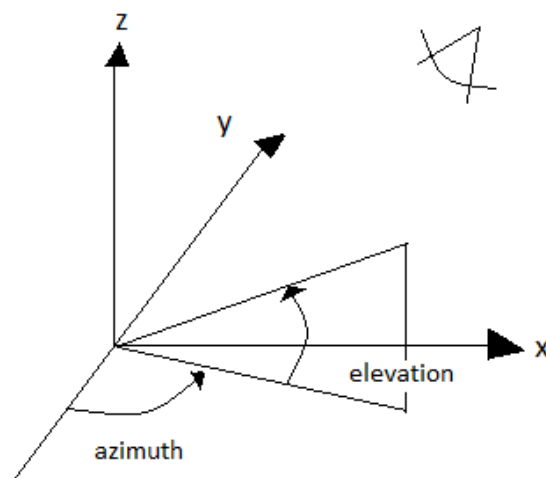
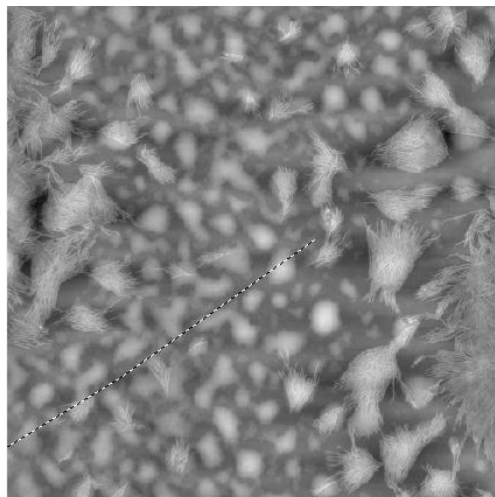


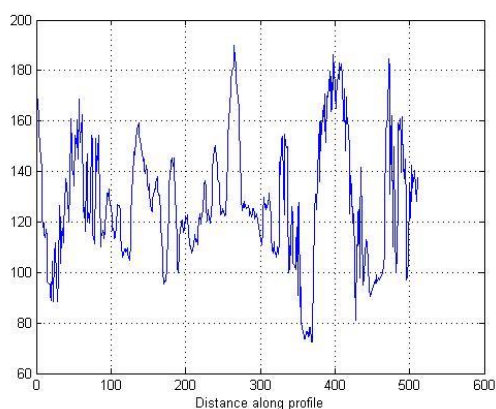
Figure 2.10. Viewpoints with Azimuth and Elevation.

2.8. PLOT INTENSITY

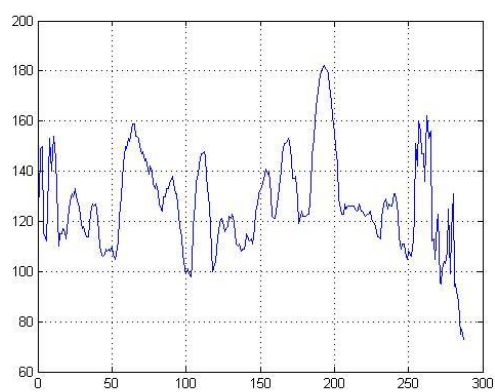
Measurement of a line profile in recorded AFM images is a basic and fundamental analysis technique in this research. In this function, users are allowed to arbitrarily define extracted lines and any direction is permitted no matter horizontal, vertical or even at any angle. Then a new plot is constructed which renders feature heights. Two types of intensity plotting function are designed. They all describe the sample height on the y axis. However, the difference is one's x axis is drawn at the distance exactly equal to the chosen line and the other's x axis is designed with the distance along the chosen line and across the entire image. An example showing an image with an extract line and plot of its height is in Figure 2.11.



(a) Example of an AFM image with extracted line



(b) Line Profile with Entire Image Distance



(c) Line Profile with Certain Chosen

Line

Distance

Figure 2.11. Example of plotting intensity.

2.9. COLORIZE

Colorizing AFM gray-level images is another classic and useful technique. Color is a powerful descriptor that often simplifies object identification and particularly crucial to manual image analysis. The reason is that compared to gray-level AFM images, this method visualizes AFM data evidently and highlights features. According to various

situations, requirements of color for AFM images are also multiple. Thus HSV, which can control color of AFM images in the specified range, is applied to colorize images. HSV stands for hue, saturation, and value, and is also often called HSB (B for brightness). 'Colorize' button we designed in this research defines a default HSV value whose color shade is in yellow. There is a color map in MATLAB and the range is through shades of blue, cyan, green, yellow and red, and end with dark red.

3. RESULT AND DISCUSSION

This section presents the experimental results of operations described in Section 2. The algorithms of Section 2 were implemented in MATLAB. Some functions, like edge detection or 3-dimensional image conversion, were executed by MATLAB toolbox directly, while some others were written by MATLAB code such as flattening image or image normalization. Processing steps change the AFM data. Essentially an image may not need all AFM image processing operations illustrated in section 2, which means different image operations are implemented according to different situations of artifacts. Technically, all processing is carried out for the purpose of enhancing the display of AFM raw data and facilitating the AFM data to be measured and analyzed accurately. As introduced at the beginning, two silicon surfaces with different structures were obtained for this research. In this section, specific procedures are listed below to modify these two silicon surfaces separately and all processed images are available to demonstrate in the center of the GUI interface.

3.1. SILICON SURFACE 1

Click the 'Browse' button to import AFM unprocessed data from the text format to the image format and display the converted image. Then, normalize the image to stretch the range of values between 0 and 255. Figure 3.1 presents how the newly converted image works on left and normalized image on right.

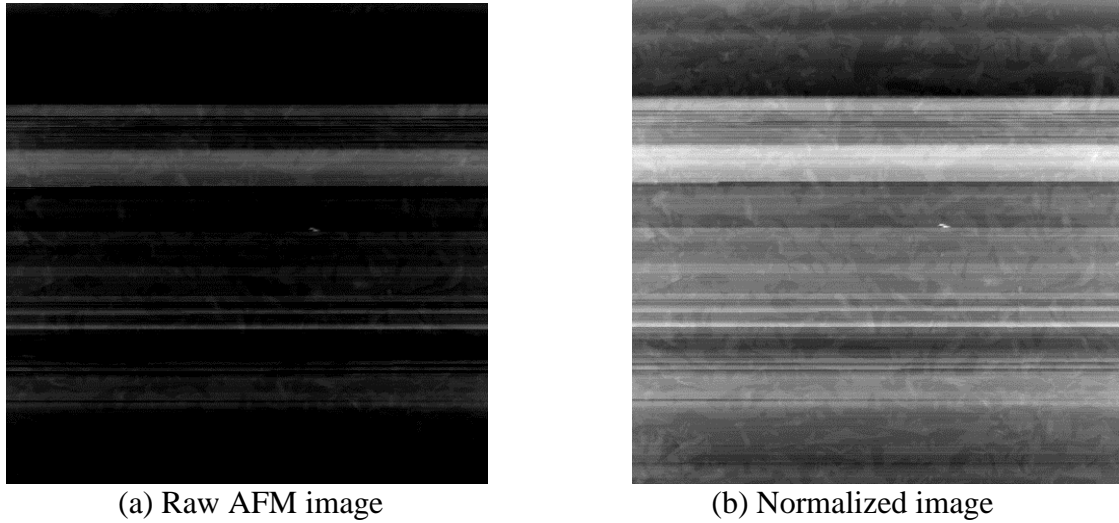


Figure 3.1. Illustration of raw AFM image and normalized image of silicon surface 1.

Figure 3.1 a and b show that the AFM data are captured in a raster format where there are data inconsistencies in adjacent row regions that require further image processing operations. With the purpose to address these artifacts and inconsistencies observed in the collected AFM data, the algorithm and technique flattening image presented in the section 2 are utilized to facilitate shape, structure, and texture analysis for object image analysis. Figure 3.2 shows the newly processed image with image flattening operation.

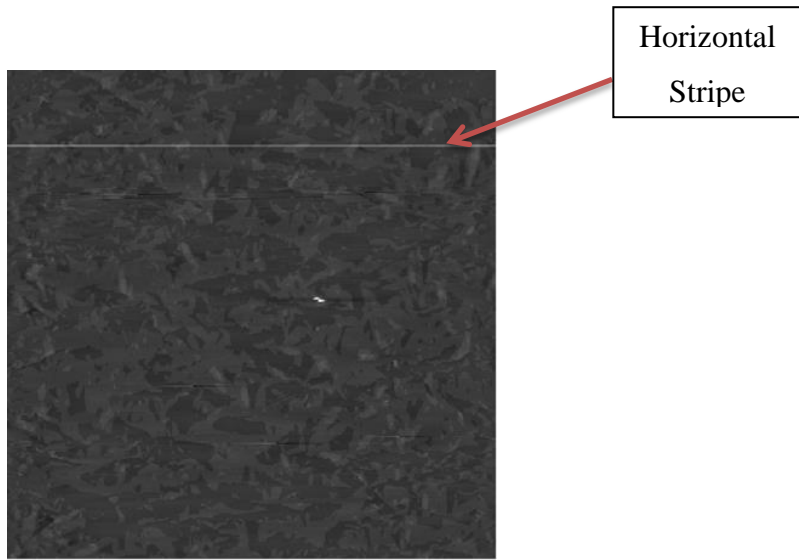
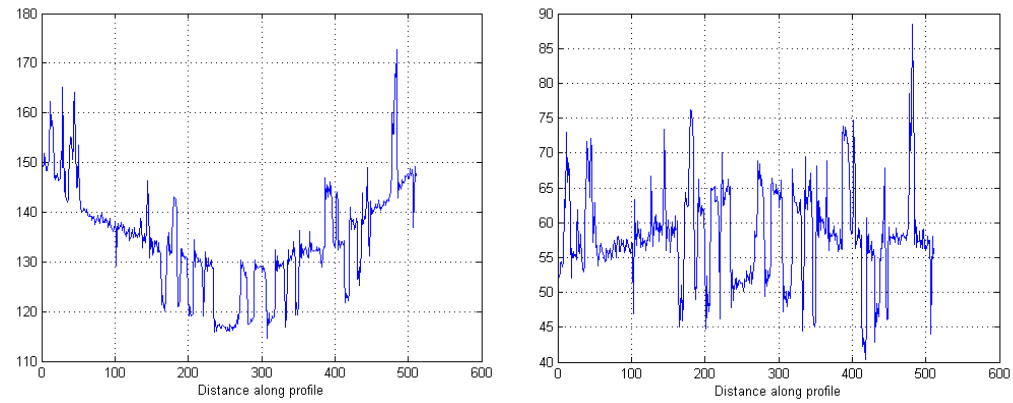


Figure 3.2. Flattened image.

With the aim to compare the difference between the no levelling AFM image and the flattened AFM image, 'Plot Intensity' image button is used to extract the line profile at the same high level between these two images.



(a) No leveling image line profile

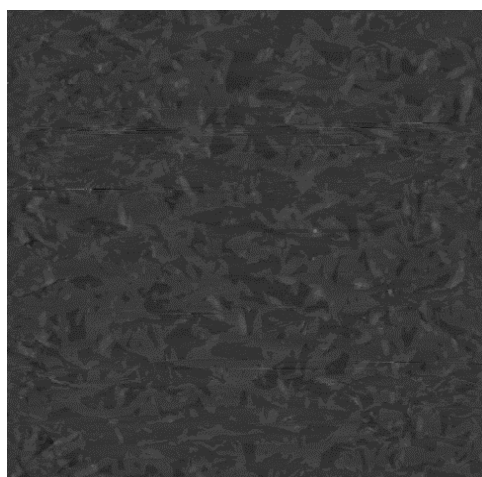
(b) Leveling image line profile

Figure 3.3. Comparison of line profile between no leveling image and leveling image.

From Figure 3.3, note that the curvature distortion has been eliminated in the flattened image. Unfortunately, three horizontal stripes or strokes appear in the figure 3.2. Consequently, the ‘Erase Line’ operation is utilized to automatically find and remove these stripes and fill them in. The result is shown at the left of Figure 3.4 below. Even though the horizontal stripes are erased successfully, there are still some small missing spots or scars on the image. ‘Linear Interpolation’ is the operation that focuses on eliminating small missing spots or scars on AFM images. The left image of figure 3.4 was used as a test to be dealt with the linear interpolation operation and the result is displayed on the right of figure 3.4. The scars were removed perfectly.



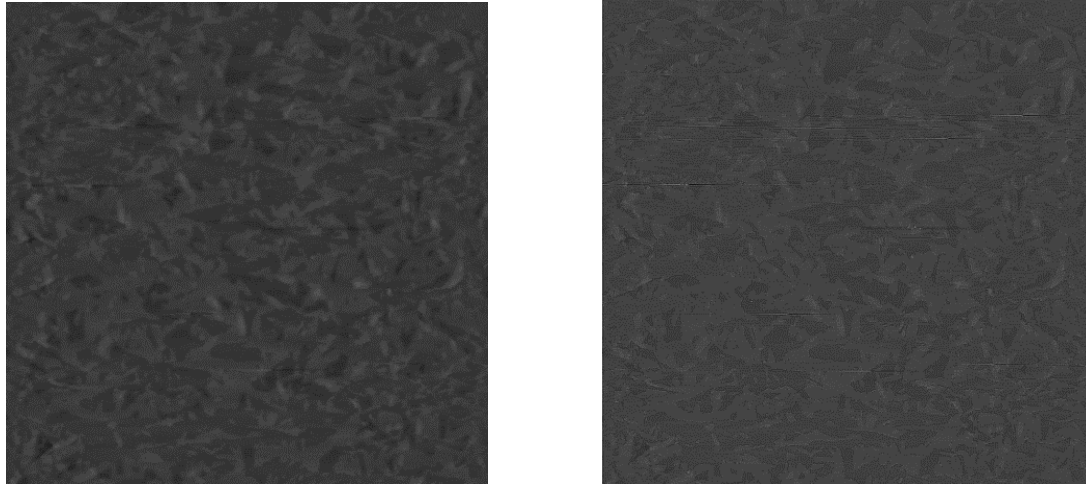
(a) Erased line image



(b) Erased line and linear interpolation image

Figure 3.4. AFM image by erase line and linear interpolation operation.

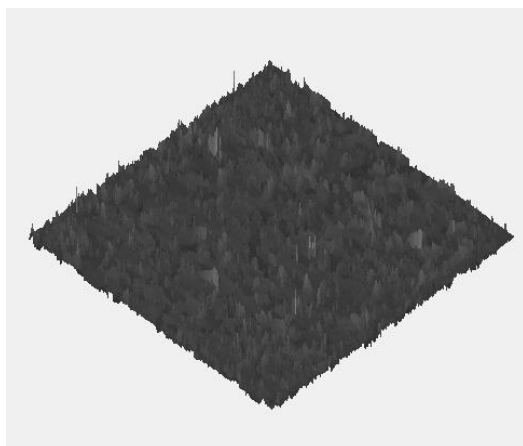
Some AFM images may have unexpected noise which is either in the high frequency domain or in the low frequency domain. Though the data of silicon surface 1 we acquired do not have too much unwanted noise, Figure 3.4 (b) is still used as a test of low-frequency filter and high-frequency filter and the results are provided in Figure 3.5. Compared with these two images in Figure 3.5, we can notice that image filtered by low-pass filter turns smoother than the original one while the features’ edges of image processed by high-pass filter become sharper and more distinct than the original AFM image.



(a) AFM image filtered by low-pass filter (b) AFM image filtered by high-pass filter

Figure 3.5. Silicon surface 1 AFM image results of low-pass filter and high-pass filter.

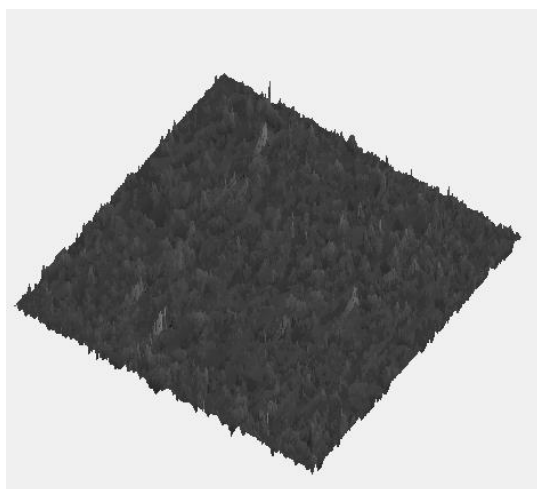
With the purpose to observe the shape of AFM data and render the height information more precisely, 3-dimensional AFM images conversion is significant and indispensable in the research. Figure 3.6 presents an example of the right image in figure 3.4 in 3-dimensional model with the different viewpoints, where left value is the azimuth angle and right one for the elevation angle.



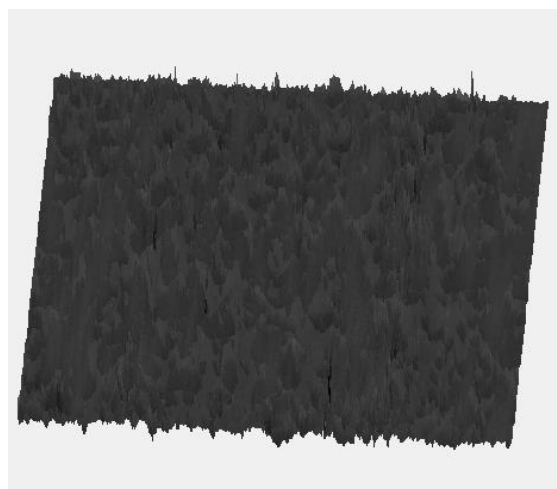
(a) 3-D AFM image with viewpoint
($45^\circ, 45^\circ$)



(b) 3-D AFM image with viewpoint
($0^\circ, 0^\circ$)



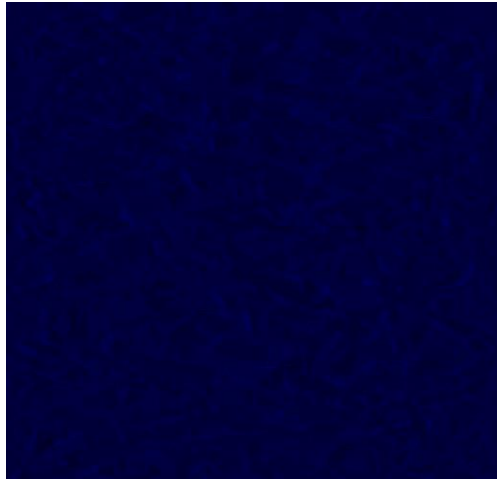
(c) 3-D AFM image with viewpoint
($315^\circ, 45^\circ$)



(d) 3-D AFM image with viewpoint
($90^\circ, 345^\circ$)

Figure 3.6. Examples of a 3-dimensional AFM image with different viewpoints.

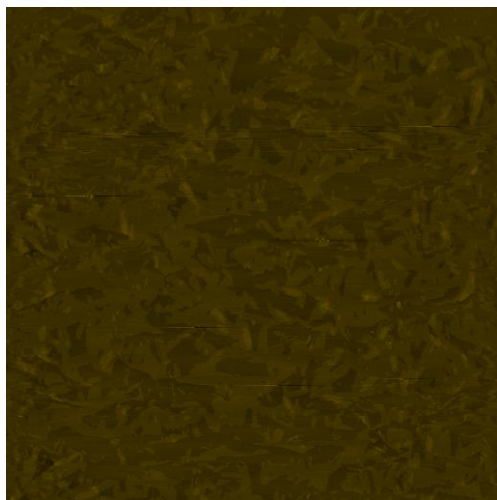
Colorize the gray-level AFM image through the range of shapes among blue, green, yellow and red. Figure 3.7 provides the example of the AFM image shown on Figure 3.4(b) with different color shades.



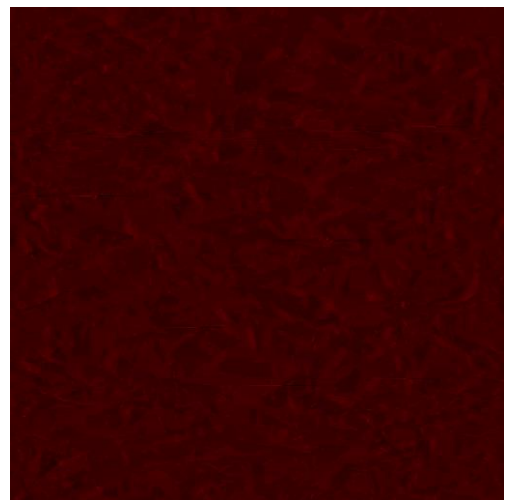
(a) Silicon surface 1 AFM image in blue



(b) Silicon surface 1 AFM image in green



(c) Silicon surface 1 AFM image in yellow



(d) Silicon surface 1 AFM image in red

Figure 3.7. Example of silicon surface 1 AFM image in different color shapes.

3.2. SILICON SURFACE 2

Import silicon surface 2 and display it as the image format. Then normalize the image. The result is displayed below in Figure 3.8.

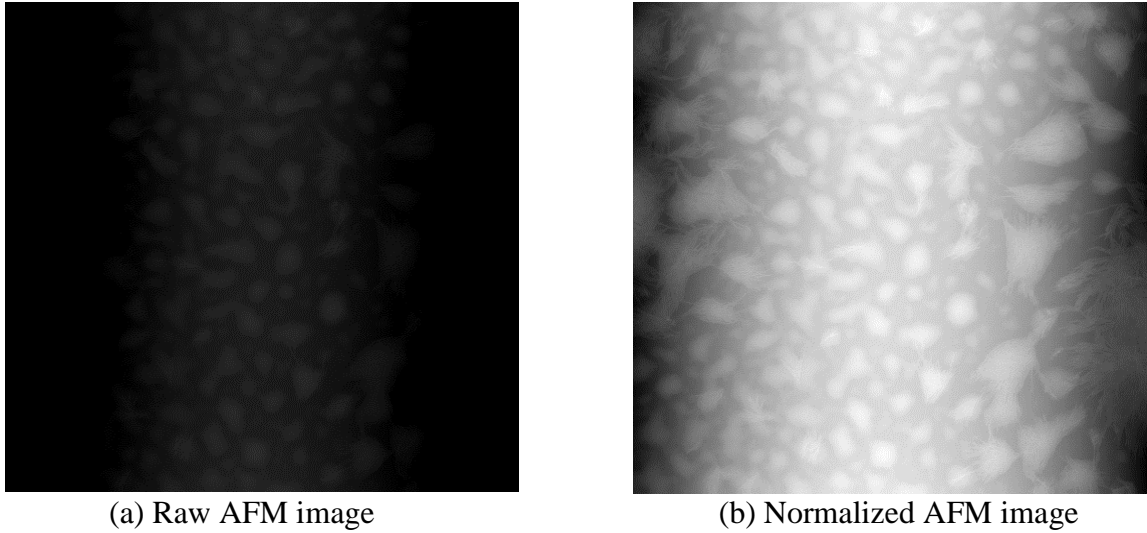


Figure 3.8. Illustration of raw AFM image and normalized image for silicon surface 2.

It can be observed that Figure 3.8 a and b have curvature distortion on the background which requires the certain image analysis technique such as image flattening to erase this background distortion. The result of the flattening image operation is demonstrated below in Figure 3.9.

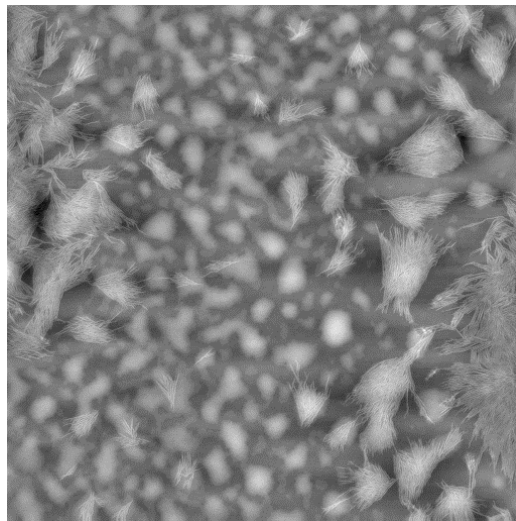


Figure 3.9. Flattened AFM image for silicon surface 2.

Figure 3.10 illustrates the line profile of uneven and flattened AFM image with the same height which is displayed in Figure 3.8 (b) and Figure 3.9 respectively.

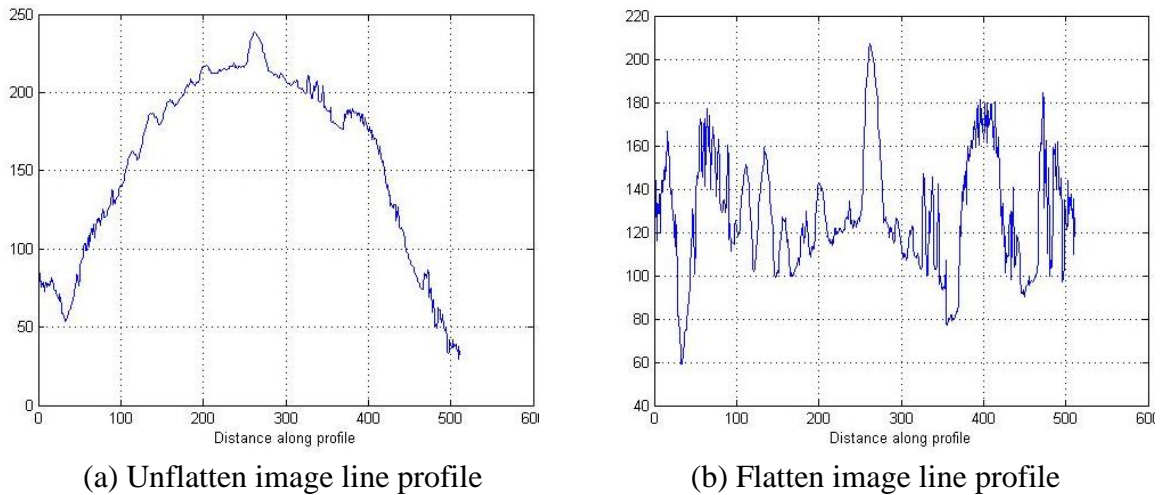


Figure 3.10. Comparison of line profile between uneven image and flattened image.

Luckily, there are no any missing stripes or areas on the surface of this acquired AFM image, so no linear interpolation or erase line operation is required. But features on this AFM image present a very clear edge. Edge detection operation can be used to process this AFM data. Figure 3.9 is applied as a test of edge detection and the results through different edge detection algorithms are presented in Figure 3.11.

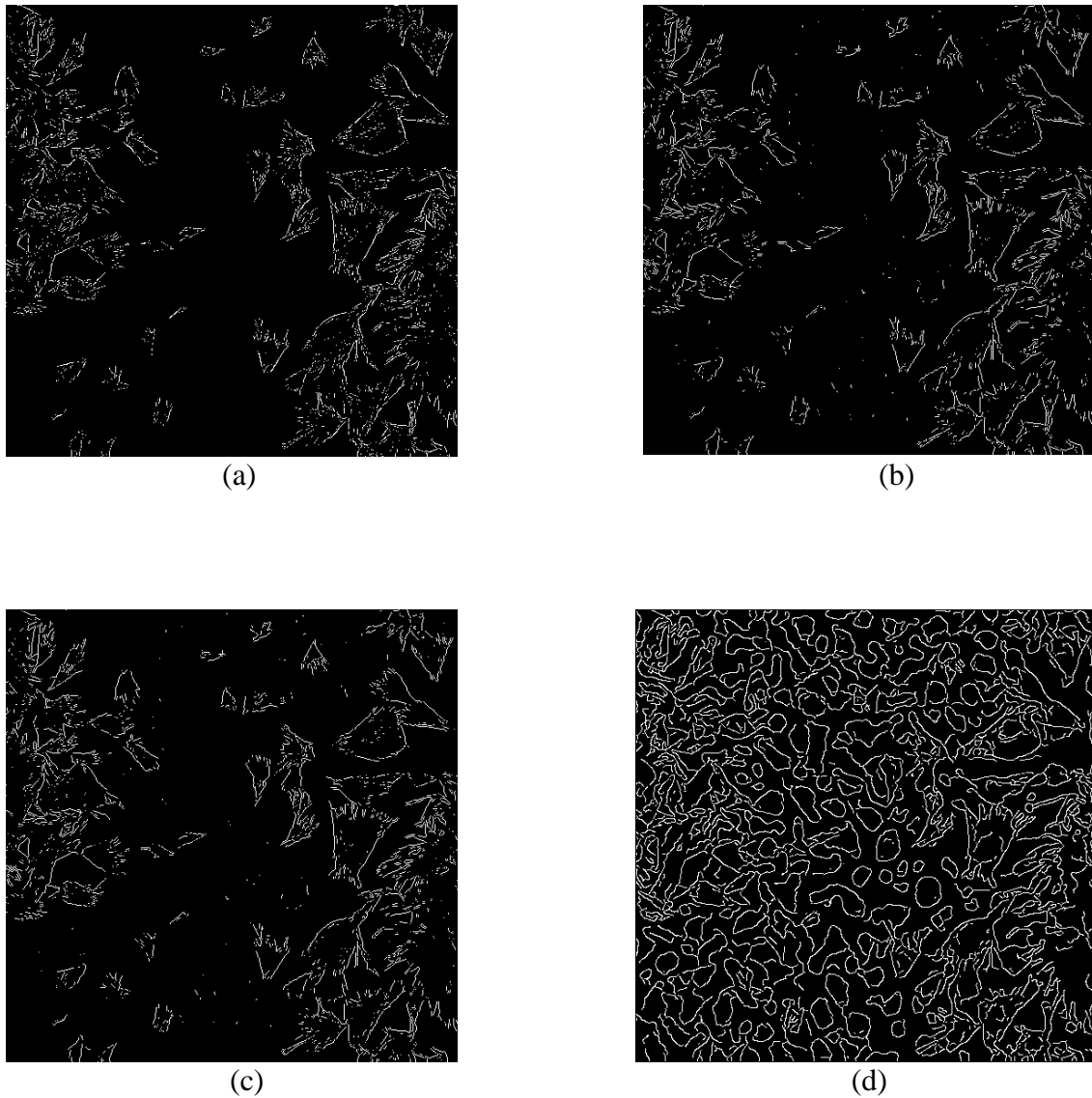
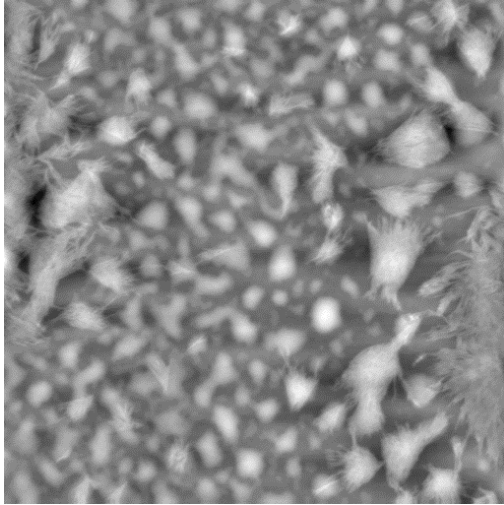


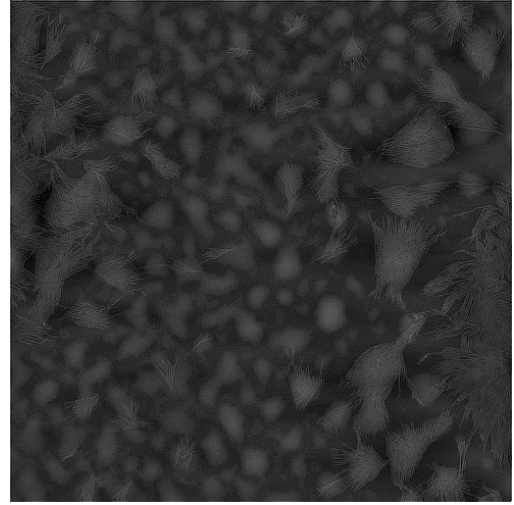
Figure 3.11. AFM images processed by different edge detection operators.

(a) (b) (c) (d) represents the edge detection images processed the Robert, Prewitt, Sobel and Canny method, respectively.

The Sobel method is selected in the research because of its higher resolution on edges and less mixed information. Figure 3.12 shows the results of low-pass filter and high-pass filter working on Figure 3.9.



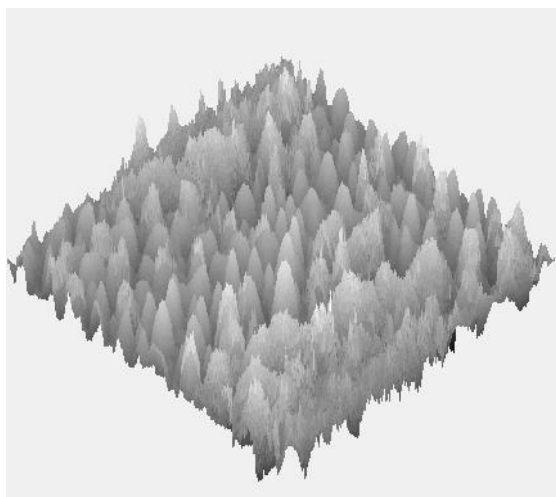
(a) AFM image filtered by low-pass filter



(b) AFM image filtered by high-pass filter

Figure 3.12. Silicon surface 2 AFM image results of low-pass and high-pass filter.

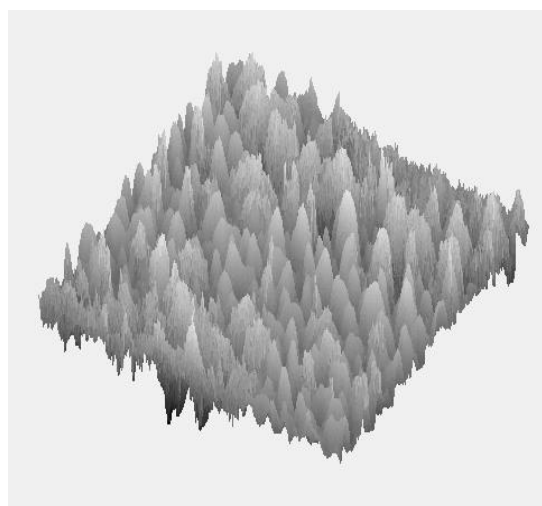
Several 3-D AFM images with different viewpoints according to Figure 3.9 are represented below:



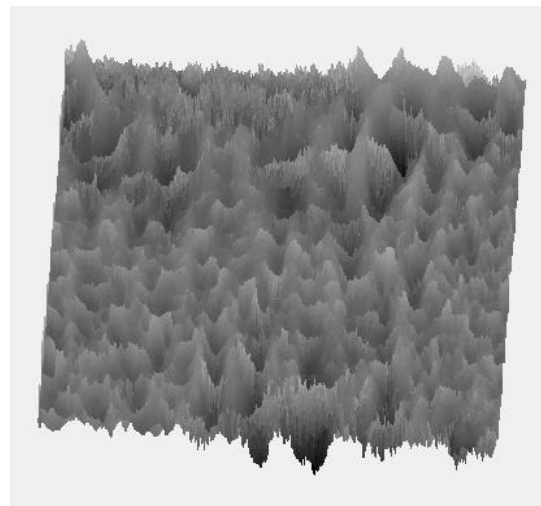
(a) 3-D AFM image with viewpoint
($45^\circ, 45^\circ$)



(b) 3-D AFM image with viewpoint
($0^\circ, 0^\circ$)



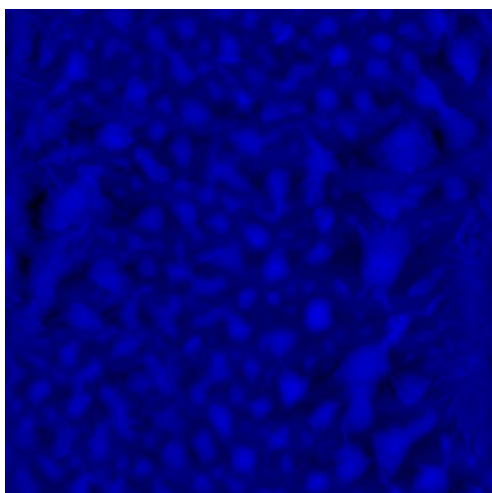
(c) 3-D AFM image with viewpoint
($315^\circ, 45^\circ$)



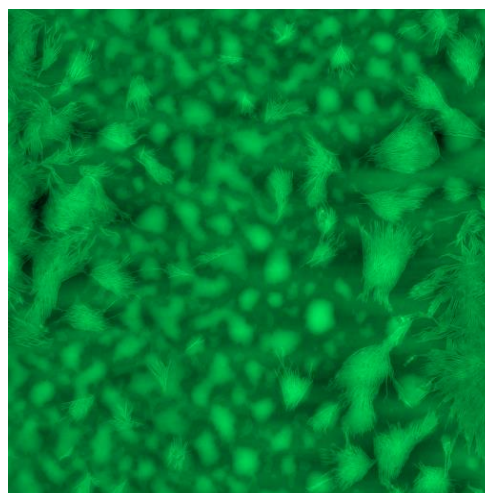
(d) 3-D AFM image with viewpoint
($90^\circ, 345^\circ$)

Figure 3.13. Example of 3-D AFM image.

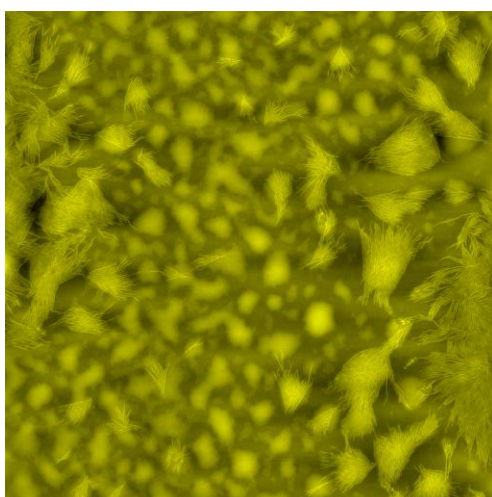
Different color shades work on figure 3.9 and the results are provided on figure 3.14.



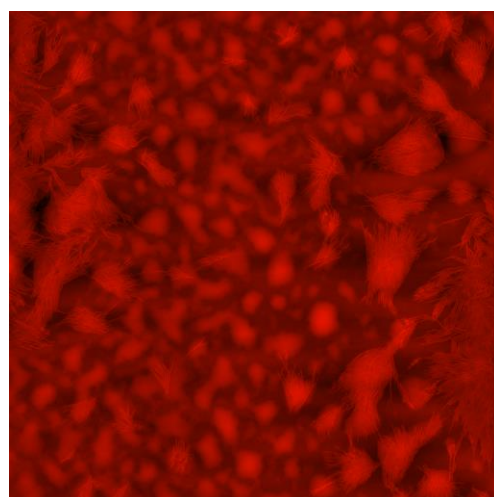
(a) AFM Image Sample in Blue



(b) AFM Image Sample in Green



(c) AFM Image Sample in Yellow



(d) AFM Image Sample in Red

Figure 3.14. Example of AFM images in different colors.

Besides these various functional buttons elucidated above, a result window is located on the top right of the GUI interface. This window will list the maximum, minimum and average height of the current image, which is displayed in the middle of the GUI interface, in nanometer unit. In some cases, users prefer to know data information in some certain positions on the image; therefore, they can move the mouse and click on the desired position on the image directly, where a window which contains

the desired point information (position and intensity information) shows up on the image. An example is provided in Figure 3.15. The image which appears in the middle of the GUI interface is the same image as figure 3.9 and the result window states the maximum, minimum, and average value at 255, 0 and 128.3452 respectively.

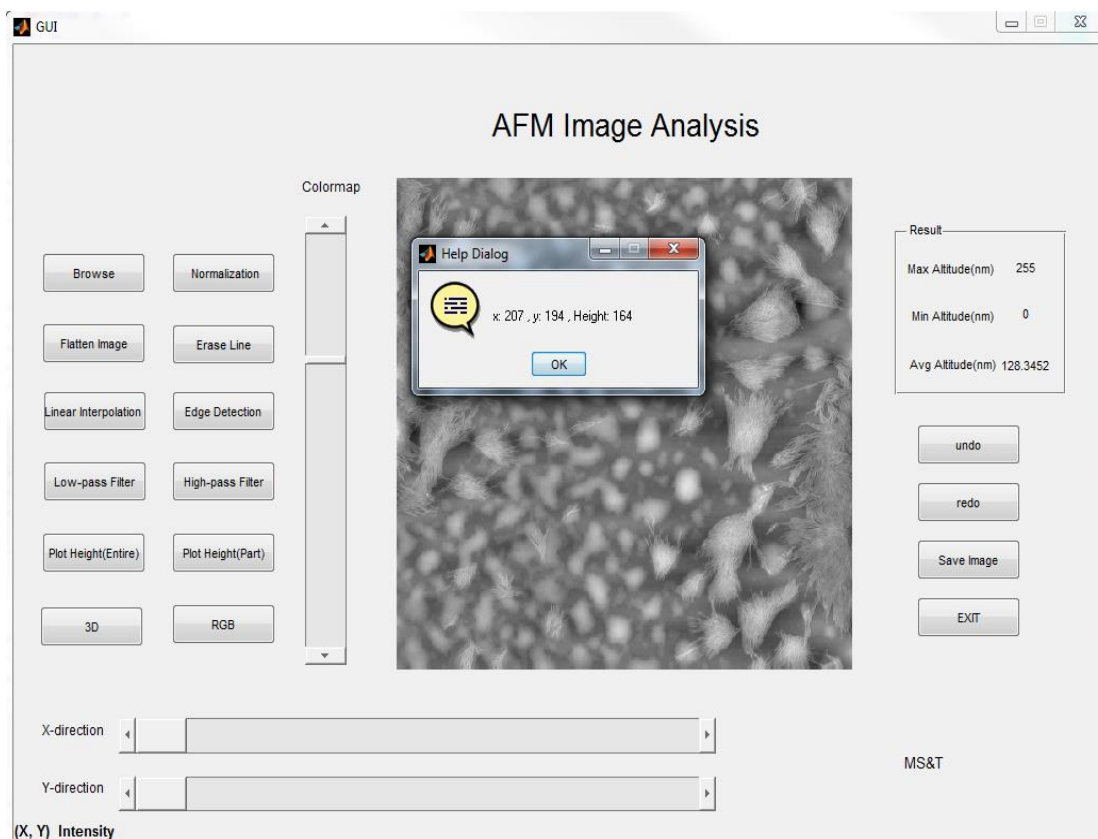


Figure 3.15. Demonstration of the result window.

The rest four buttons below the result window are 'undo' 'redo' 'Save Image' and 'Exit' buttons. 'undo' button is applied for users to go back to the previous step, while 'redo' button is utilized to repeat the former step. When all a series of AFM image processing operations is finished, 'Save Image' button can help users to save the final AFM image in a specified path. Users can also click 'EXIT' button to drop out from the

current interface. Figure 3.16 and Figure 3.17 illustrate how the ‘Save Image’ button and ‘EXIT’ button work respectively.

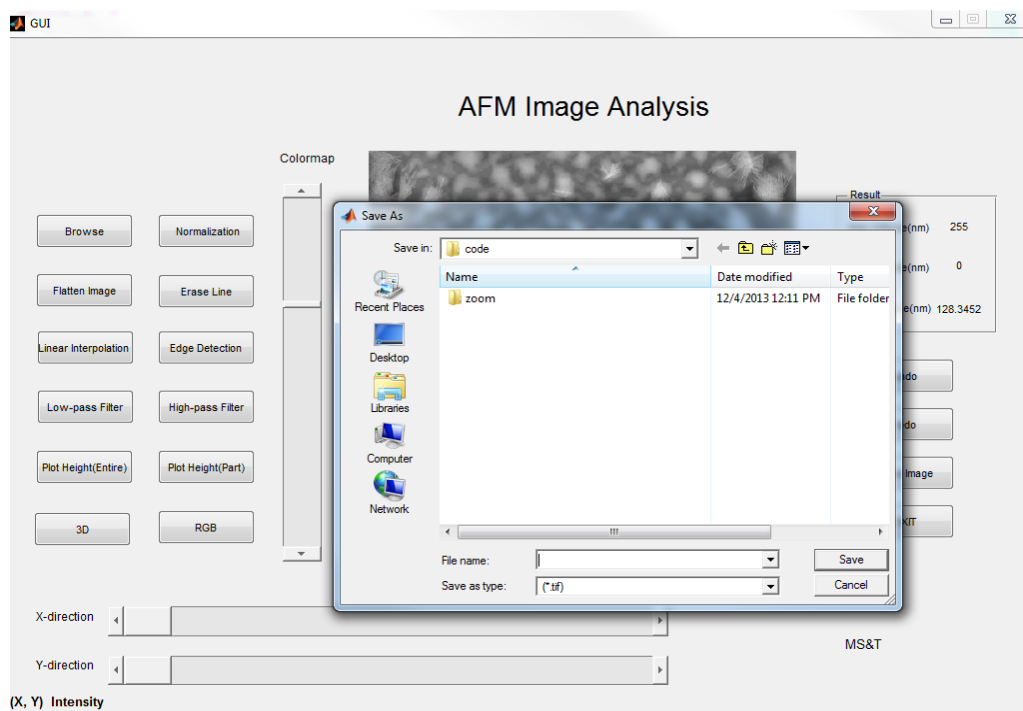


Figure 3.16. Demonstration of ‘Save Image’.

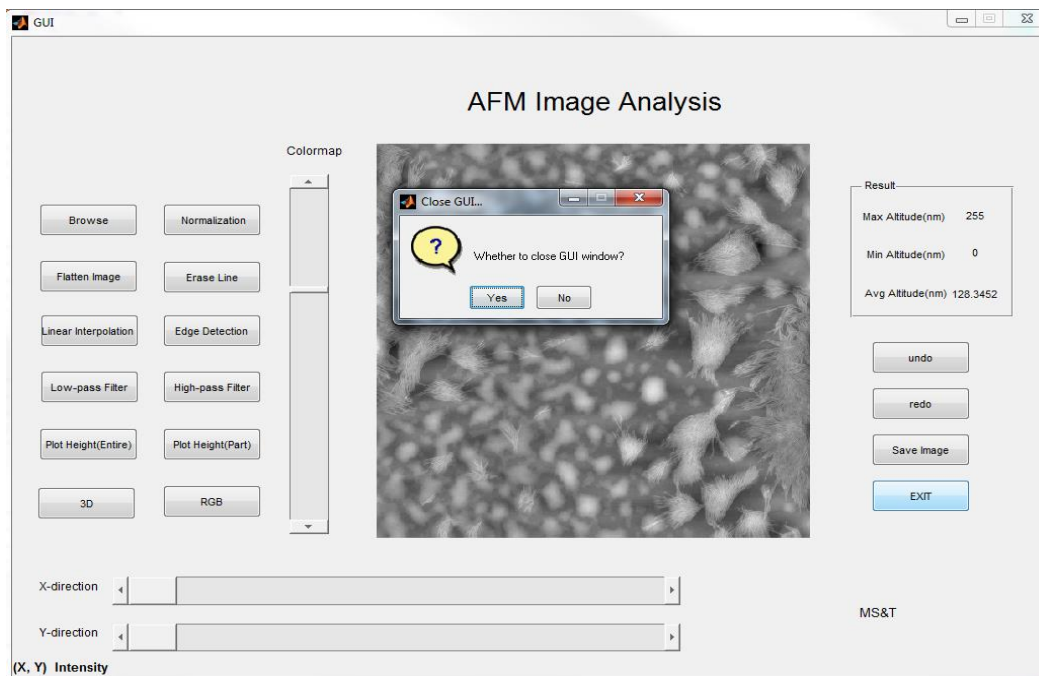


Figure 3.17. Demonstration of 'EXIT' window.

4. CONCLUSION

In this thesis, several functions and algorithms for processing the nonascale Atomic Force Microscope images are presented. These algorithms mainly consist of data conversion, image normalization, automatic flattening images, erasing missing horizontal lines, linear interpolation, filtering images, 3-D images conversion, colorizing images and rendering line profile. The combination of above methods in a certain order helps in erasing artifacts which appeared in the raw AFM data as well as enhancing the microscopic images for analysis. Instructions to use the AFM GUI interface for AFM data processing and analysis have been developed in MATALB. The AFM GUI application was developed for the analysis of AFM data that can be acquired from a variety of sources. Additional AFM image processing functions and approaches will need development for data analysis such as image capture, Fourier Transform filter, roughness analysis and particle analysis.

BIBLIOGRAPHY

- [1] Beveridge, T. J., and Koval, S. F. (1981). Binding of metals to cell envelopes of *Escherichia coli* K-12. *Applied and environmental microbiology*, 42(2), 325-335.
- [2] Binnig, G., Quate, C. F., and Gerber, C. (1986). Atomic force microscope. *Physical review letters*, 56(9), 930.
- [3] Shibata, M., Yamashita, H., Uchihashi, T., Kandori, H., and Ando, T. (2010). High-speed atomic force microscopy shows dynamic molecular processes in photoactivated bacteriorhodopsin. *Nature nanotechnology*, 5(3), 208-212.
- [4] Dufrière, Y. F. (2002). Atomic force microscopy, a powerful tool in microbiology. *Journal of bacteriology*, 184(19), 5205-5213.
- [5] Tello, M., and García, R. (2001). Nano-oxidation of silicon surfaces: Comparison of noncontact and contact atomic-force microscopy methods. *Applied Physics Letters*, 79(3), 424-426.
- [6] Yang, C. W., Hwang, S., Chen, Y. F., Chang, C. S., and Tsai, D. P. (2007). Imaging of soft matter with tapping-mode atomic force microscopy and non-contact-mode atomic force microscopy. *Nanotechnology*, 18(8), 084009.
- [7] Albrecht, T. R., Grütter, P., Horne, D., & Rugar, D. (1991). Frequency modulation detection using high-Q cantilevers for enhanced force microscope sensitivity. *Journal of Applied Physics*, 69(2), 668-673.
- [8] Ricci, D., and Braga, P. C. (2004). Recognizing and avoiding artifacts in AFM imaging. In *Atomic Force Microscopy* (pp. 25-37). Humana Press.
- [9] Gonzalez, R. C., and Woods, R. E. (2008). *Digital Image Processing* (pp. 76-80), New Jersey: Pearson Education.
- [10] Velegol, S. B., Pardi, S., Li, X., Velegol, D., and Logan, B. E. (2003). AFM imaging artifacts due to bacterial cell height and AFM tip geometry. *Langmuir*, 19(3), 851-857.
- [11] Tsaftaris, S. A., Zujovic, J., and Katsaggelos, A. K. (2008). Restoration of the cantilever bowing distortion in Atomic Force Microscopy. *16th European Signal Processing Conference* (pp166-171), Lausanne, Switzerland: EURASIP.
- [12] West, P. E. (2007). *Introduction to atomic force microscopy: Theory, practice, Applications* (pp. 119-121). Pacific Nanotechnology: Santa Clara CA, USA.

- [13] Raposo, M., Ferreira, Q., and Ribeiro, P. A. (2007). A guide for atomic force microscopy analysis of soft-condensed matter. *Modern Research and Educational Topics in Microscopy, 1*, 758-769.
- [14] Eaton, P. J., and West, P. (2010). *Atomic force microscopy* (Vol. 10, pp. 105-108). Oxford: Oxford University Press.
- [15] Chen, A., Bertozzi, A. L., Ashby, P. D., Getreuer, P., and Lou, Y. (2013). Enhancement and recovery in atomic force microscopy images. In *Excursions in Harmonic Analysis, Volume 2* (pp. 311-332). Boston: Birkhäuser.
- [16] Singh, S. (2013). Microscopic Image Analysis of Nanoparticles by Edge Detection Using Ant Colony Optimization. *IOSR Journal of Computer Engineering* (Vol. 11, pp.84-89).
- [17] Jain, R., Kasturi, R., and Schunck, B. G. (1995). *Machine vision* (Vol. 5, pp. 140-185). New York: McGraw-Hill.
- [18] Costen, N. P., Parker, D. M., and Craw, I. (1996). Effects of high-pass and low-pass spatial filtering on face identification. *Perception and Psychophysics*, 58(4), 602-612.

VITA

Xiao Pan was born in Dongying in Shandong province, China in 1989. She received her education at Affiliated Middle School of Liaoning Normal University (2004-2007) and Yuming Middle School (2007-2008) and then obtained her Bachelor of Science degree in Electrical Engineering from the Department of Information Science and Engineering in the Ocean University of China, Qingdao, Shandong (2012). After that, she studied at Missouri University of Science and Technology, Rolla, MO and received her Master of Science degree in Electrical Engineering from the Department of Electrical and Computer Engineering in May 2014.

