

High-speed atomic force microscopy on soft matter



École Polytechnique Fédérale de Lausanne

Lawrence National Berkeley Laboratory

Master's thesis by

Arnaud Bénard

Supervisors:

Prof. Philippe Renaud, EPFL

Dr. Paul Ashby, LBNL

Dr. Dominik Ziegler, LBNL

Berkeley, 2013

Master Thesis: Arnaud Bénard

High-speed atomic force microscopy on soft matter

Molecular Foundry, Lawrence Berkeley National Laboratory, Berkeley, California, USA.

In cooperation with: EPFL-STI-IMT-LMIS4, Microsystems Laboratory 4, EPFL Lausanne, Switzerland

Project Description:

The ability of atomic force microscopy (AFM) to achieve atomic resolution made it one of the most important tools for nanoscience. But piezo-based scanning and serial data collection from the local probe makes image collection slow and unable to match the timescales of many dynamic processes. To overcome this problem we are using advanced scan algorithms such as spiral and cyclic scanning or boundary tracking to increase temporal resolution compared to slow raster scanning techniques. Spiral scanning better matches the mechanical limitations of the AFM scanner and allows higher tip velocities without distorting the image, and using image processing techniques such as inpainting high-resolution images can be restored from sparse quickly collected images without applying more force or increasing bandwidth.

The objective of this master thesis is to achieve video rate imaging using commercially available piezo scanners while maintaining sub-nanometer resolution. To this end already implemented scan patterns need to be improved as follows, the execution time of current inpainting algorithms need to be shortened (processing on the graphics processor), and the noise in the position sensor data is to be reduced using intelligent filter design (Kalman filtering). The fast scan algorithms are to be tested in combination with our newly developed encased cantilevers. The encasement of these cantilevers keeps the resonator dry and only the sharp tip interacts with the sample. This maintains low damping and high resonance frequency as operating in air, which results in increased force sensitivity and allows higher scan speeds operating in liquids. The combination of these cantilevers with fast scan algorithms is believed to enable capturing dynamics of soft samples without perturbing the system under observation. A variety of relevant samples ranging from suspended lipid bilayers, collagen, crystal growth, polymer blends, or membrane proteins in living cells are to be considered.

The results should be thoroughly prepared so that they can be easily incorporated into a manuscript to be submitted to well-known peer-review journals.

The Molecular Foundry

The master thesis will be performed at the Molecular Foundry, in the Lawrence Berkeley National Laboratory, Berkeley, California USA. (<http://foundry.lbl.gov/>). The Molecular Foundry is a US Department of Energy funded institution, providing support to researchers from around the world whose work can benefit from or contribute to nanoscience. The Foundry's research program is centered around four interdisciplinary themes: Combinatorial Nanoscience; Nanointerfaces; Multimodal in situ Nanoimaging; and Single-Digit Nanofabrication. The six in-house research facilities provide state-of-the-art instruments, materials, technical expertise and training to fulfill all tasks involved in this master thesis.

Tasks and Objectives:

- Literature research on fast scanning AFMs techniques.
- Familiarize with existing scan algorithms implemented for on existing home build and Asylum (MFP-3D) microscopes.
- Quantify the performance of existing algorithms in terms of scan-speed, quality of images.
- Improve in-painting techniques (by using OpenGL for instance)
- Implement Kalman filtering on the sensor data
- Test scan algorithms on relevant dynamic samples.
- Preparation of the results for presentation and composition of a report.

Literature:

A collection of high-speed AFM data can be found at
<http://www.highspeedscanning.com/high-speed-afm-gallery.html>

Ando, T., Uchihashi, T., & Fukuma, T., (2008). High-speed atomic force microscopy for nano-visualization of dynamic biomolecular processes. *Progress in Surface Science*, 83(7-9), 337-437.

Yong, Y. K., Moheimani, S. O. & Petersen, I. R. (2010). High-speed cycloid-scan atomic force microscopy. *Nanotechnology*, 21(36), 365503.

Mahmood, I. A., & Reza Moheimani, S. O. (2009). Fast spiral-scan atomic force microscopy. *Nanotechnology*, 20(36), 365503.

Meyer T., et al., Fast Atomic Force Microscopy Imaging using Self-Intersecting Scans and Inpainting, in preparation.
www.csulb.edu/~jchang9/files/AFM_UCLA_REU_2011.pdf
www.csulb.edu/~jchang9/files/toward_capturing_soft_molecular_material_dynamics.pdf

Project Start: (To be discussed) July, 2012.

Project End: 6 month from starting date

Project Supervisor:

Prof. Dr. Philippe Renaud (philippe.renaud@epfl.ch, Phone: +41-21-693-2596)
Microsystems Laboratory 4, EPFL-STI-IMT-LMIS4, Station 17, CH-1015 Switzerland.

Project Advisors:

Dr. Dominik Ziegler (dziegler@lbl.gov, Phone: +1-510-486-7347, Building 67 Room 1202)
and Dr. Paul Ashby (pdashby@lbl.gov, Phone: +1-510-486-7081, Building 67 Room 2228)
Materials Sciences Division, Mail Stop 67R2206, Lawrence Berkeley National Laboratory,
1 Cyclotron Road, Berkeley, California 94720, USA.

Berkeley, December 21st 2011.



Summary

Contents

Summary	v
List of figures	vii
List of tables	ix
1 Introduction	1
2 Visualization of non-gridded data	3
2.1 Image rendering techniques	3
2.1.1 Inpainting algorithms	3
2.1.2 OpenGL	4
3 Techniques for fast z feedback	9
3.1 Dual actuators system	9
3.2 Tilt compensation	10
4 Results	13
4.1 Calcite experiment	13
4.2 Tilt correction	15
5 Conclusion	19
A Programming	21
A.1 Structure of the Spiral Scan program	21
A.2 User interface Igor Pro	23
A.3 Other figures	24
Bibliography	26

List of Figures

2.1	The first image shows that we don't have enough informations to fill out all the pixels. We need to project the data onto the neighbors. Finally, the heat equation inpainting will spread the weighted data onto the missing spots.	4
2.2	Workflow of the rendering program.	5
2.3	Before and after triangulation. This image is rendered by using the triangle.c library. The input data has 20,000 points and the program generates 38,784 triangles.	5
2.4	Delaunay triangulation: From 2D to 3D	6
2.5	OpenGL 3D rendering(1.2M points) of a calibration grating using spiral scans. .	6
2.6	Workflow of the immediate mode and the vertex buffer objects	7
3.1	Normal AFM setup	10
3.2	Z-feedback using fast and slow piezos.	10
3.3	Asylum	11
4.1	Calcite geometry	13
4.2	Setup for the calcite experiment	14
4.3	5s per frame / 100 loops / $5\ \mu m$	14
4.4	Evolution of the calcite dissolution over time.	15
4.5	Path on the XY plane	16
4.6	Input of the tilt compensation	16
4.7	Height of the calibration	17
4.8	Histogram of the calibration	17
4.9	Before and after the tilt correction	17
A.1	Flowchart of the spiral scanning program	21
A.2	Function call of the spiral scanning program	22
A.3	User interface spiral scanning	23
A.4	Memory for the VAO	24



List of Tables

2.1	Rendering results[ms]	8
4.1	Plane fit coefficients	16

1 Introduction

In 1986, a group of scientist from IBM research developed the first Atomic Force Microscope.[3] The original idea was to sense forces between a sharp tip and a surface. The deflection of the cantilever is measured using a laser reflected from the top of the cantilever into a photodetector(array of photodiodes). Nowadays, AFM's applications range from measuring elastic properties of biological samples to imaging of surface topography[15] [5]. The former technique is achieved by using a XY-scanner to move the sample on the horizontal plane. Therefore, the AFM acquires data on different points of the surface. Also, AFMs have a feedback system to maintain a constant tip-sample force.

The most conventional way to scan is with a raster pattern. This technique steers the tip of the AFM on specific points of a grid. Past AFM research has been focused on improving position controllers. Indeed, the AFM precision depends on the quality of the closed loop feedback on XY. These improvements, however, didn't solve fundamental problems with raster scanning. Most of the data is thrown away (trace and retrace) and the actual position on the XY plane is inaccurate - and directly correlated with the efficiency of the position controller.

Because of its limited bandwidth, fast raster scans generate distortions in the image.[20] Spiral pattern techniques generate high-quality images at higher scan frequencies than the raster one. [11]. Moreover, this method reduces the number of data points necessary. [7]

With raster scanning, the data is evenly distributed in space so generating an image becomes trivial. Spiral scan, however, needs new techniques to render images. Fortunately, image processing algorithms like inpainting [16] or Delaunay triangulation have been developed to generate images from sparse data.[7].

The bandwidth on the z-axis control loop is limited by the dynamics of the z scanner.[8] We can achieve higher frequencies by using a small piezoelectric ceramic.[19]

In this thesis, we will see how we can use non-raster scan patterns to improve the bandwidth on the XY-plane. Also, we have developed image processing algorithms to render non-gridded data. We use the popular Open Graphics Library (OpenGL) for the rendering of the X, Y, Z data

Chapter 1. Introduction

and investigate ways to use the Graphics Processing Unit (GPU) to improve the computing time. Finally, we will implement new ways to go beyond the limitations on the z-axis with tilt correction and dual actuators feedback. If an image is tilted, you can use first-order plane fitting to flatten the image. A more efficient way is to dynamically compensate for the tilt of the sample. Also we will see how to improve the bandwidth on the z-axis by introducing a small piezoelectric ceramics.

2 Visualization of non-gridded data

Current AFMs run in closed loop where position controllers on the XY axis are needed to steer the tip at the right position. Instead of using this method, we will work in open loop and register the data of the position sensors. One of the advantages of using sensor data is that we don't need accurate position controllers: it has no impact on our data. Instead of positioning the tip at an exact position, we're embracing its inaccuracy. The precision of our system is limited by the sensors and not the feedback on XY. Our current setup (Asylum Research MFP3D) uses inductive sensors.

Using this technique frees us from raster scan. We don't acquire specific points on a grid so we need image processing algorithms to render missing parts of our scan. We will see in the next section how to find these points.

2.1 Image rendering techniques

We are going to investigate the two following techniques: inpainting and OpenGL. The inpainting methods have been designed and implemented on Matlab by the Paul Ashby Group at LBNL in collaboration with Prof. A. Bertozzi of UCLA. We will briefly discuss its implications in AFM imaging.

Then, we will discuss using OpenGL for the rendering. We will see different methods to implement this library and benchmark its performances.

2.1.1 Inpainting algorithms

Reconstructing missing parts of images was first developed for restoring photographs and paintings or removing undesirable data like text and publicity. The art of restoration was performed manually. Nowadays, tools like Photoshop or Gimp are widely used in the media. It can also be used to produce special effects [16].

This process is called inpainting. The principle behind it is to fill a patch with its surroundings. Mathematicians have developed a wide range of algorithms to solve that problem. We will investigate a special case of partial differential equations (PDE) called heat equations. The heat equation is a PDE that represents the distribution of heat in a region over time.

$$\frac{\partial u}{\partial t} - \alpha \nabla^2 u = 0 \quad (2.1)$$

α is the thermal diffusivity - that is interpreted as a "thermal inertia modulus" - and u is the temperature over space and time (i.e. $u(x, y, z, t)$). A high thermal diffusivity implies that the heat moves rapidly.

This algorithm will spread out the information of each point onto missing parts of our grid.

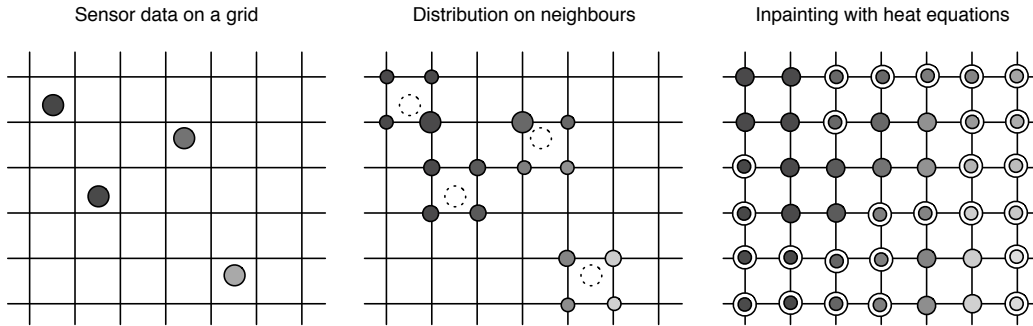


Figure 2.1: The first image shows that we don't have enough informations to fill out all the pixels. We need to project the data onto the neighbors. Finally, the heat equation inpainting will spread the weighted data onto the missing spots.

Literature shows that heat equations are powerful enough to fill out these patches of missing data, but they smooths the sharp edges (high frequency data) [2]. These edges are blurred out by the algorithm.

2.1.2 OpenGL

In this section, we will see how to render images with OpenGL (Open Graphics Library). It is an API (Application Programming Interface) developed by Silicon Graphics to hide the complexities of interfacing with different 3D accelerators and is mainly used for 3D modeling in video games and simulations. OpenGL leverages the fact that GPUs are designed to render triangles. It optimizes the rendering by using the repetition of a geometric shape (tessellation). The more complex the shape the harder it will be for the GPU to process it. If we already pre-process the data into triangles, we minimize processing costs. [1]

The flow of our program is described by the Fig 2.2. First, we take the X, Y, Z data and generate a list of triplets of points (triangles). Then, we render this list with OpenGL. We will see two methods of rendering these triplets: immediate mode and Vertex Buffer Objects.

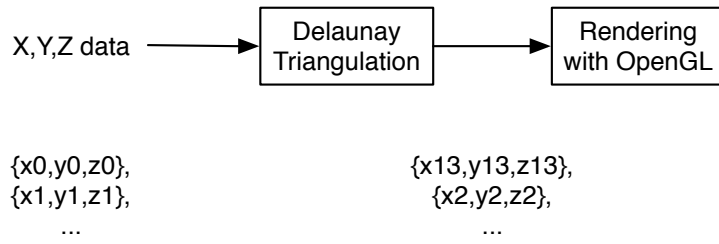


Figure 2.2: Workflow of the rendering program.

Triangulation with Delaunay

The first problem is generating triangles from sparse data. Indeed, OpenGL can only render triangles from a triplet of points. If we render the X, Y, Z data without preprocessing, OpenGL will generate triplets from the closest index in the list. We need efficient algorithms like Delaunay triangulation to take into account the X, Y position of the point and not the index.

The algorithm minimizes the angles of each triangle. The triangulation is successful if no vertex (i.e. 3-dimensional point) is inside a triangle.

Jonathan Shewchuk [17] has developed a library, `triangle.c`, to compute Delaunay triangulations and other meshes. The Figure 2.3 shows the effect of the program on a spiral scan, which will be discussed later.

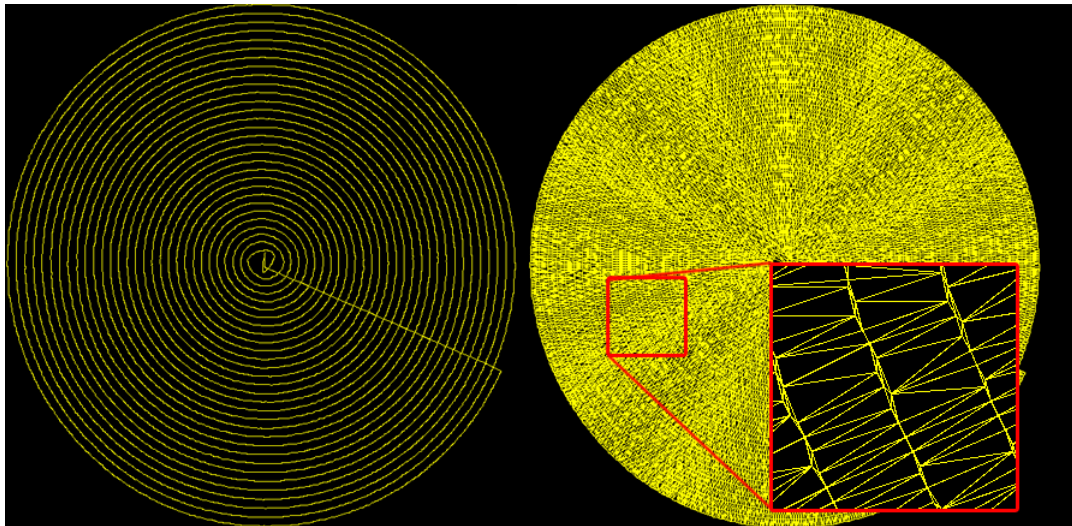


Figure 2.3: Before and after triangulation. This image is rendered by using the `triangle.c` library. The input data has 20,000 points and the program generates 38,784 triangles.

With this method we can connect the points on a 2D plane. In the figure 2.4, we use the z-axis data to compute the height of each point and render 3D models of our scans. To add colors to our data, OpenGL will create a linear gradient between each of the data points.

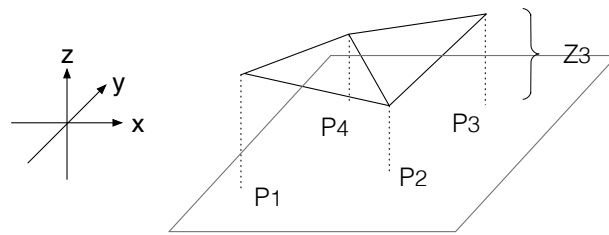


Figure 2.4: Delaunay triangulation: From 2D to 3D

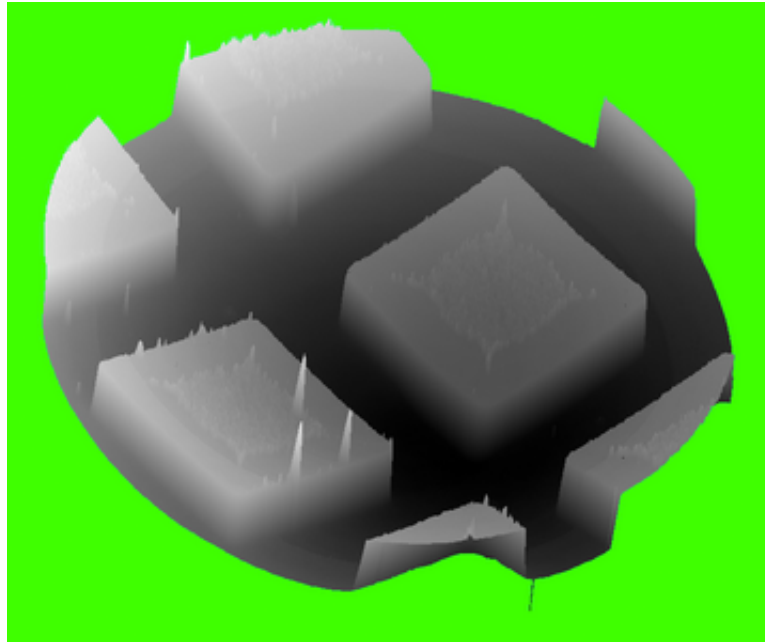


Figure 2.5: OpenGL 3D rendering(1.2M points) of a calibration grating using spiral scans.

Immediate mode vs Vertex Buffer Objects

We will discuss two ways to render our surface with OpenGL: the immediate mode and vertex buffer objects.

The immediate mode is the simplest implementation of OpenGL. Indeed, we render every frame. If we rotate our 3D model, we'll have to regenerate the frame. The power of the immediate mode is its simple implementation (no initialization and extra code). Moreover, it is easier to debug. For a small number of vertices (< 10,000) the immediate mode is appropriate. [9] states that the immediate mode is more convenient and less overhead than other implementations (Vertex Buffer Objects).

The display function is called when GLUT(OpenGL Utility Toolkit) determines that the window needs to be redisplayed. Action like rotation, translation or resizing of the model will trigger the display event. Each time the display function will be called, the program will upload the vertices to the graphic processing unit(GPU).

If we try to display a significant number of triangles ($> 10,000$ vertices), the CPU will be the bottleneck. The GPU doesn't start rendering data before the last callback. Thus, the CPU is spoon-feeding the GPU by transferring the data triangle by triangle. Moreover, the number of API calls is proportional to the number of triangles. For instance, if you have 10 triangles you will make $(10 * (2 + 3 + 3))$ 80 API calls [13]. In conclusion, if you want to render less than 10,000 vertices and are not planning to make a lot of changes in your rendering, the immediate mode is ideal.

One of the problems with the immediate mode is the transfer from the system memory to the GPU. We've seen there is a bottle neck in the transfer. With 10,000 points we can only have 3 frames per second which means that our computer takes 300 ms for the whole process.

Instead of transferring the data from the memory to the GPU, the GPU could read the memory of the program. Buffer objects have been created to allow the GPU to have access to that memory. The process of reading the memory from the GPU is called Direct Memory Access (DMA). A buffer object is a contiguous untyped memory which the CPU and the GPU both have access to.

We can't just upload our data into the memory without any structure. We need to map the data and make it readable for the GPU. We store our data in a vertex array object.

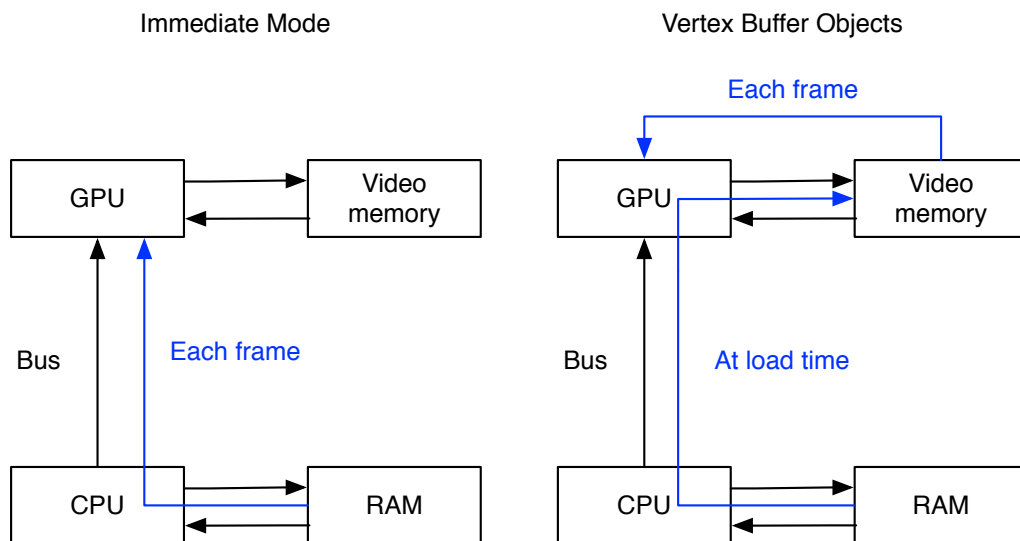


Figure 2.6: Workflow of the immediate mode and the vertex buffer objects

The advantage of this implementation is that you directly pull your data to a shared memory between the CPU and the GPU. Your CPU will spend less cycles making API calls thus improving the performance of the program. The power of the VBOs is that you just need to upload your data and your display function will just bind the VBO. Our performance has improved from 3FPS to 130FPS for 100,000 data points. Having a higher FPS count makes the animations

Chapter 2. Visualization of non-gridded data

smoother.

Table 2.1 shows the non-linearity of our implementation. We see that Delaunay triangulation doesn't scale well for 1,000,000 points. In AFM scans we will rarely sample 1,000,000 datapoints. The limits of our AFM is 100,000 kHz. If we take 10 second scans at the limit rate, we observe that the computation time is still way below the scanning time.

Table 2.1: Rendering results[ms]

Nb of points	Delaunay	VBO	Total time
1000	2.9	23.9	26.8
10000	8.1	27	35.1
100000	66.9	181	247.9
1000000	640.7	267	907.7

3 Techniques for fast z feedback

Implementing model based controllers or high frequency actuators improves the AFM feedback loop[19]. The drawback with these actuators is the decrease in the positioning range.

We will improve the bandwidth and the scan range of our device. Past research has introduced an external piezoelectrical actuator on top of the cantilever. With this scheme, we can combine the advantages of a high bandwidth from the piezoelectrical actuator and the long range of the tip.

3.1 Dual actuators system

The principal feature of an AFM is its probing system. The feedback control system is designed to adjust the motion of the tip on the z-axis. It will adjust the tip-to-sample distance.

The figure 3.1 shows the most classic feedback loop in an AFM. The reference of the controller is the force setpoint. The output of the controller is the topography of our surface. Also, the error signal is the sum between the reference and the deflection of the cantilever.

The problem with this system is that it can't pick up abrupt changes in the topography.[8] The bandwidth of the cantilever is not large enough. This leads to scanning artifacts like parachuting and /ADD OTHER. We could use a cantilever with a higher bandwidth, but the amplitude would be too small. We decided to take to the best of both systems and design a dual-actuator system. We use a standard cantilever to scan the surface and we will add a small piezoelectrical ceramic on top of the X-Y scanner. It enables the AFM to have a larger bandwidth thus allowing it to pick up high spatial frequency topography.

We design another feedback system in order to make the X,Y,Z scanner and the piezoelectrical ceramic work together. [4]

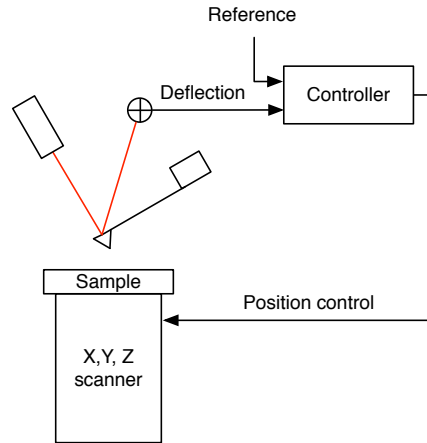


Figure 3.1: Normal AFM setup

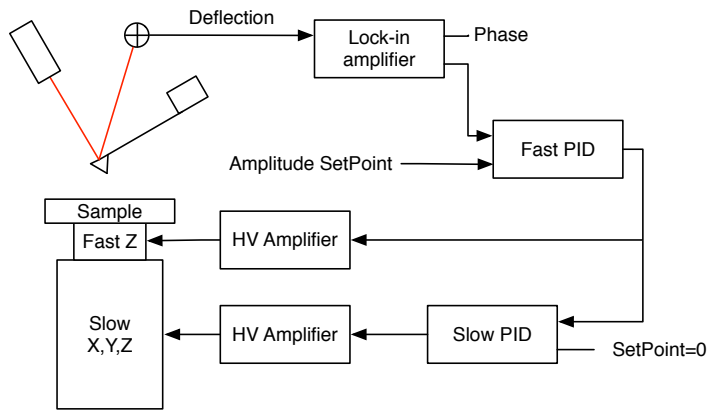


Figure 3.2: Z-feedback using fast and slow piezos.

We need a high-voltage amplifier to operate the small piezoelectric device.

3.2 Tilt compensation

In theory, the mounted sample should be parallel to the XY-Scanner. If the probe/sample angle is not perpendicular, we observe a tilt on the surface. This tilt is problematic when it becomes larger than the features. There are multiple ways to compensate for this. The most common technique is to use post-processing to adjust the image. Flattening algorithms or first-order plane fitting restore the image and put the data on the same level. This technique works if the range of the tip is large enough. We have decided to take another approach and dynamically compensate for the tilt. Before performing our scan, we will do a first scan to compute the tilt of the sample by considering our tilt as a 3D plane. Then, we'll generate a tilt correction signal that will be added to z scan output.

We use a circle pattern to scan the surface of the sample. The radius of the circle is equal to the scan size. It gives us information about the general topography of the surface. Then, we compute the plane equation of the surface by applying a fit in Igor Pro. This fit will generate a plane that models the tilted surface. The input of this fit is the theoretical X-Y output waves of the circle pattern: a sine and a cosine. The data on the z axis is the height.

$$z = a_1 x + a_2 y + a_3 \quad (3.1)$$

The coefficients are computed by minimizing the values of Chi-Square (error function). Then, we generate the waves to send to the controller with the equation ref.

$$w_{avetosend} = a_1 w_{avex} + a_2 w_{avey} \quad (3.2)$$

Wavex and wavey are the output of our scan pattern. With this method, we can do a tilt correction with any scan pattern.

Then, we send the previously computed wave to the controller.

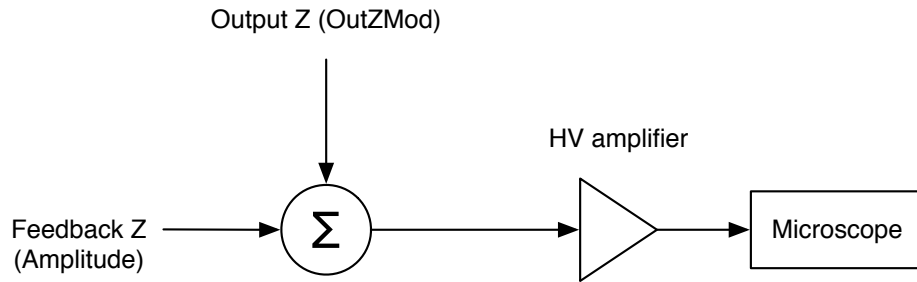


Figure 3.3: Asylum

4 Results

4.1 Calcite experiment

In this experiment we've observed the dissolution of calcite ($CaCO_3$) with water. We have decided to choose this reaction because the chemical reaction leading to the dissolution is simple. Moreover, it is an abundant and geologically important material.[6][10] [14]

The figure 4.1 shows the tilted shape of the calcite dissolution. One part has an angle of 78° and the other 102° [18] [12]

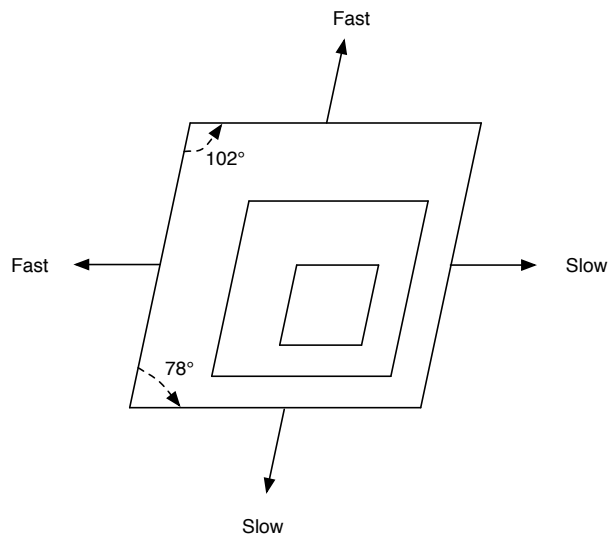
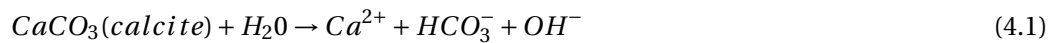


Figure 4.1: Calcite geometry

We have used our double PID feedback system and /InsertTipType. Calcite dissolution is an interesting process to observe with a high speed AFM.



Chapter 4. Results

We have mounted our calcite sample on top of the fast piezo-electrical ceramics. Also, we have stuck a plastic cover underneath the piezo for the water. The calcite will dissolve itself with the pattern shown on figure 4.1. It is linked to its original geometry.

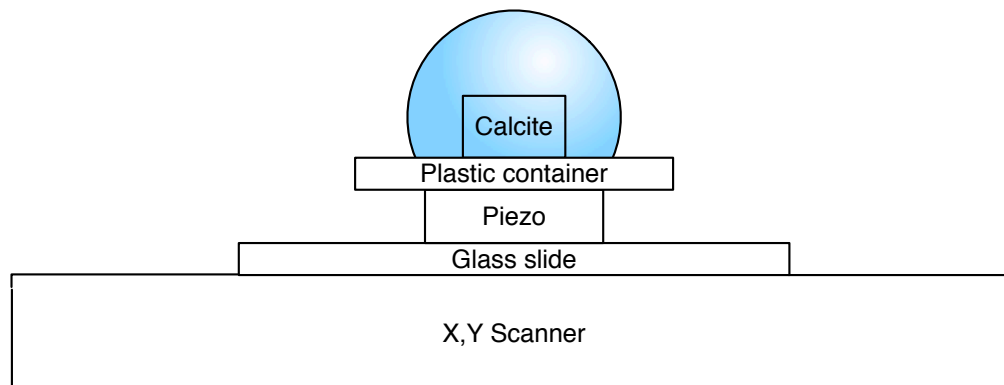


Figure 4.2: Setup for the calcite experiment

We have imaged the same sample with different parameters including scan size, number of spirals and scanning time.

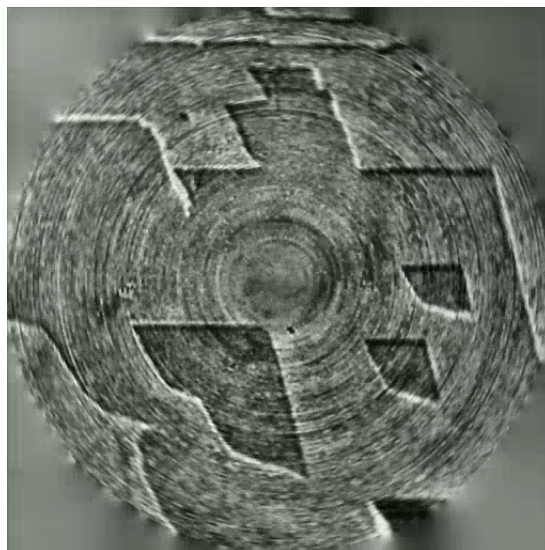


Figure 4.3: 5s per frame / 100 loops / $5\text{ }\mu\text{m}$

The figure 4.4 shows the evolution of the calcite. Every scan took 10 seconds over a surface of $10\text{ }\mu\text{m}$. The scan pattern is an Archimedian spiral with 50 loops. We observe the evolution of the front wave over time. After a few minutes, the water will saturate the calcite and the reaction will stop. You can redo the experiment by removing the water and adding fresh water afterwards.

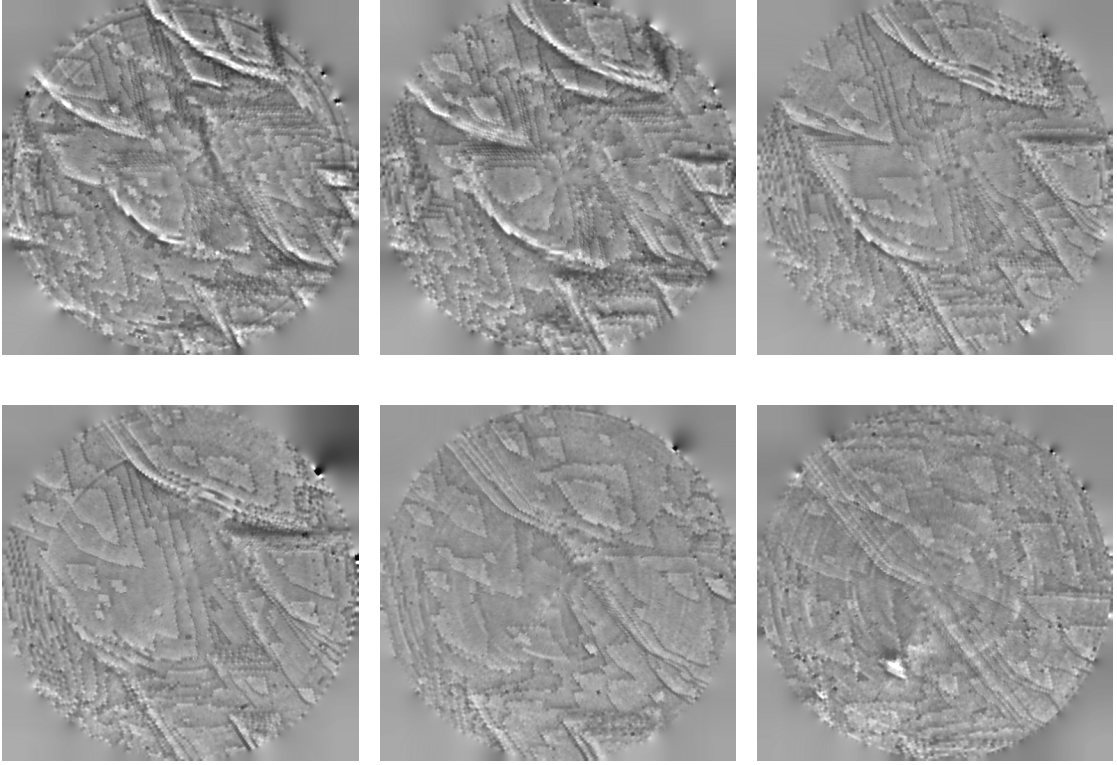


Figure 4.4: Evolution of the calcite dissolution over time.

4.2 Tilt correction

In this experiment, we will show the efficiency of the tilt correction. Our sample is a calibration grating with pyramidal features. We use the dual actuators feedback control described in the section 3.1. Also, the scan pattern is an Archimedean spiral, with a radius of $30 \mu m$ and 80 loops, described by the equations 4.2.

$$\begin{aligned} x(t) &= \alpha\sqrt{t}\cos(\beta\sqrt{t}) & y(t) &= \alpha\sqrt{t}\sin(\beta\sqrt{t}) \end{aligned} \quad (4.2)$$

Before applying the tilt correction, we see on the figure 4.5 a) that our fast piezoelectrical ceramic is saturating. We definitely need to take a load off the actuator.

We have seen in the section 3.2 (equation 3.2) that we need to sample the topography of the surface to compute the plane coefficients. We put a high integral gain for the slow piezo to get the general topography of the sample. The figure 4.5 b) shows that the surface is not perfectly flat (see z-axis).

Then we compute our fitting algorithm on the x,y and z data of the figure 4.5 b).

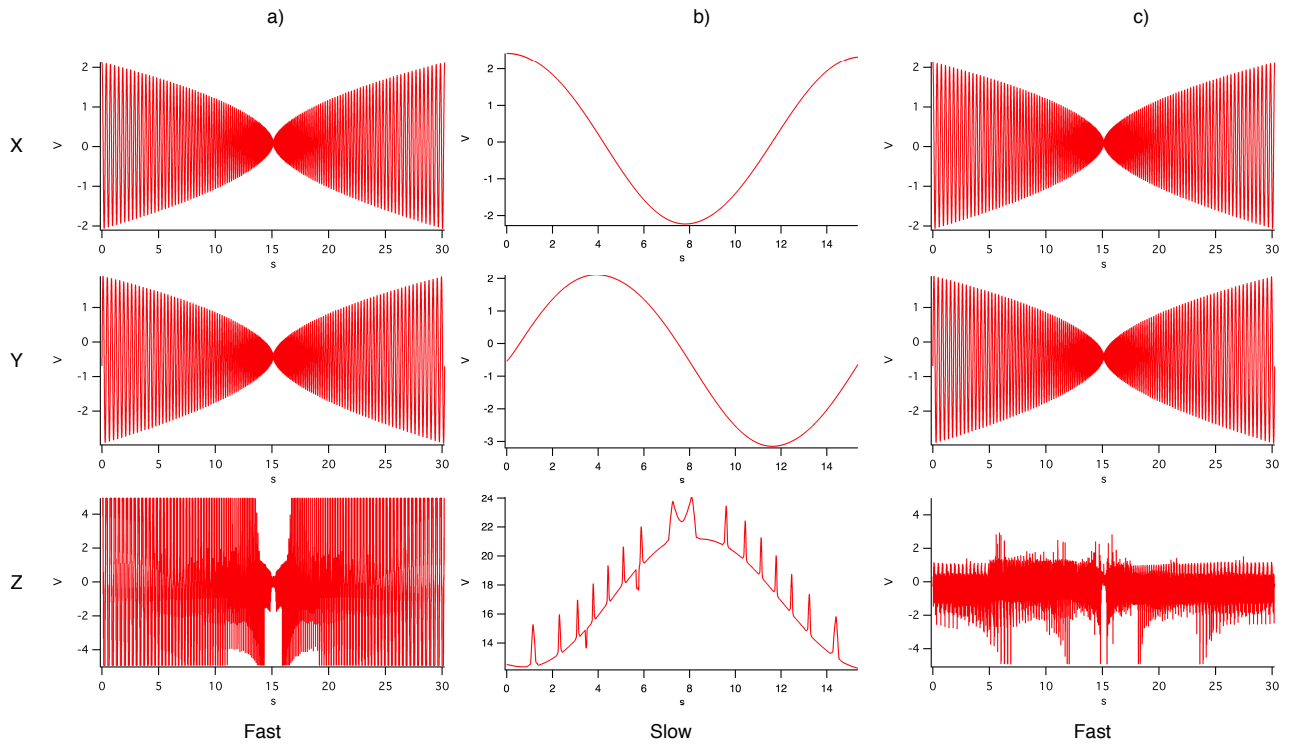


Figure 4.5: Path on the XY plane

a_1	a_2	a_3
-0.14615	-0.031882	29.537

Table 4.1: Plane fit coefficients

The figure 4.6 shows the wave we are going to send to the microscope. The x,y values of the wave's equation (3.2) are the output values of the scan pattern.

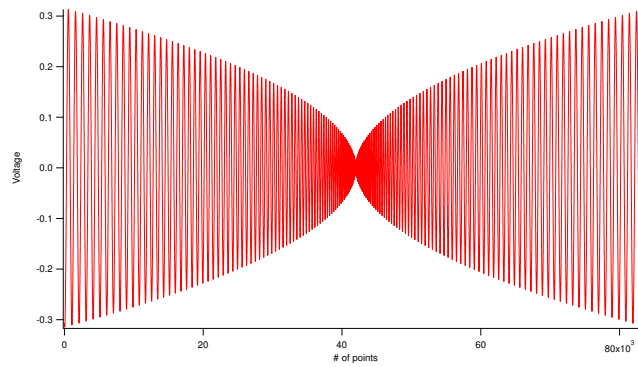


Figure 4.6: Input of the tilt compensation

We have calibrated the small piezoelectric ceramic and found that a step of 5V is equal to 90 nm.

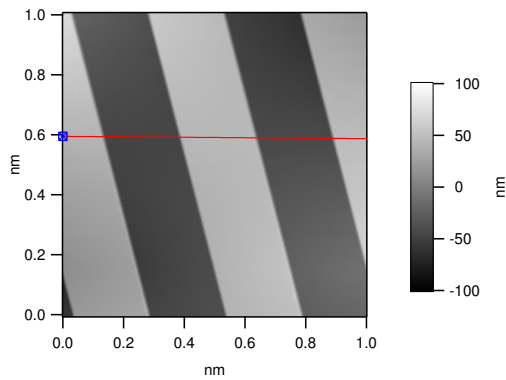


Figure 4.7: Height of the calibration

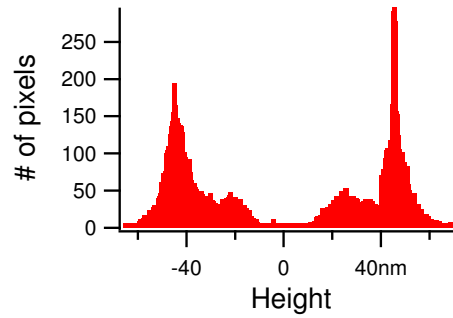


Figure 4.8: Histogram of the calibration

The tilt compensation takes a load off the small fast piezoelectrical ceramics. The Figure 4.9 shows the efficiency of our method. Indeed, the fast piezo was previously saturating. The piezo was trying to reach features that are larger than its range. If we use the tilt correction, we see that our piezo has no problem reaching those features.

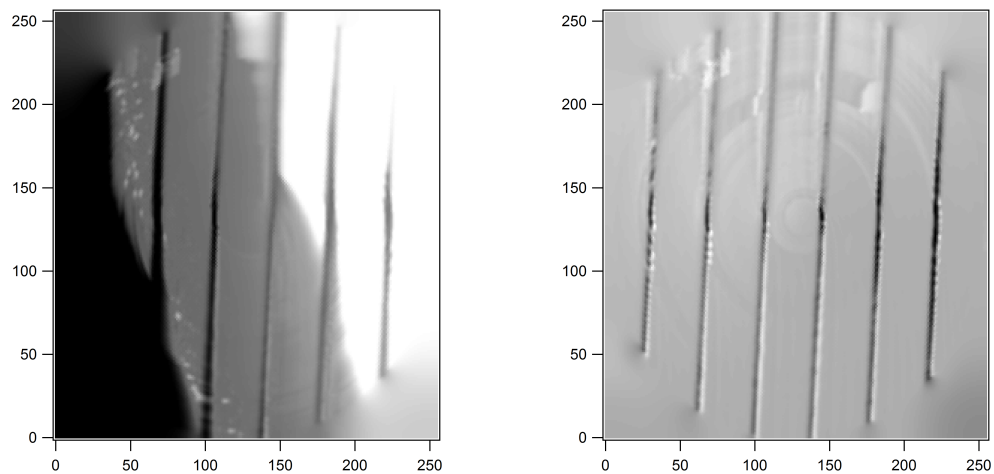


Figure 4.9: Before and after the tilt correction

5 Conclusion

In this thesis, we saw different methods to improve AFM bandwidth and speed on the X,Y,Z axis. Also, we have developed new techniques to render images.

A Programming

A.1 Structure of the Spiral Scan program

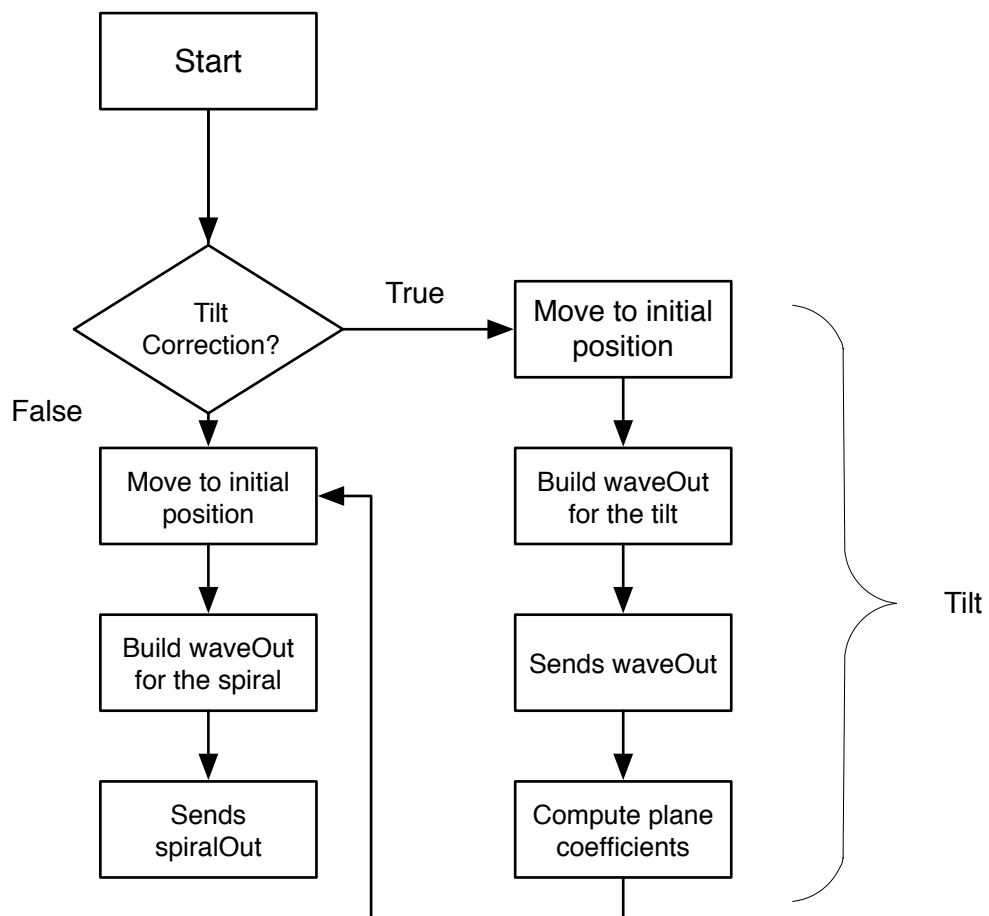
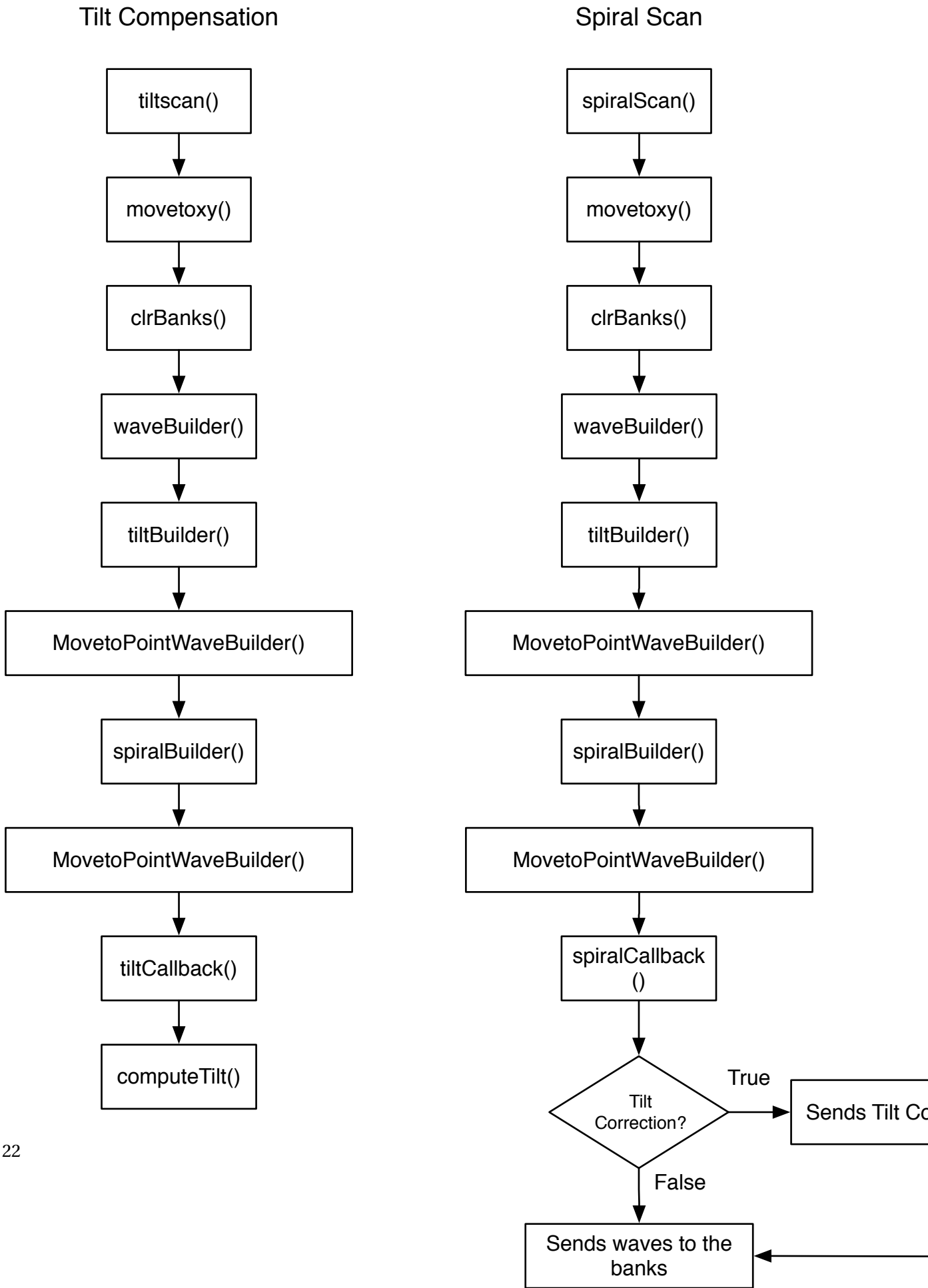


Figure A.1: Flowchart of the spiral scanning program



A.2 User interface Igor Pro

SpiralScanPanel

Scan Settings

Update while Scanning ☒ Controller Frequency 100000

Scan Size 1.00 μm Scan Type ErfSpiral

Scan Angle 90.00 $^\circ$ # of Loops 0100 Time Per Scan 1.00 s

X Offset 0 nm NumPntsOut 50176 InptRdctnFctr 2^A 1

Y Offset 0 nm NumPntsIn 50176 WRF 2

Use Sliding ☐ Spiral Alpha 0.30 DisplayR

Spiral Beta 0.30 DisplayTheta

Display Settings

Scan Pattern XY Sensor XY Sensor Meter **MeasuredVelocity** ☒

PDR Density Tilt Path Avg. Pixel Dist Ratio 262172

Angular Velocity Velocity Points&Lines 64 Avg. Velocity [$\mu\text{m/s}$] 313.057

DisplayChannel 1 Deflection DisplaySelector BothSepar...

DisplayChannel 2 Amplitude Calib Noliac(nm/V) -3.60 m/V PlaneFit ☒

Feedback Settings

ClosedLoopXY(Beta) ☐ FeedbackType Asylum Imaging Mod Contact

Use XY Sensor Data ☐ Integral Gain 5.00 Set Point 1.000 V

Scan Control

Engage MeasureTiltCoefficients Do Scan STOP

X 2.16e-17 Y 1.85e-17

ApplyTiltCorrection ☐

Save

Name CA_150nm_1p5s 1643 Reset Path...

Save ☐ SaveAICSV ☐ SavePNGs ☐ SaveIBW ☐

Only Active Window


Rename Save Color 

Figure A.3: User interface spiral scanning

A.3 Other figures

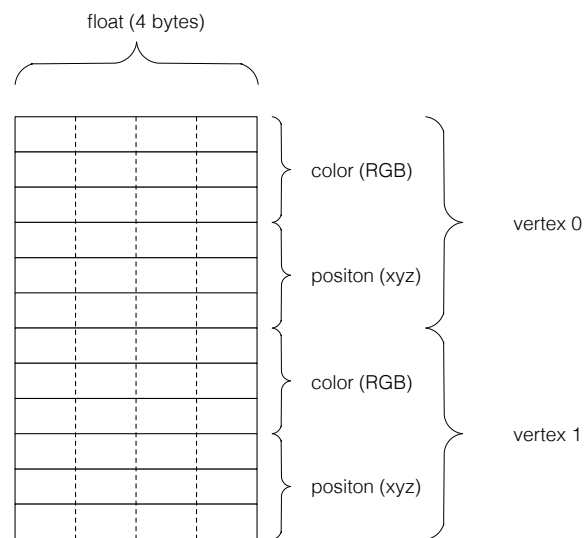


Figure A.4: Memory for the VAO

Bibliography

- [1] Abobe. Tips for optimizing gpu rendering performance. 2013.
- [2] Gilles Aubert and Pierre Kornprobst. *Mathematical problems in image processing: partial differential equations and the calculus of variations*, volume 147. Springer, 2006.
- [3] Gerd Binnig, Calvin F Quate, and Ch Gerber. Atomic force microscope. *Physical review letters*, 56(9):930–933, 1986.
- [4] Khalid El Rifai, Osamah El Rifai, and Kamal Youcef-Toumi. On dual actuation in atomic force microscopes. In *American Control Conference, 2004. Proceedings of the 2004*, volume 4, pages 3128–3133. IEEE, 2004.
- [5] PK Hansma, VB Elings, O Marti, CE Bracker, et al. Scanning tunneling microscopy and atomic force microscopy: application to biology and technology. *Science (New York, NY)*, 242(4876):209, 1988.
- [6] PE Hillner, AJ Gratz, S Manne, and PK Hansma. Atomic-scale imaging of calcite growth and dissolution in real time. *Geology*, 20(4):359–362, 1992.
- [7] Peng Huang and S.B. Andersson. Generating images from non-raster data in afm. In *American Control Conference (ACC), 2011*, pages 2246 –2251, 29 2011-july 1 2011.
- [8] Younkoo Jeong, G. R. Jayanth, and Chia-Hsiang Menq. Control of tip-to-sample distance in atomic force microscopy: A dual-actuator tip-motion control scheme. *Review of Scientific Instruments*, 78(9):093706, 2007.
- [9] Mark J. Kilgard. Modern opengl usage: Using vertex buffer objects well. Technical report, 2008.
- [10] Yong Liang, Donald R Baer, James M McCoy, James E Amonette, and John P Lafemina. Dissolution kinetics at the calcite-water interface. *Geochimica et Cosmochimica Acta*, 60(23):4883–4887, 1996.
- [11] IA Mahmood and SO Reza Moheimani. Fast spiral-scan atomic force microscopy. *Nanotechnology*, 20(36):365503, 2009.

Bibliography

- [12] John W Morse and Rolf S Arvidson. The dissolution kinetics of major sedimentary carbonate minerals. *Earth-Science Reviews*, 58(1&2):51 – 84, 2002.
- [13] OpenGL, 2012.
- [14] Jeanne Paquette and Richard J Reeder. Relationship between surface structure, growth mechanism, and trace element incorporation in calcite. *Geochimica et Cosmochimica Acta*, 59(4):735–749, 1995.
- [15] Manfred Radmacher. Measuring the elastic properties of biological samples with the afm. *Engineering in Medicine and Biology Magazine, IEEE*, 16(2):47–57, 1997.
- [16] Manuel M Oliveira Brian Bowen Richard and McKenna Yu-Sung Chang. Fast digital image inpainting. In *Appeared in the Proceedings of the International Conference on Visualization, Imaging and Image Processing (VIIP 2001), Marbella, Spain, 2001*.
- [17] Jonathan Richard Shewchuk. Triangle: Engineering a 2D Quality Mesh Generator and De-launay Triangulator. In Ming C. Lin and Dinesh Manocha, editors, *Applied Computational Geometry: Towards Geometric Engineering*, volume 1148 of *Lecture Notes in Computer Science*, pages 203–222. Springer-Verlag, May 1996. From the First ACM Workshop on Applied Computational Geometry.
- [18] Ryoji Shiraki, Peter A Rock, and William H Casey. Dissolution kinetics of calcite in 0.1 m nacl solution at room temperature: An atomic force microscopic (afm) study. *Aquatic Geochemistry*, 6(1):87–108, 2000.
- [19] T Sulchek, SC Minne, JD Adams, DA Fletcher, A Atalar, CF Quate, and DM Adderton. Dual integrated actuators for extended range high speed atomic force microscopy. *Applied physics letters*, 75(11):1637–1639, 1999.
- [20] YK Yong, SOR Moheimani, and IR Petersen. High-speed cycloid-scan atomic force microscopy. *Nanotechnology*, 21(36):365503, 2010.