AOS Software Manual

DM Controller
Hartmann Wavefront Sensor
Adaptive Optics

ACTIVE OPTICAL SYSTEMS
www.aos-llc.com

9019 Washington St. NE, Suite B2
Albuquerque, NM 87106
(505) 207-4202
www.aos-llc.com

Table of Contents

1 Introduction

The AOS Adaptive Optics software suite contains three major programs:

- DMController – Software to perform basic deformable mirror operations
- Hartmann Wavefront Sensor – Software to perform basic measurement operations with a Shack-Hartmann or Hartmann wavefront sensor.
- AO – Software to perform both metric and matrix adaptive optics by linking either a camera or a Hartmann sensor to the deformable mirror for beam shaping or active wavefront control.

All these sections may not be available to every user. Each user will only be able to operate sections of code consistent with their license. This software is designed to operate on Windows XP, Vista, and 7. Both 32 bit and 64 bit platforms are supported.

Note: This document is based on version 1.8.4 of the AOS Software. If you are using a newer version, some features may not be documented here and the screen shots may be different.

1.1 Select Warnings

Licensing: You will only be able to operate sections of software consistent with your license. You may not be able to operate all the sections of the code.
User-Servicing: The AOS hardware is not designed for any user servicing or cleaning. Any attempt at servicing the hardware may void the warranty, cause damage to the hardware, and/or may cause injury to the user.
High-Voltage: The deformable mirrors and drive electronics operate with high voltage. Make sure that all connections are made properly before powering up the devices. Do not attempt to probe the output voltages or use the drive electronics without a deformable mirror attached.
Manuals: Read all manuals before operating all the hardware and software. More information may be available on the AOS web site (www.aos-llc.com) via updated manuals or application notes.
Non-English Windows Operating Systems: The software is designed for US English Windows Operating Systems. Many of the data files for the AOS software use comma separated values (CSV) with decimal numbers. Some foreign operating systems switch the characters such that commas and periods mean different things. If using a foreign operating system, the meaning of commas and periods in numeric values may need to be switched before the software will operate properly.

1.2 Quickstart Guide

For users that want to get started quickly, below is the general procedure for getting started. For more details, please consult the more detailed sections of the manuals. Operation of the devices in a way other than outlined in the manuals may cause damage to the device and void the warranty.

1. **Install the AOS software.** Do this before attaching any hardware to the system so that the drivers are available. Do not install any versions of the software other than those provided by AOS because they may not work with our software.

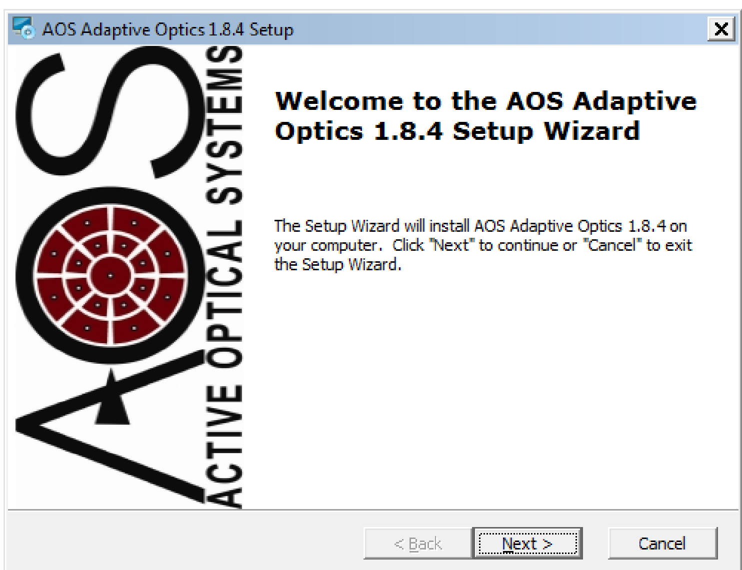
2. **Setup the optical system.** Note that the deformable mirror will have some small amount of static aberration that will need to be removed actively, so the beam reflected from the deformable mirror into the wavefront sensor will not be flat.
3. **Make the electrical connections to the hardware.** Be very careful because making these connections can misalign the optical system if proper care is not taken. Make the power connections LAST.
4. **Turn on the power.** The computer should recognize the hardware and prompt you to install the proper drivers.
5. **Run the AOS software to operate the device.**

2 Software Installation

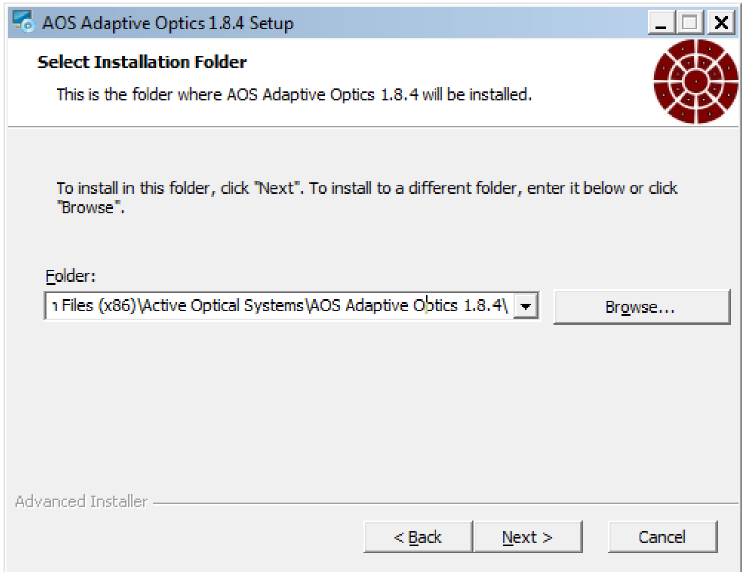
2.1 AOS Adaptive Optics Software Suite

Do not connect any of the AOS hardware to the computer before installing the software because the drivers will not be ready for proper operating system interpretation. Insert the AOS software disk into the computer. Open the disk in an explorer window and run the installation program appropriate for your platform (32 or 64 bit). Upon running the installer, the following windows will appear:

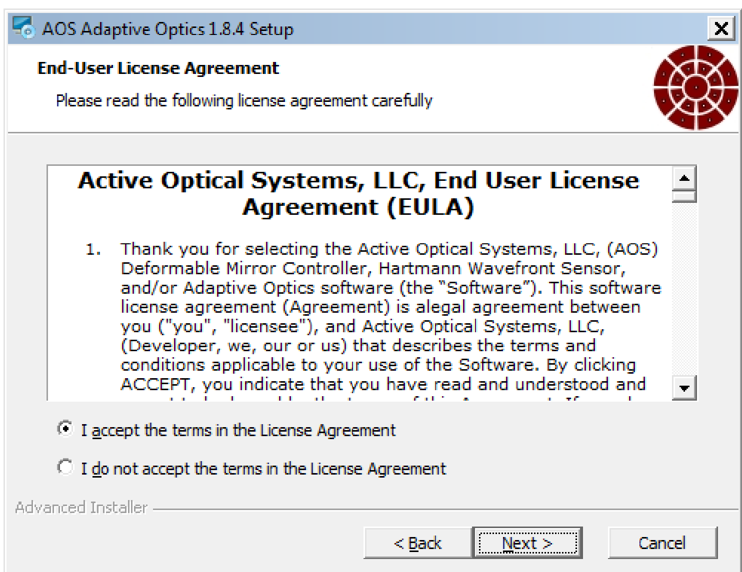
The installer will lead you through an installation of the AOS Adaptive Optics software suite. Press Next to continue.



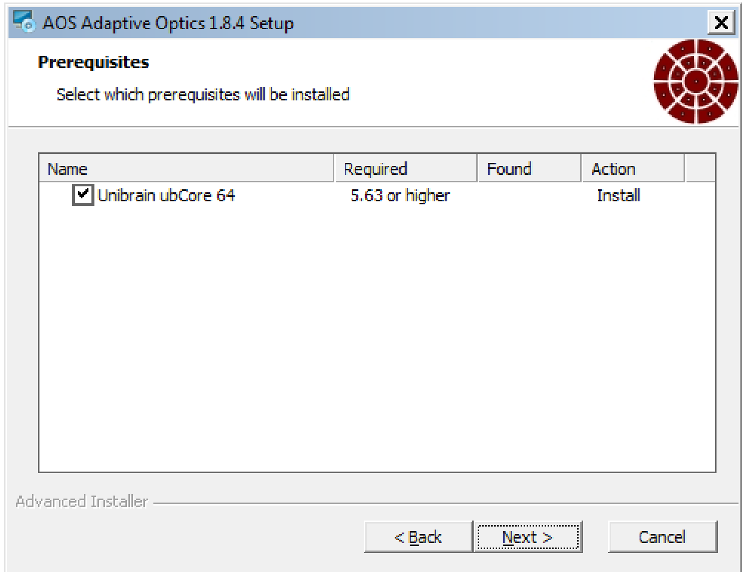
The following screen will allow you to change the installation location. It is recommended that you accept the default installation location. Click Next to continue.



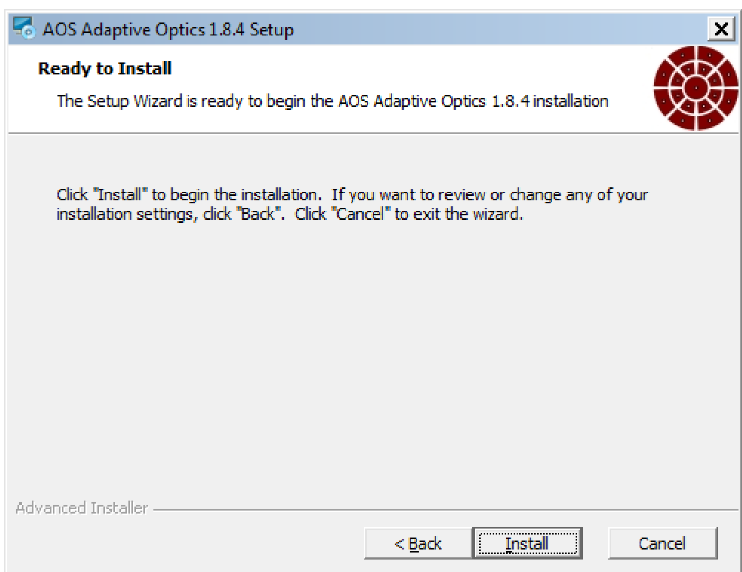
The following screen will require you to agree to the end-user license agreement (EULA) that is displayed on the screen. Click Next to continue.



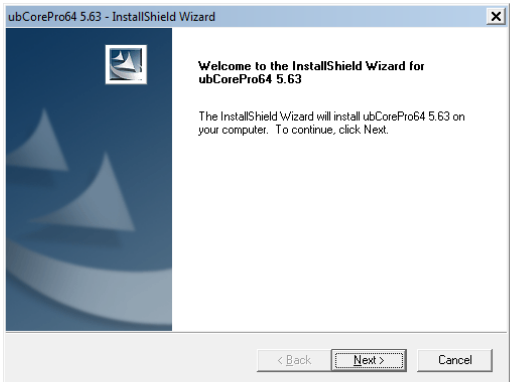
The next window will display the status of required prerequisites. In the case below, the setup program detected that the Unibrain ubCore needs to be installed. It is generally recommended that these settings not be changed. After reviewing the prerequisites, click Next to continue.

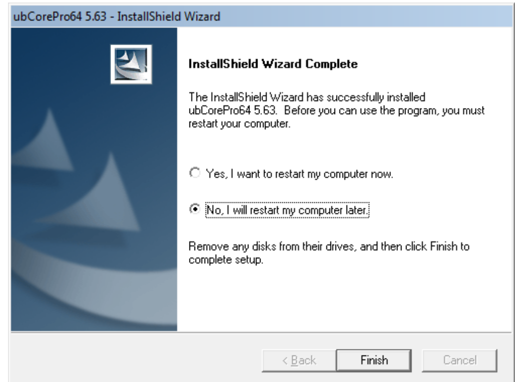


The setup program is now ready to begin the installation. Click next to begin.

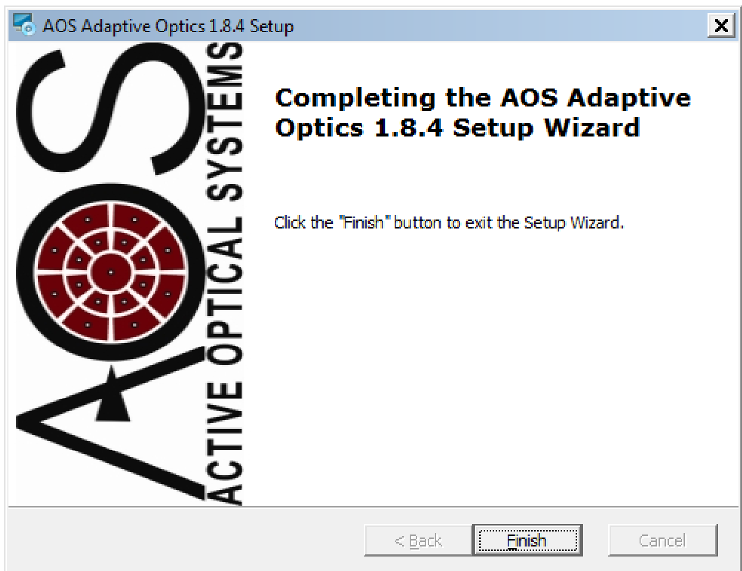


If the Unibrain camera driver prerequisite needs to be installed, the setup program may launch the Unibrain ubCore installer. Please follow the instructions in the installer, accepting all default options. At the end of the Unibrain ubCore installation, choose the option “No, I will restart my computer later” and click Finish.

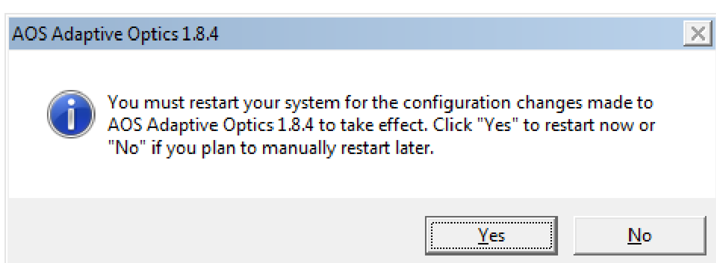




The following screen will appear when the installation is complete. Click Finish to exit the installer.



It is highly recommended that you restart the computer after the installation has completed. Click Yes to restart the computer.



2.2 *Driver Installation*

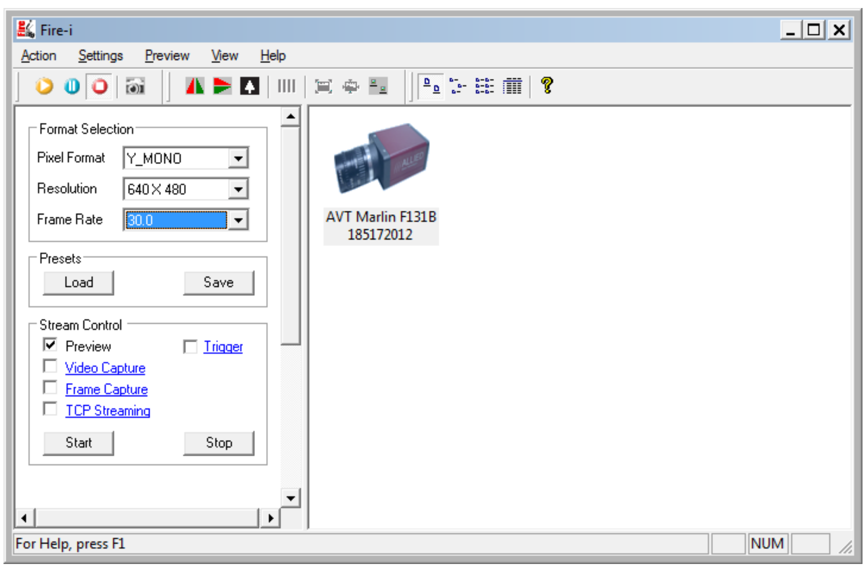
To complete the installation, you will need to install the appropriate hardware drivers for your hardware. Below is a guide on how to install most of our standard driver suites.

2.2.1 FireWire Camera Setup

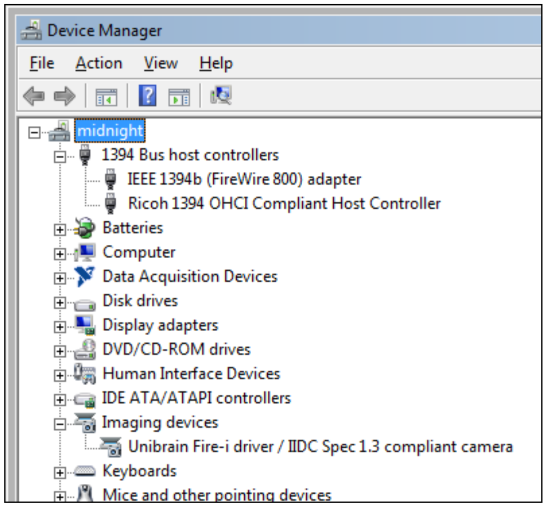
> **NOTE:** The CMU FireWire camera driver is no longer actively supported by AOS primarily due to lack of 64 bit platform support. While the CMU interface currently still works with the AOS software, support for it may be dropped in future versions. Therefore, it is recommended that users switch to the Unibrain driver.

2.2.1.1Driver Installation

After the AOS Adaptive Optics Software has been installed, connect the wavefront sensor camera to the computer. Open the Unibrain preview application to verify the presence of the camera (Programs → Unibrain ubCore → Fire-i Application).



If the camera is not shown, ensure that the correct driver is installed in Windows Device Manager. The camera should be listed as a Unibrain device under Imaging Devices. If it is not, right click on the device and update the driver software to the Unibrain driver (Windows should pick the Unibrain driver automatically).



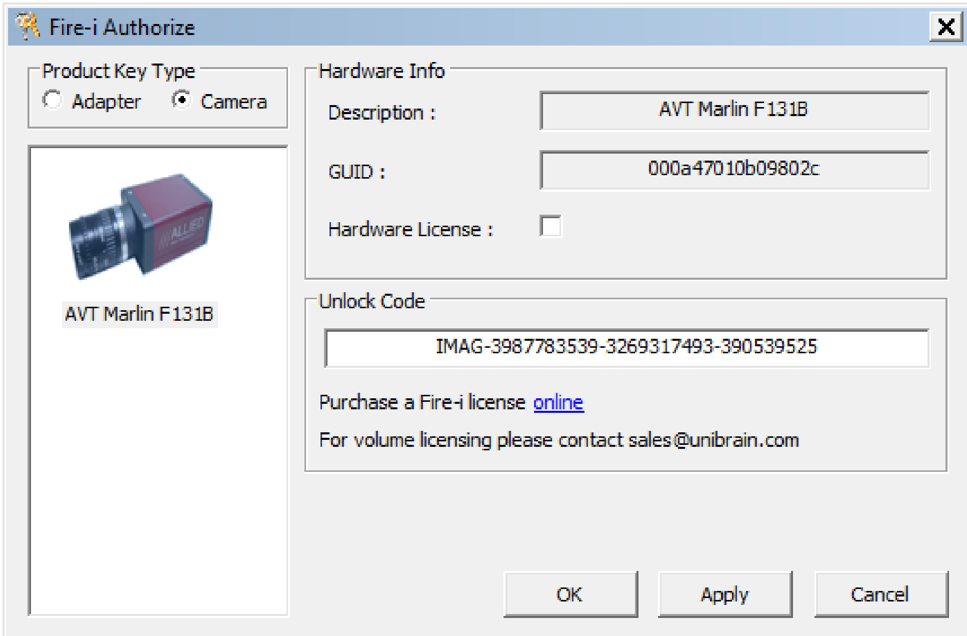
2.2.1.2Driver Licensing (Unibrain-Versions <2.0)

NOTE: Versions of the software beyond 2.0 moved to the AVT camera drivers and do not require this step.

The Unibrain driver requires a runtime license that will be provided by AOS. The license is tied to the wavefront sensor, so you may install it on multiple computers. To install the Unibrain runtime license, run the Unibrain licensing utility:

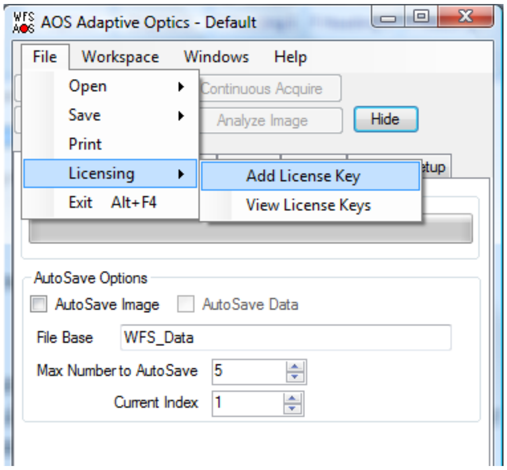
Programs → Unibrain ubCore → Fire-i Authorize

Select "Camera" for product key type. The wavefront sensor camera should be shown on the left. Select the camera and paste the AOS provided Unibrain runtime license key into the Unlock Code field and click Apply. A message should be shown indicating successful installation of the license key. Click OK to exit the window.

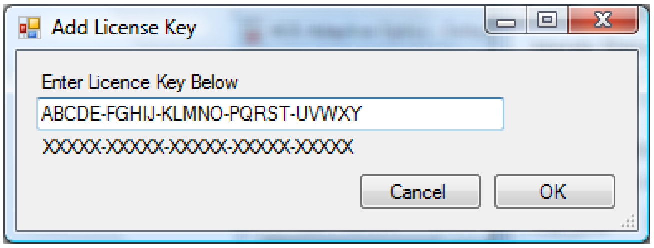

Fire-i Authorize
Product Key Type
Adapter
Camera
AVT Marlin F131B
Hardware Info
Description :
AVT Marlin F131B
GUID :
000a47010b09802c
Hardware License :
Unlock Code
IMAG-3987783539-3269317493-390539525
Purchase a Fire-i license online
For volume licensing please contact sales@unibrain.com
OK
Apply
Cancel

2.3 AOS Software License Key Installation

The AOS software is locked to your wavefront sensor camera. You should have received a license key with your AOS software. If you require a key, please contact AOS. Run the AOS software from the desktop shortcut or Windows start menu shortcut. To enter your license key, from the main window select File→Licensing→Add License Key.



Enter your license key in the following window and click OK.



If your license key is valid, your licensed features should become available in the AOS software. Note: you may need to select your licensed wavefront sensor camera on the camera tab to enable all features of the software.

2.4 USB Software Key (Version 2.0.3+)

Beyond version 2.0.3, users who purchase a wavefront sensor or the AO software license will receive a USB key like the one shown below with their software. The drivers for this key must be installed in addition to the AOS software. This key must be attached to a USB port of the computer running the software to enable features of AO and wavefront sensing. This key is enabling future versions of the software to have an improved API and more flexible camera options. Users who purchased hardware earlier than version 2.0.3 should contact AOS about obtaining a key.

2.5 USB Drive Electronics Driver Installation

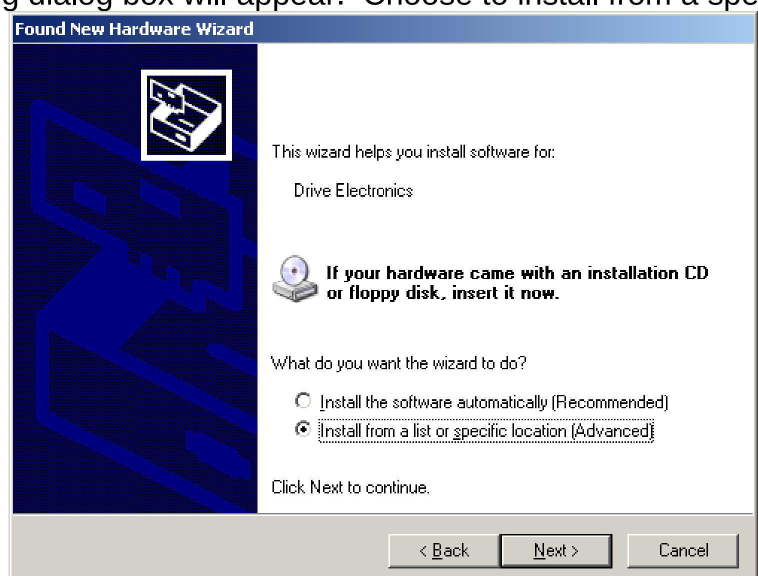
The AOS drive electronics driver installation will require you to connect the hardware to the computer before it completes. Some of the drivers will automatically install when connected to the computer, but some require some user intervention so that the correct driver is used. In this section, we describe some of the procedures required to get the proper driver installed. If you have any difficulties with this or any step in the installation, contact AOS for help (support@aos-llc.com).

2.5.1 USB Drive Electronics Driver Installation Completion

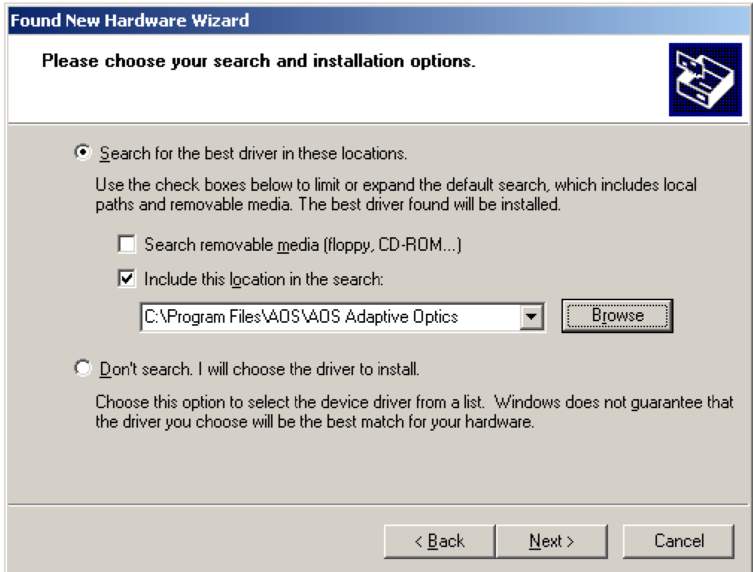
Upon connecting the AOS USB Drive Electronics to the computer, the following message will probably appear in the icon tray on task bar:

Then the following dialog box may appear. Choose “No, not this time”.

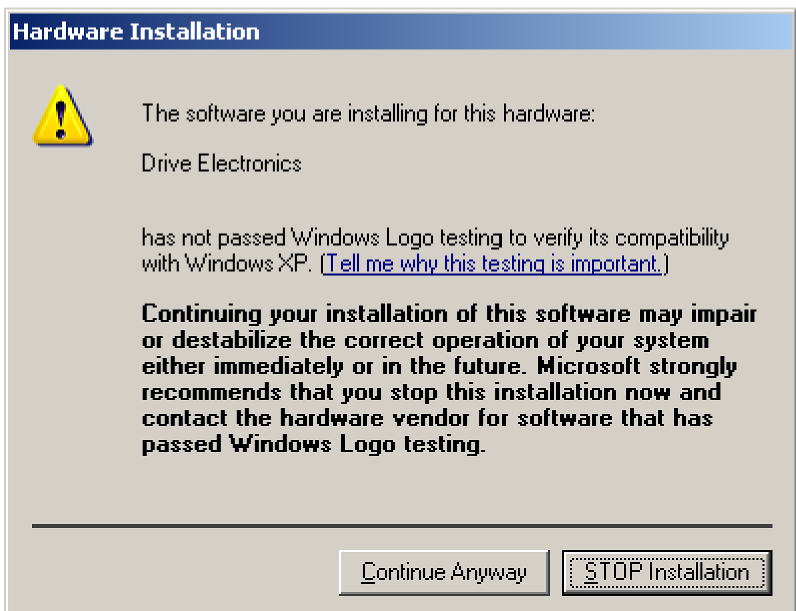
Then the following dialog box will appear. Choose to install from a specific location.



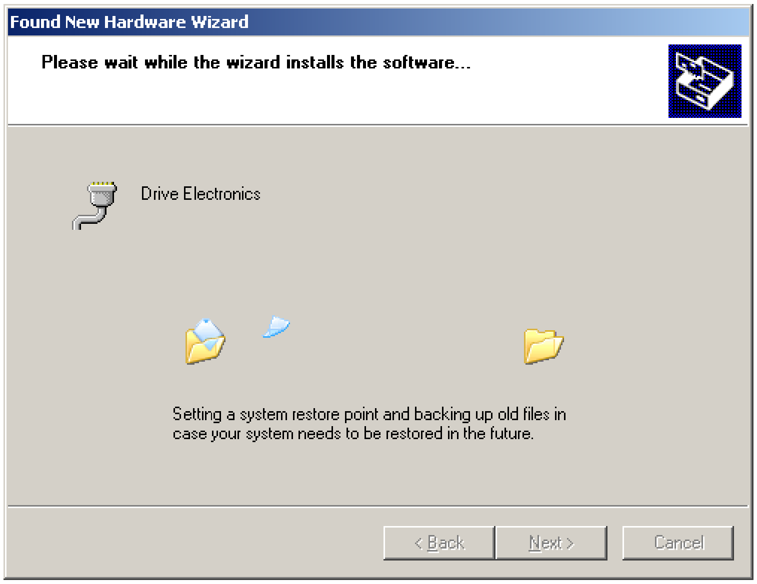
When the following dialog box appears, do not search removable media. Instead, specify the location “C:\Program Files\AOS\AOS Adaptive Optics” and select “Next”.



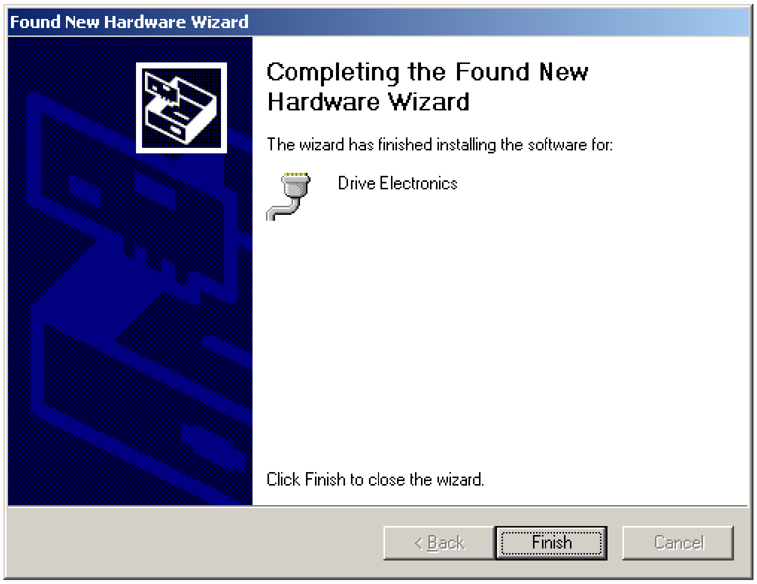
Windows will then warn you that this driver has not passed Windows Logo testing. Select “Continue Anyway”.



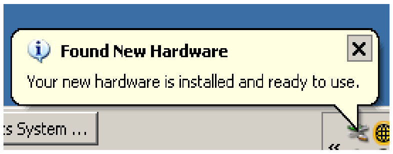
The drivers will now be installed. The following dialog box will appear.



When the installation is complete, the following dialog box will appear. Select “Finish”.



A pop-up from the task bar (shown below) will appear to inform you that the hardware is ready to use.



2.6 Hardware Setup

Please refer to the AOS application notes and hardware manuals to get more details on how to setup the hardware electrically and optically. Setup all the hardware before running the software.

3 Hartmann Sensor Software

The Hartmann Sensor software package enables the user to interpret the data from an AOS Hartmann sensor. This section describes the options in the user interface to use the Hartmann sensor. The sections below are broken down by the tabs in the Hartmann sensor software.

3.1 Main Window

The main window, shown in Figure 3.1, allows the user to load and save individual files and control acquisition from the sensor. The Open menu item allows the user to load images or data from files. The data can be loaded and displayed or just the calibration data can be loaded from a data file. The Save menu item on the File menu allows the user to save images, data, or both. The Print menu item prints a short set of images from the Wavefront, Image, and Slopes windows. The Workspace menu provides a convenient way to manage saved data. The Windows menu allows the user to toggle the appearance of any of the display windows and the DM Controller. The Help menu provides software information. General help and information is available on the AOS web site (www.aos-llc.com) via updated manuals or application notes, or AOS can be contacted for help at (support@aos-llc.com).

The buttons below the main menu allow the user to acquire a single frame, create a new reference from data in memory, continuously acquire frames, or re-analyze data in memory after changing some analysis setting. The "Hide" button allows the user to hide or show the analysis tab sheets below the buttons to maximize display real estate.

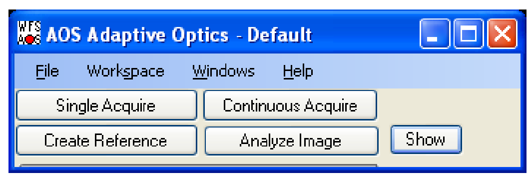


Figure 3.1 – Main Window with hidden tabs

In order to make an accurate measurement with the Hartmann sensor, it is important to make sure that the Hartmann sensor's imager is not saturated. There is some ability for the user to control the brightness and gain in software, but in extreme cases, this will have to be addressed in hardware via absorbing neutral density filters. Do not use reflecting neutral density filters! They will cause the Hartmann spots to be corrupted such that an accurate measurement cannot be made. To adjust the brightness and gain, we recommend the following procedure:

1. Make sure a valid calibration file is loaded.
2. Select the Camera tab.
3. Press the Continuous Acquire button.
4. Press the Setup button to open the camera properties window.
5. On the Basic tab
   a. Adjust Black Level to 16
6. On the Exposure tab.
   a. Reduce the Gain control to a low level. (To make adjustments, make sure the control is toggled to manual, not automatic.)
   b. Adjust the Shutter control until the desired image is obtained

7. Close the camera properties window and stop the continuous acquisition.

Our recommendation is to run at the lowest gain possible, especially for red and infrared (IR) wavelengths due to bleeding that can occur in some kinds of cameras.

3.1.1 Setup Tab

Figure 3.2 shows the Setup Tab in the HWFS software. The Sensor Setup box allows the user to specify the separation length (distance from the Hartmann array to the imager), and the X and Y pixel sizes of the camera. The separation length is different for every sensor since these devices are assembled by hand, but will be provided by AOS in the reference file for the sensor. The X and Y pixel sizes are also in the AOS-provided reference file. The separation and pixel sizes are automatically adjusted during the loading of a calibration file and should not be adjusted by the user.

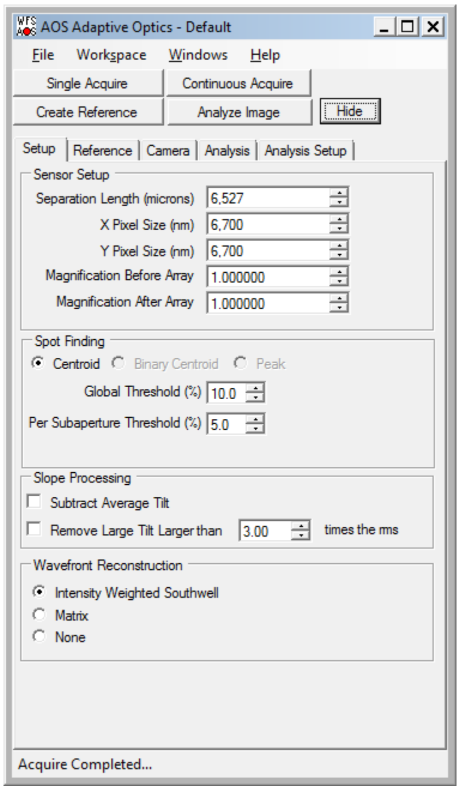


Figure 3.2 - Setup Tab

3.1.1.1Spot Finding

The Spot Finding box allows the user to control how the positions of the diffracted spots are determined. The most common algorithm is to calculate the first moment or centroid of the diffraction pattern, which is given by

$$\bar{x} = \frac{\sum_{i=i_{\min}}^{i_{\max}} \sum_{j=j_{\min}}^{j_{\max}} I(i,j)\cdot x(i,j)}{\sum_{i=i_{\min}}^{i_{\max}} \sum_{j=j_{\min}}^{j_{\max}} I(i,j)}$$

where i and j refer to the pixel indices, I(i,j) is the intensity measurement, x(i,j) is the x pixel coordinate, and i_{min}, i_{max}, j_{min}, and j_{max} are the coordinates of the area of interest (AOI) that define the pixel region behind each sub-aperture.

Before calculating the centroid, the intensity pattern is often thresholded to reduce the effect of noise in the signal on the measurement. This threshold is applied in two different ways in the software. There is global threshold that is applied to all the sub-apertures. Then there is a per subaperture threshold that is determined based on the maximum value of the intensity in each sub-aperture. The per subaperture threshold is useful when there is significant variation in intensity over the image or significant dynamic intensity variation. The thresholding is usually applied by subtracting the threshold value from the intensity and then zeroing the values that are less than zero.

There is an option for higher speed spot finding leveraging better optimized code. This option only uses the global threshold.

3.1.1.2 Slope Processing

The Slope Processing box offers the user two different options: average tilt subtraction and slope filtering. The average tilt subtraction allows the user to remove the average tilt of the measured slopes. The slope filtering removes the slopes with magnitudes larger than a user-selectable multiple of the rms slopes. This is done to remove poor slope measurements.

3.1.1.3 Wavefront Reconstruction

The Wavefront Reconstruction box offers three options for reconstruction. The default reconstructor is an iterative Southwell-type reconstruction that involves calculating the phase at each point inside the edge as the average of the adjacent slope projections, or

$$\phi_1(i,j) = \phi(i-1,j) + \left[\frac{\frac{d\phi}{dx}(i-1,j) + \frac{d\phi}{dx}(i,j)}{2}\right] \bullet \Delta_1$$

$$\phi_2(i,j) = \phi(i+1,j) + \left[\frac{\frac{d\phi}{dx}(i+1,j) + \frac{d\phi}{dx}(i,j)}{2}\right] \bullet \Delta_2$$

$$\phi_3(i,j) = \phi(i,j-1) + \left[\frac{\frac{d\phi}{dx}(i,j-1) + \frac{d\phi}{dx}(i,j)}{2}\right] \bullet \Delta_3$$

$$\phi_4(i,j) = \phi(i,j+1) + \left[\frac{\frac{d\phi}{dx}(i,j+1) + \frac{d\phi}{dx}(i,j)}{2}\right] \bullet \Delta_4$$

$$\phi'(i,j) = mean(\phi_1,\phi_2,\phi_3,\phi_4)$$

The edge sub-apertures do not have four phase measurements, but instead use fewer as measurements are available. The user has the option of combining the four phases as an intensity weighted sum instead of a simple mean by selecting the Intensity Weighting check-box.

Another option which is still under development is the matrix reconstruction (Matrix Recon). If Matrix Recon is checked the program will establish a matrix reconstructor for the wavefront sensor by inverting a simple adjacent neighbor gradient calculation matrix. This reconstructor is very fast, but is still in its early development stages, so we recommend that users use the Southwell reconstructor wherever possible. For debugging purposes, the program will write-out the reconstructor to a text file in the program directory when it is generated.

The Zernike option allows the user to fit the slopes to the derivative of the Zernike polynomials and construct the wavefront as a sum of the selected Zernikes. The number of Zernikes is selected in the Analysis Setup tab.

Finally, wavefront reconstruction may be disabled by selecting “None.” This is useful for increasing the acquisition rate. The wavefront slopes will still be computed when reconstruction has been disabled.

3.1.2 Reference Tab

Figure 3.3 shows the Reference Tab where the user has three options for translating an image into a reference. Each reference is comprised of a set of areas of interest or AOIs. An AOI is defined by its opposing coordinates.

3.1.2.1AOI Determination

The Peak Reference algorithm sums the measured intensity pattern in the two axes and then determines the peak locations by looking for points with intensity greater than its immediate neighbors. There is some code to reject the local peaks by only allowing the highest amplitude peak in an area approximately the size of the user specified aperture pitch. AOIs are then created around the peaks with the size specified by the user in the Aperture Pitch input box. In cases where the data is clean, this algorithm works well, but it can fail in regions of excessive noise or when the aperture pitch is set incorrectly. The Threshold is not used in this algorithm. The Auto checkbox next to the Aperture Pitch input box uses a data analysis technique to automatically guess the aperture pitch in pixels. If this is known (the sub-aperture spacing divided by the pixel size), the user can specify the aperture pitch directly, but this Auto option is useful when the incident beam’s wavefront is not nearly planar.

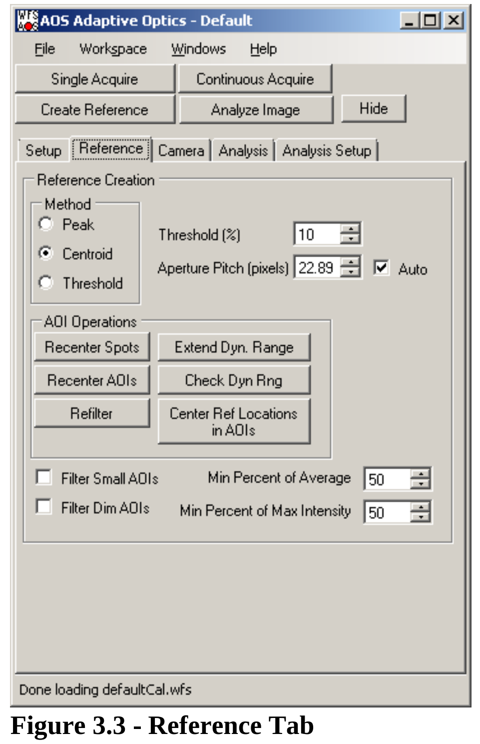


Figure 3.3 - Reference Tab

The Centroid Reference algorithm starts with the peak reference algorithm to establish an initial set of AOIs, but then moves all the AOIs around together to try to place the centroid in the center of the AOIs. This algorithm is often better than the Peak Reference algorithm, but can also occasionally fail. The Threshold is not used in this algorithm.

The Threshold Reference algorithm begins by summing the intensity profiles in the two axes to reduce the intensity data to vectors. Then the vectors are scanned for up transitions and down transitions of the intensity about the threshold specified by the user in the Threshold box to the right. The center location of each AOI is at the average of the location of the up and down transition of the 1D intensity profile. Finally it checks for low-intensity AOIs by summing the intensity values above the Absolute Threshold specified in the Setup Tab and eliminating those with no intensity points above that threshold. Although all the above algorithms work, we recommend using the threshold reference algorithm.

After a set of AOIs are established, the reference centroid locations are determined and the neighbor mapping required for the Southwell reconstructor is established.

3.1.2.2Filtering AOIs

There are several filtering techniques that the user can use to reject poor AOIs. When the Filter Small AOIs option is checked, the software establishes the average AOI size and then eliminates those below the user-specified percentage of the average. This option is good for automatically rejecting AOIs at the edge of the sensor that might only have a fraction of them on the sensor.

The Filter Dim AOIs option causes the program to establish the intensity in each of the AOIs and then remove AOIs below the user-specified percentage of the maximum measured AOI intensity. This option is good for rejecting poorly illuminated AOIs that sometimes appear at the edge of an aperture.

One further option for manually filtering AOIs is to right click on the AOI in the Intensity Window and then select Delete this AOI from the pop-up menu. The user can now also right click and drag a circle and rectangle which will allow for the deletion of a group of AOIs inside or outside the circle or rectangle.

3.1.2.3AOI Options for Reference Spot and AOI Manipulation

We added several options to the software to enable the user more control over the references. We recommend that the existing calibration be saved before trying any of these steps.

The Recenter Spots button allows the user to use the existing AOIs, but to establish new reference spot locations. This is very useful when making a new differential measurement or to quickly remove residual drift from the system.

The Recenter AOIs button is similar moves the AOIs to center them on their existing spots while maintaining their size. This is useful if making a measurement where the spots are drifting toward the edge of their AOIs due to a large aberration. This option will only work when the spots are all still within their AOIs.

The Refilter button allows the user to apply the previously described filters to the current set of AOIs. This is useful when a previously established reference (AOIs) needs to be used on a new beam that does not illuminate all the reference AOIs or the user forgets to turn-on filtering during the reference creation.

The final two buttons in the AOI Options box are an implementation of a new algorithm invented at AOS for attempting to extend the dynamic range of a Hartmann sensor in the presence of the most common three large aberrations: x-tilt, y-tilt, and focus. This algorithm is not fool-proof, but has been shown through experimentation and in the lab to work in most cases. We recommend that the user save their data before using this option because it may deleteriously affect the existing AOIs. To determine whether the sensor detects any problem with the sensor's dynamic range, the user can select the Check Dyn Range button. The program will report in a dialog box whether the sensor is near or beyond its dynamic range. If the sensor is in this condition, the user can click the Extent Dyn. Range button to have the software search for a new set of AOIs and map those AOIs to the existing set so that the user can make a measurement of an aberration with large focus or tilt.

3.1.3 Camera Tab

Figure 3.4 shows the Camera Tab. In previous versions of the software, we supported DirectX cameras, but we have moved away from the unreliable DirectX interface to a FireWire and Camera Link camera interface.

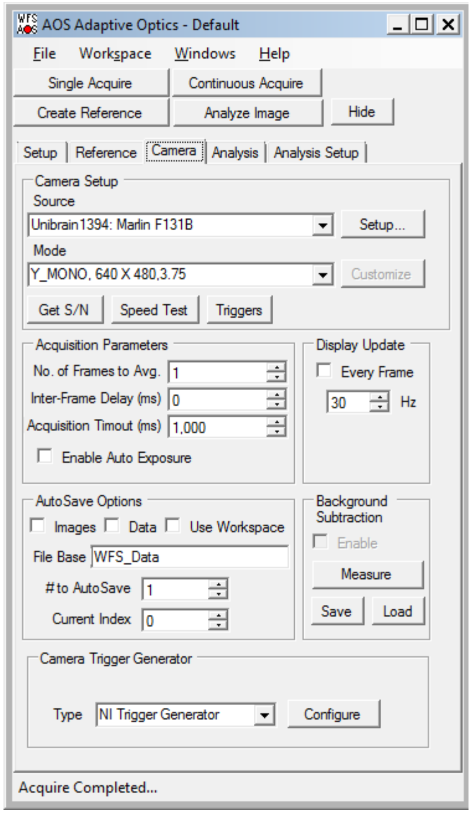


Figure 3.4 - Camera Tab

3.1.3.1 Camera Setup

The Camera Setup group box allows the user to select the camera if multiple cameras are connected to the system. The user can also click “Setup” to set the camera acquisition properties (Exposure, Black Level, etc...). We recommend that the user lower the gain to the minimum setting, and adjust brightness, auto exposure, and shutter to bring the camera to just below saturation (as shown on the Intensity Bar). Also, the acquisition mode may be selected in the Mode drop-down.

A file-based simulated camera is provided for playback of simulated or recorded data. To use the simulated camera, choose “Simulated: File Camera” from the Source drop-down. The Setup button will bring up the Simulated Camera Control window (Figure 3.5). The file names of images that will be frames for the simulated camera appear in the list on the right. The Controls group can be set to traverse the file list once, or loop over the list repeatedly.

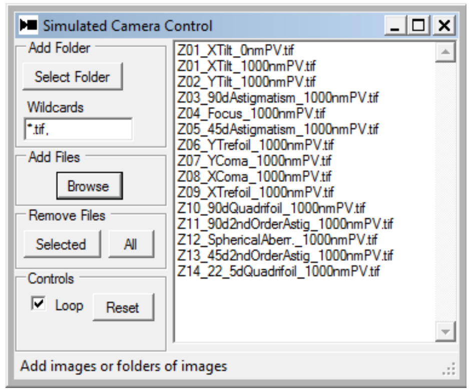


Figure 3.5 – Simulated Camera Control

The frame size and shape on some higher-end cameras can be adjusted by clicking the Customize button (Figure 3.6). Custom frame sizes are only supported for video modes containing the word “Max” in the mode name (i.e. Y_MONO Max 1280 X 1024). For certain applications, reducing the frame dimensions can result in a significant increase in acquisition rate at the cost of reduced spatial resolution. Customizing the frame will cause the previously-used calibration to be invalid, so the user will need to create a new reference or load an appropriate reference.

The “Auto Set Pixel Size” checkbox uses a look-up table to automatically set the pixel sizes based on the camera selected.

The “Get S/N” button returns the camera serial number to the user in a dialog box.

The “Speed Test” button acquires a set of frames with no processing to test the maximum frame rate of the camera with the current settings.

The “Triggers” button is discussed in the Triggering section below.

3.1.3.2Acquisition Parameters

The Acquisition Parameters change how the HWFS software acquires data from the camera. The first parameter determines the number of frames to average per reported result. During averaging, the intensity profile is averaged as a floating point number to create an image with less camera noise. Inter-Frame Delay (ms) determines the time period between frames if a Continuous Acquire command is given. Changing this value can be useful if the deformable mirror (DM) takes multiple frames to settle. For example, if a DM takes 4ms to shift to its commanded shape, the Inter-Frame Delay should be set above 4ms to prevent images being recorded of the DM in transition. Also, changing Inter-Frame Delay to a very high number of milliseconds, such that the camera records a frame every few minutes, can also be useful to perform long term data analysis. An example of long term analysis is the study of DM drift.

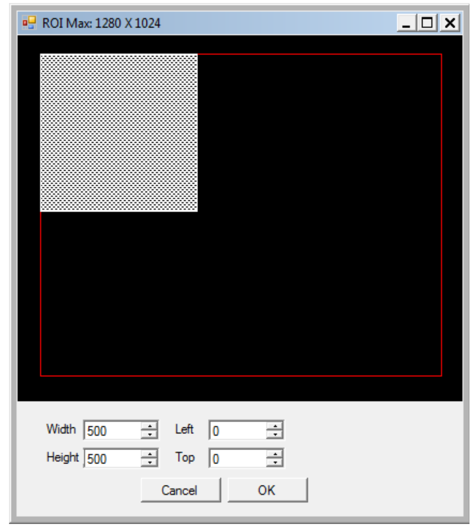


Figure 3.6 – Custom Frame Size

3.1.3.3AutoSave

The AutoSave Options box allows the user to setup images and/or data to be saved during acquisition. This will work for both Continuous Acquire and Single Acquire modes. The check boxes are used to specify what the user wants to save (data and/or images). The file base is the first part of the file name. The index number is appended onto the file name before the appropriate extensions are applied. The user can set the maximum number of images to auto-save and where to start the indexing. If the Use Workspace checkbox is checked, files are saved to the WFS (wavefront sensor) workspace folder:

My Documents\AOS\<Workspace Name>\WFS

If the Workspace checkbox is not checked, an absolute path to a save location may be specified:

C:\LogFolder\WFS_Data

3.1.3.4 Display Update Rate

When using high speed cameras or small regions of interest the frame processing time is often dominated by display drawing and updating tasks. Therefore, it is often useful to reduce the display update rate. This can be accomplished by setting the desired update rate using the rate control numeric up down control. If the Every Frame checkbox is checked, the display will be updated at each iteration, regardless of the update rate setting.

3.1.3.5 Background Subtraction

It is often useful to subtract a computed or measured background image from frames acquired from the wavefront sensor. This feature is useful if the camera sensor has a “hot” pixel that is affecting your measurement. To enable background subtraction, follow the steps below:

1. Click the Measure button to acquire a new frame to be used as a background image.
   OR
   Click the Load button to load a previously measured or computed background image.

2. Check the Enable checkbox to enable background subtraction in subsequent frames.

Note: If you are loading a previously measured background image, its dimensions must match the current acquisition frame size settings exactly.

3.1.3.6 Triggered Acquisition

For adaptive optics applications, it is often desired or necessary to operate the wavefront sensor camera in a triggered mode. All AOS wavefront sensors and intensity cameras except for the Fire-I Hartmann WFS support triggered acquisition. AOS currently supports configuration of the wavefront sensor camera for triggered acquisition as well as automatic generation of camera triggers using National Instruments GPIO capable hardware. Configuration for triggered acquisition consists of two primary tasks.

1. *Configure wavefront sensor camera for triggered acquisition mode (Figure 3.7).*

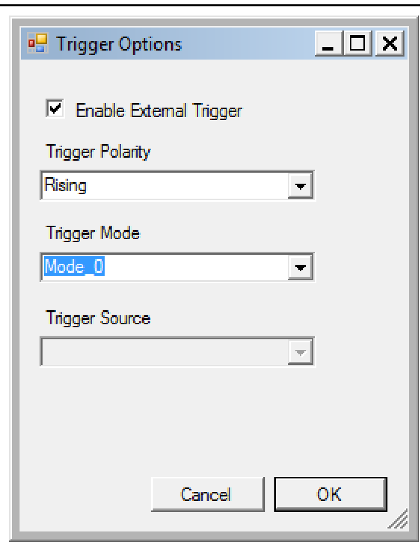


Figure 3.7 – Camera triggering options

This step can be accomplished by clicking on the Triggers button in the Camera Setup group box. Checking the “Enable External Trigger” checkbox will place the camera in a triggered acquisition mode. Set the desired polarity (rising or falling) and Trigger Mode. The available trigger modes depend on the particular camera, but generally Mode_0 is the most common selection (acquire on rising or falling edge depending on Polarity; ignore trigger signal duty cycle / pulse width). The trigger source selection box will only become active if more than one trigger source is available (internal to the camera).

NOTE: If the camera is configured for triggered acquisition and no trigger is received, the acquisition will time out. This timeout period may be adjusted in the Acquisition Parameters group box.

2. *Configure the trigger generator/source*

An external signal generator may be used to generate a trigger signal for the wavefront sensor. This is useful for synchronization of the AOS software with external events.

If an appropriate trigger source is available (may be obtained from AOS), the AOS Software may be configured to automatically generate a trigger signal for the camera. If desired, the trigger signal can be generated from an AO loop or just for acquisition from the wavefront sensor. To configure the internal trigger generator, launch the trigger generation configuration dialog by selecting a possible trigger source (by default NI Trigger Generator) and clicking “Configure.” The dialog in Figure 3.8 will be shown. Check the checkbox next to the trigger source to enable it. A list of valid ports will be available in the Port drop down list (if no ports are shown, either the trigger generation hardware is not available, or the proper drivers are not installed on the system). Select a trigger polarity matching the polarity selected for the camera in the previous step. The pulse length should generally be set to zero, but may optionally be set to a non-zero value to increase the length of the generated pulse.

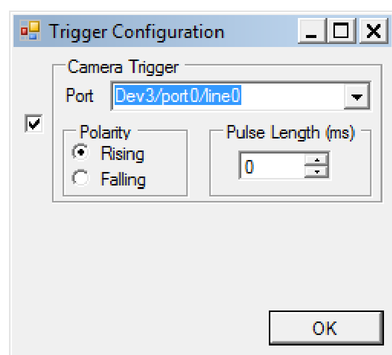


Figure 3.8 – Trigger generation configuration dialog.

After configuration of the camera and trigger source (internally generated or external), click Single Acquire to test settings. If the acquisition times out, the trigger may not have been received by the camera.

3.1.4 Analysis and Analysis Setup Tabs

Figure 3.9 shows the Analysis and Analysis Setup Tabs. The Analysis Setup Tab allows the user to specify what analysis they want done on the data and how that data is to be displayed. The Analysis Tab shows the user the results of their selected analysis.

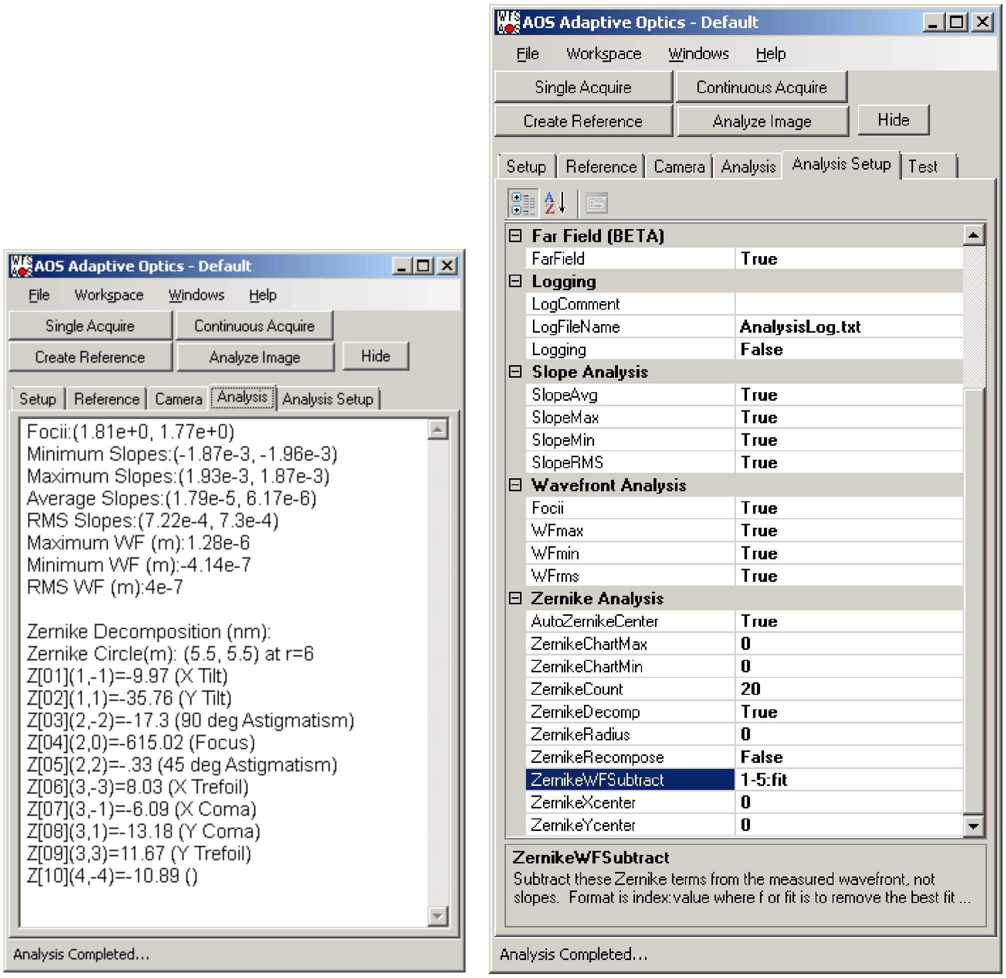


Figure 3.9 - The Analysis and Analysis Setup Tabs

Most of the analysis options are self-explanatory. Slope analysis includes calculations of the average, maximum, minimum, and RMS wavefront slopes. The wavefront analysis options include determination of the effective focal length (aka wavefront radius of curvature) of the measured wavefront, the maximum and minimum, and RMS wavefront values after reconstruction.

The Zernike options control the decomposition and recomposition of the wavefront into Zernike polynomials. The Zernike results are in peak-to-valley amplitude. The ZernikeWFSubtract option allows the user to subtract Zernike terms from the wavefront. This subtraction only works when doing Zernike decomposition. This does not affect the slopes. The format for this string is comma separated index:value pairs. For example, if we wanted to subtract 1e-6 of Zernike 1 and 4, the string would be “1:1e-

6,4:1e-6”. A range of Zernike indices can be specified as well by separating them with the dash character. For example, to remove 1e-6 of the Zernikes 1 through 5, use the string “1-5:1e-6”. Use the letter “f” or the word “fit” to subtract the best fit from the wavefront. For example, to subtract the best fit of Zernikes 1 through 5 and one micron of Zernike 7, use, “1-5:fit,7:1e-6”.

The Alignment Window option uses calculations of first and second moment to draw an ellipse on a new window indicating the overall beam position and size. At the center of the ellipse is a line indicating the overall average beam tilt. If the Alignment Window is selected, the overall beam moments and tilt are displayed in the Analysis tab.

The Atmospheric Analysis option calculates the phase structure function from the reconstructed wavefront and extracts Fried’s coherence length and the atmospheric power law which are displayed in the Analysis tab.

Logging of analysis data may also be enabled on the Analysis Setup tab. Specify the filename in the LogFileName field and set the Logging field to true. Depending on the analysis settings, the results of the analysis will be logged in CSV (comma separated value) format to the specified file. This file can be opened in Matlab or Microsoft Excel for further analysis / plotting of the analysis results. The analysis log contains a header of the recorded values in the first line. The first data members are the index of the frame and the time stamp in milliseconds since 1/1/1970. It is very important to start a new file if any of the analysis options are changed because the column description header will not be saved out to an existing file.

3.1.5 Display Windows

There are four display windows in the AOS Hartmann Sensor software: the Wavefront, Intensity, Slopes Display, and Zernike Bar Chart windows. Each of these windows is shown in Figure 3.11.

3.1.5.1Wavefront Display Window

The Wavefront Display Window shows the user a false-color 2D plot or 3D rendering of the reconstructed wavefront. The window shows the minimum and maximum wavefront values and the peak to valley wavefront distortion. The user can select the colormap (see Figure 3.10and select to fix the scale of the display by entering values into the Minimum and Maximum text boxes. In the 3D rendering the user can adjust the view by clicking and dragging on the rendering.

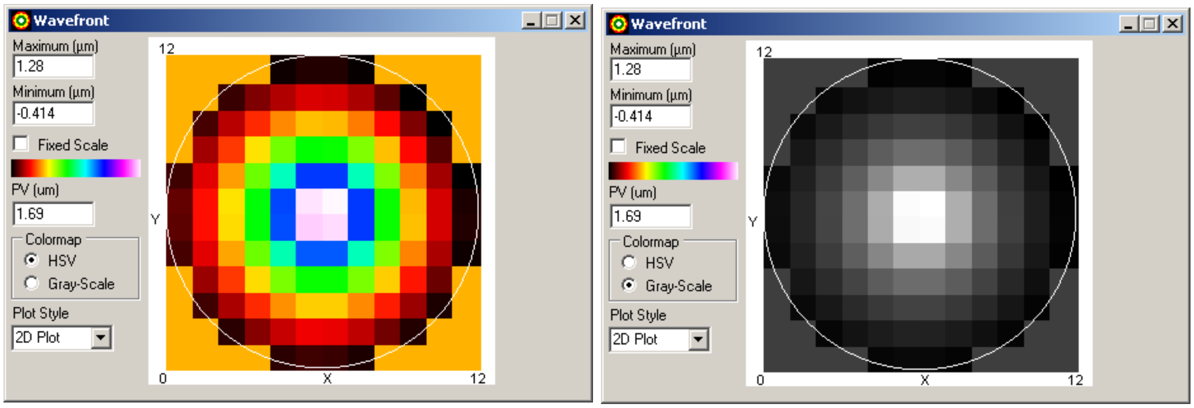


Figure 3.10 - Wavefront Display with the different colormaps

3.1.5.2Image Display Window

The Image Display Window shows the user the measured intensity, the AOIs as red boxes, and the centroid locations as green crosses. The maximum and minimum intensity values are shown as well. There is a progress bar in the bottom left that shows the relative intensity to help avoid saturation.

Three different intensity visualization modes are available in the Display Type group box. Image is the standard view with the measured intensity, AOIs and tracked centroids. Sub-Ap Sum display mode shows a normalized plot of the total measured intensity in each AOI. Finally, a histogram of the measured data may be displayed.

The user can use the right mouse button to delete individual AOIs with a single click or a set of AOIs in an ellipse or rectangle by clicking and dragging to draw the shape and then selecting the deletion operation from a pop-up context menu.

3.1.5.3Slopes Display Window

The Slopes Display window show the user a quiver-type plot of the measured wavefront slopes. The average and peak slope magnitudes are shown in the text boxes. The user can scale the length of the slope lines using the Scale Factor input box.

3.1.5.4Zernike Bar Chart Window

If the user selects Zernike decomposition of the wavefront, the Zernike Bar Chart Window shows a bar chart representing the measured decomposition. The scale can be fixed on the plot in the Analysis Setup tab. The numerical values of the decomposition can be obtained in the Analysis tab.

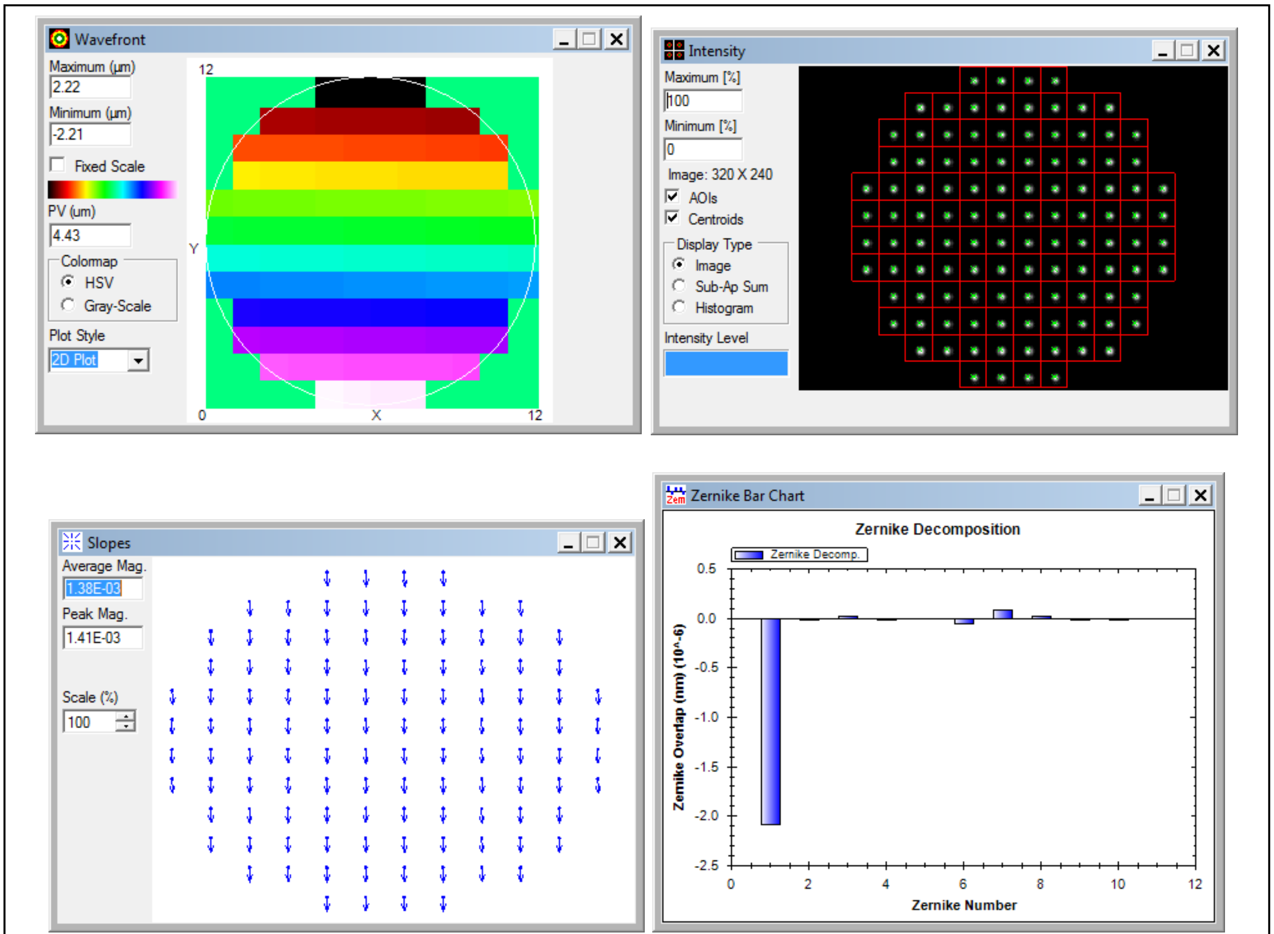
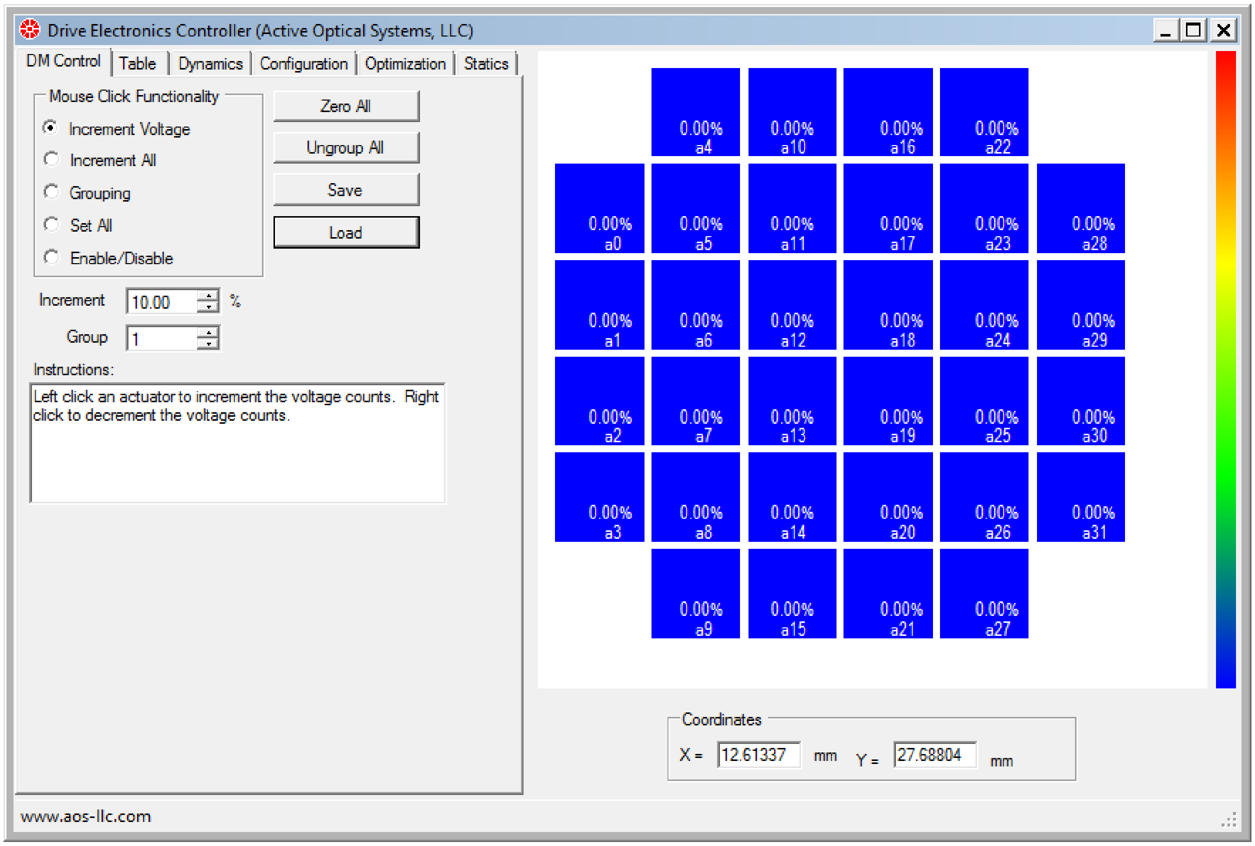


Figure 3.11 - The Display Windows: (from top left to bottom right) Wavefront (2D showing tilt), Image, Slopes, and Zernike Bar Chart

4 DM Controller Software

The software package that is provided for sending commands to the AOS Drive Electronics is called DM Controller. The main program window is shown here:



4.1.1 The Configuration Tab

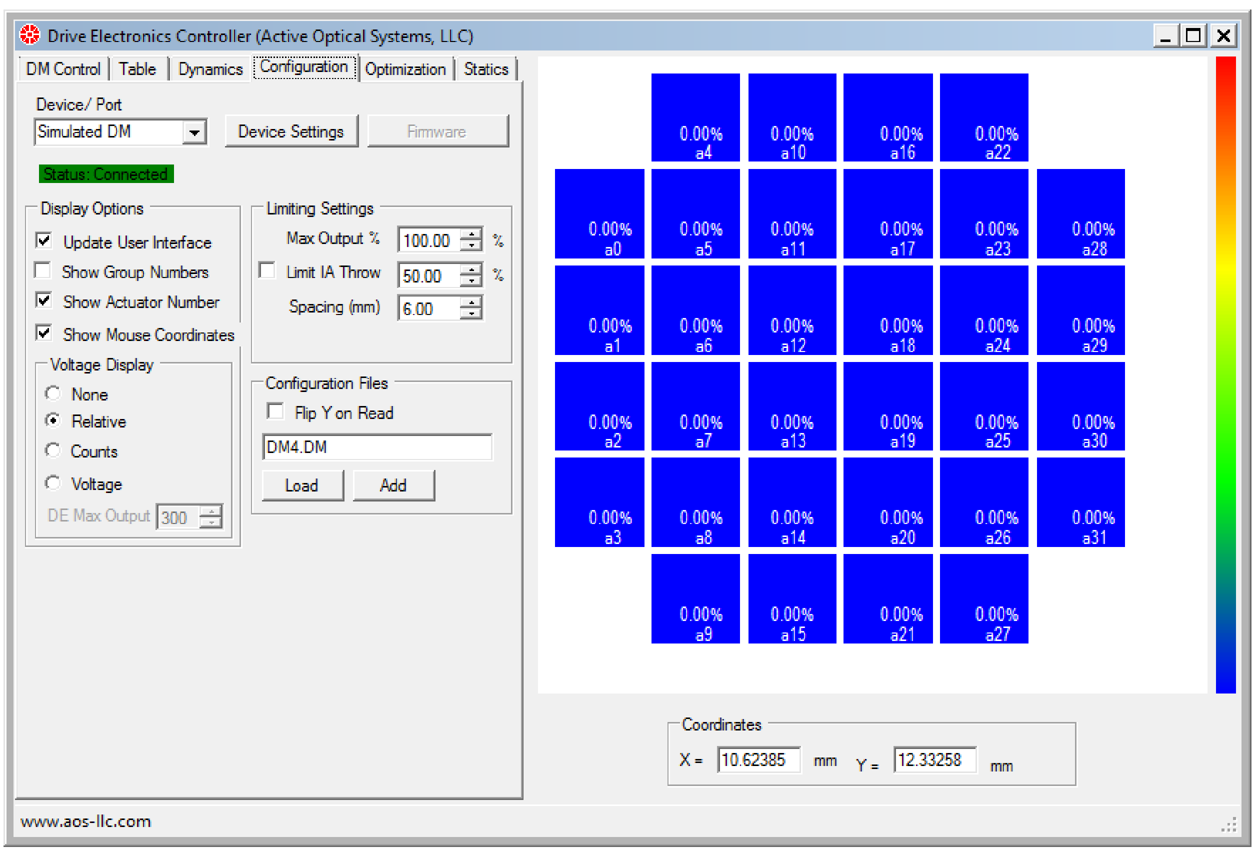
The configuration Tab is the central location for setting options associated with the DM Controller. The Device/Port drop down contains a list of all AOS drive electronics that are currently connected to the computer. Some devices have advanced settings associated with them that may be accessed by clicking on the Device Settings button. However, all of the required settings for most devices are shown directly on the Configuration tab.

In future version of the AOS Software, the Firmware button will allow users to update the firmware on their drive electronics with updates provided by AOS. However, this is not currently supported.

Various display options for the DM Controller may be set in the Display Options group box. The display options are summarized in Table 4.1.

	Option	Description
Display	Update User Interface	Enable/Disable update of actuator display (useful for increasing output command rate)
	Show Group Numbers	Show/Hide group numbers on actuator display
	Show Actuator Numbers	Show/Hide actuator numbers on actuator display
	Show Mouse Coordinates	Show/Hide mouse coordinates panel below actuator array
Voltage	None	Do not show voltage values on actuators
	Relative	Show voltage values as relative to full range output [0 – 100%]
	Counts	Show voltage values as quantized counts depending on precision of connected drive electronics (8,12,16 bit)
	Voltage	Show absolute voltage relative to DE Max Output defined below
	DE Max Output [V]	Max output voltage of drive electronics (used only for Voltage display mode)

Table 4.1 – DM Controller display options

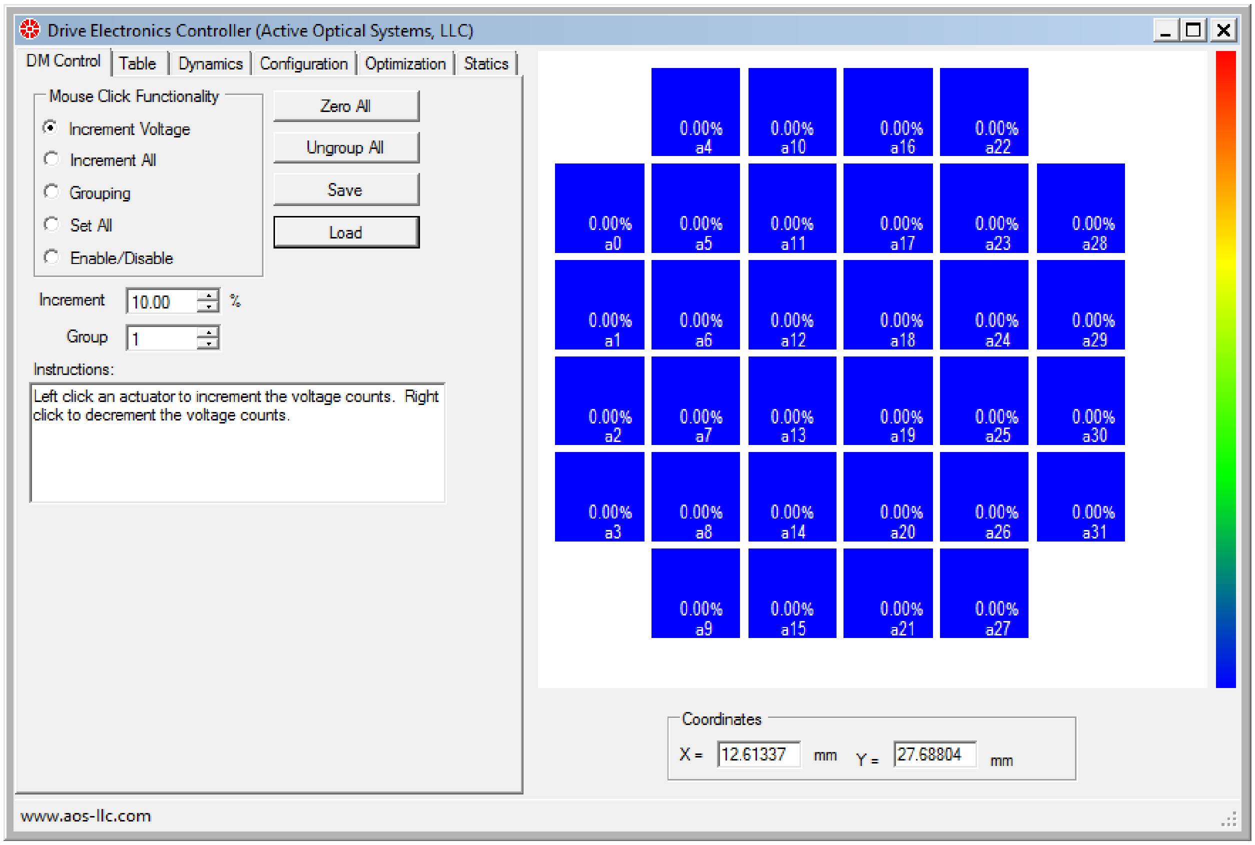


The DM configuration can be changed by loading a new *.DM file by pressing the Load button in the Configuration Files group box. DM configuration files contain information about the actuator geometry, channel values, physical output ports, grouping information, and disabled actuators. By clicking the “Add” button instead of the “Load” button, the voltages in the loaded DM config file will be added to the existing file.

The limiting settings may be adjusted in the Limiting Settings group box. The Max Output % value defines the absolute maximum value that an actuator may be commanded to, regardless of the values of neighboring actuators. By checking the Limit IA Throw check box, the values of actuators will be limited to not exceed their neighbors by the given value. Certain DM architectures are sensitive to inter-actuator throw. The spacing parameter is only required when inter-actuator throw limiting is enabled. It should be set to the actuator pitch of the mirror.

4.2 The DM Control Tab

There are several ways to control the DM by clicking on the actuator array interface. The modes of operation can be selected in the “Mouse Click Functionality” group box. The most common use is the “Increment Voltage” mode. In this mode the user can click on any of the actuator with the left mouse button to add an increment (specified by the “Increment” box) or right click to decrement the voltage. The “Increment All” mode works in a similar way except all the actuators are incremented or decremented when any one of them is clicked.



The “Grouping” mode allows the user to gang together any set of actuators so that an operation done on any one of them affects all of them in the same way. Grouping actuators also reduces the number of control points visible to an external control loop. This can be useful for grouping a “poorly sensed” actuator to its neighbor. To begin grouping a set of actuators together, start by select the “Grouping” mode from the “Mouse Click Functionality” group box. Then choose a group number for the actuators in the “Group” box. The 0 group is reserved for ungrouped actuators. When a group number is selected, left click on any of the actuators with the left mouse button to add them to the group. Right click on an actuator to remove it from all groupings. The “Ungroup All” button removes all actuators from any groups. As actuators are grouped, the border of each of the actuators changes color and, if selected in the Configuration tab, the group number appears near the center of the actuator.

The “Set All” mode is a way of setting all actuators to the same voltage as one of the other actuators. In this mode, simply select the actuator that you want all the others to mimic in voltage and all the voltages will change.

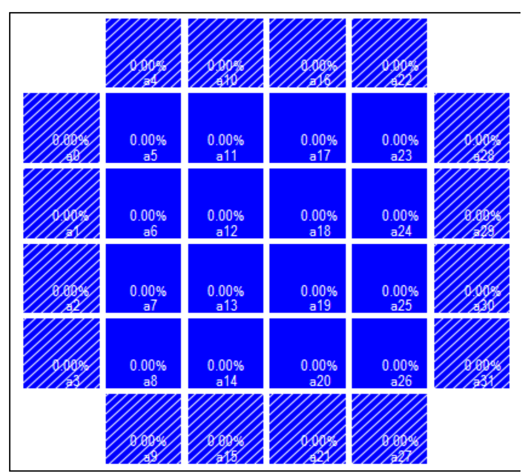


Figure 4.1 – Actuator array with disabled outer ring

The "Enable/Disable" mode allows the user to designate certain actuators as disabled. When in this mode, left clicking an actuator will enable it and right clicking an actuator will disable it. Disabled actuators are indicated by a cross-hatch pattern overlaid on the actuator (Figure 4.1). The behavior of a disabled actuator depends on the context of the command to the disabled actuator. This is necessary for the mirror to respond properly to bias commands and AO inputs. The behavior of different actuators depending on control context is summarized in Table 4.2.

Control Context	Disabled Actuator Behavior
Interactive Control from GUI	Always respond to inputs
Control from AO loop	Not visible to AO loop
Bias command from AO	Always respond to bias inputs from AO
Member of group	Always move with group members

Table 4.2 – Behavior of disabled actuators from different control contexts.

The Load and Save buttons provide the user with a way of loading and saving DM configuration files.

The colorbar used to visualize actuator values may be changed by right clicking on the colorbar. Several colorbars are available for selection. YellowBlue is the standard colorbar used in past versions of the DM Controller Software.

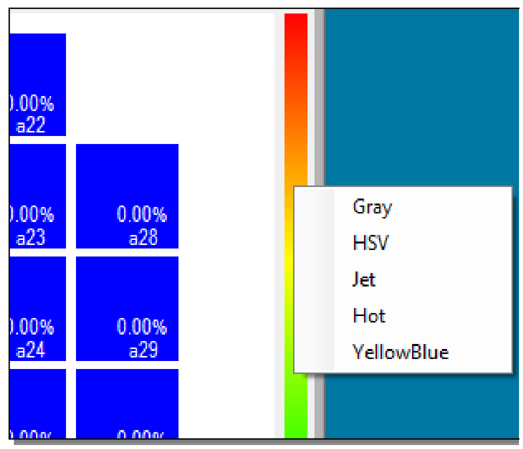
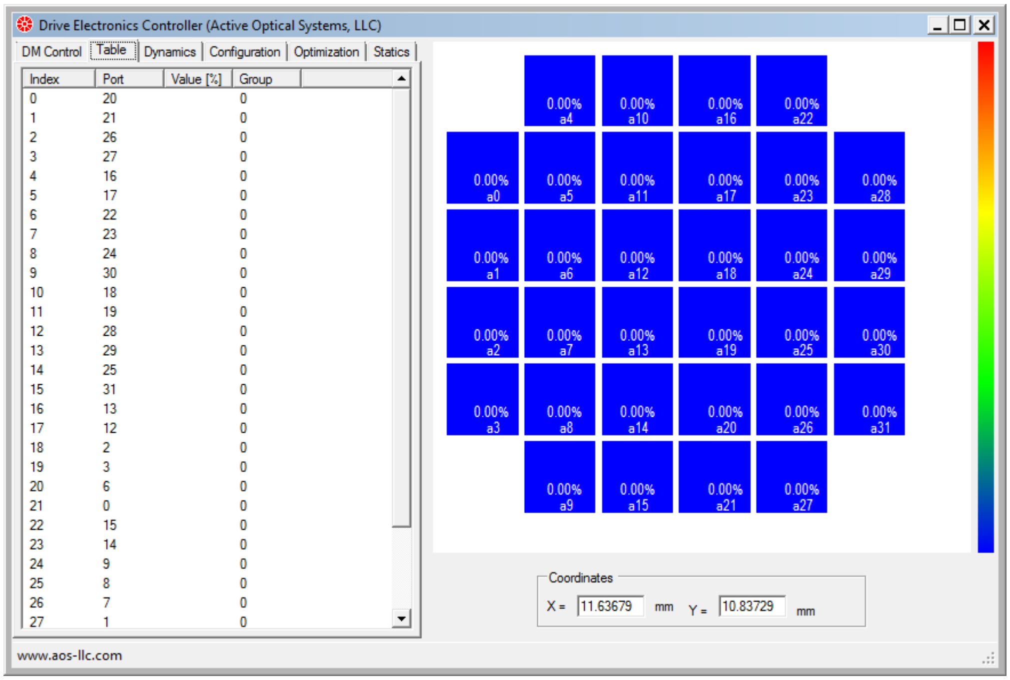


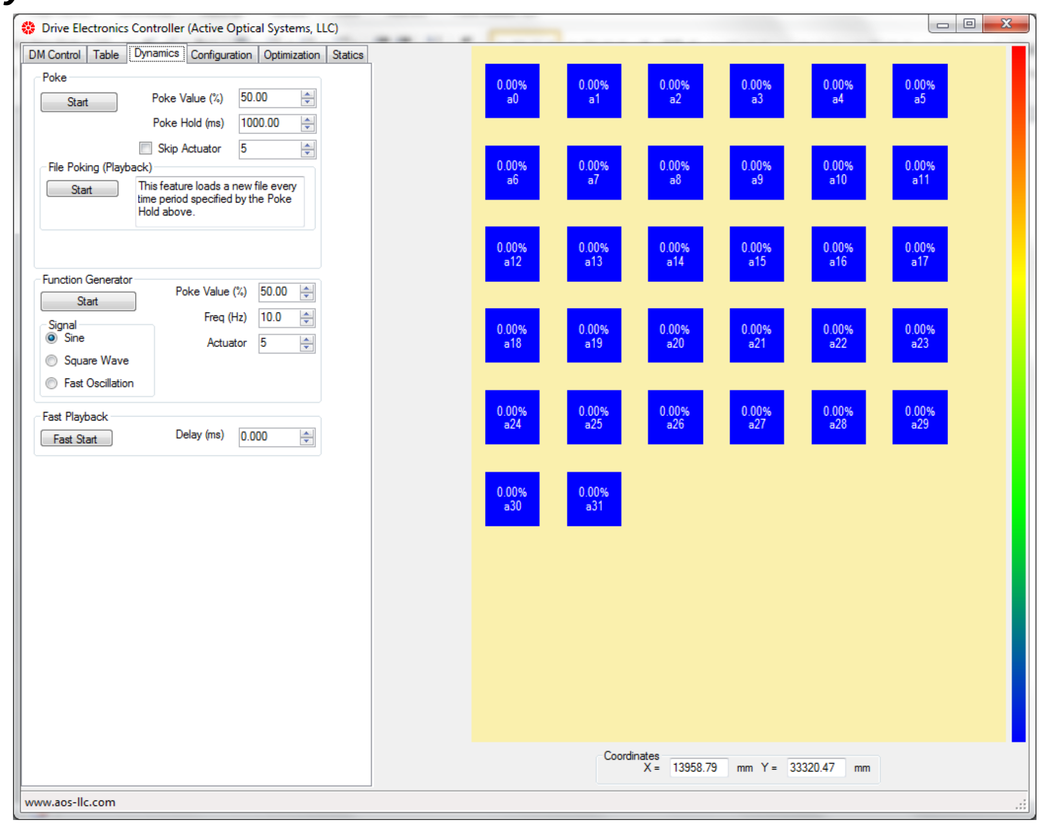
Figure 4.2 – Colorbar context menu

4.3 The Table Tab

The Table tab is simply a convenient way to see the state of all the actuators numerically. There is no way to do any form of data entry in this tab at this time. To save or load the state of the drive electronics, right click on the table and a pop-up context menu will appear with these options. Be careful when loading a DM file because the voltages will automatically be written to the DM upon load.



4.4 Dynamics Tab



In the Dynamics tab, the user can either poke each actuator or apply a function to a single actuator. In the poke mode, all the actuators are set to zero except one actuator

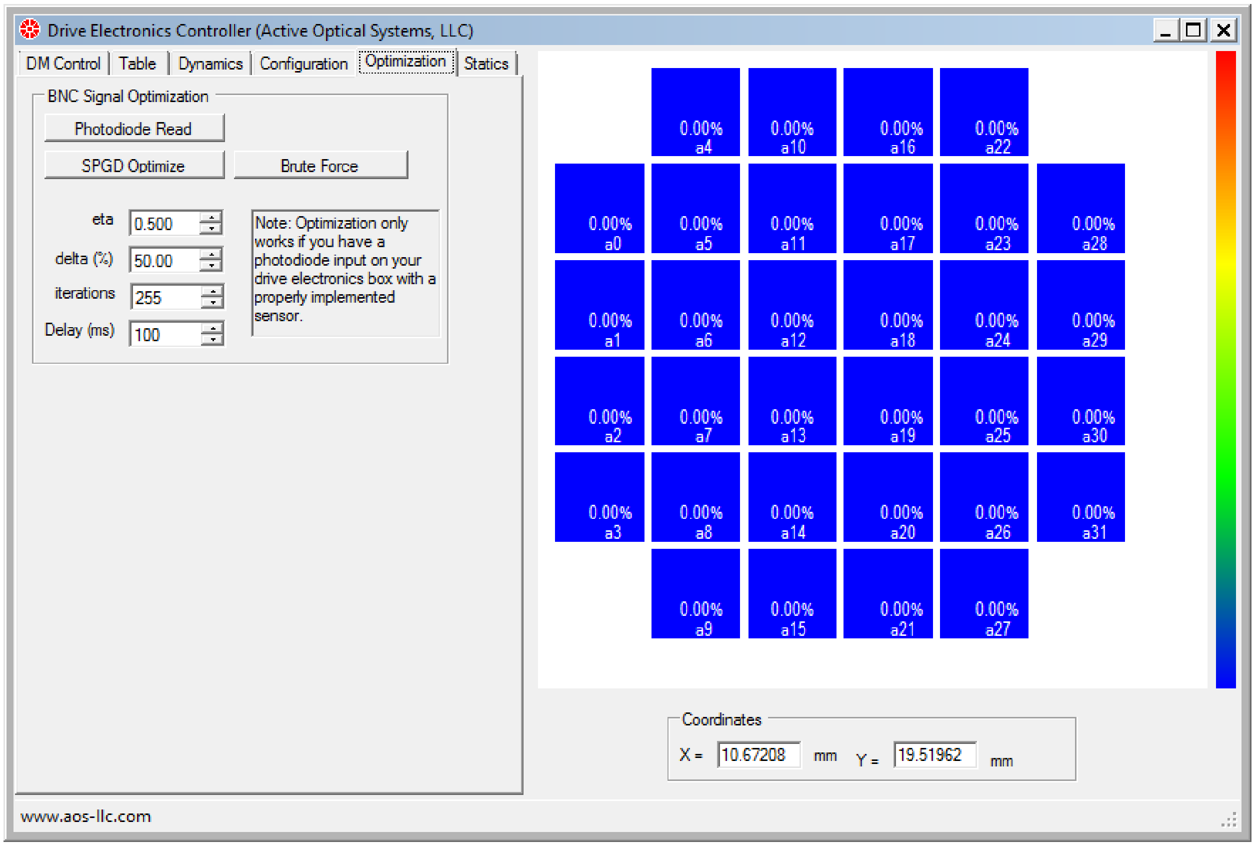
that is set to the poke value. The actuator remains poked for the number of milliseconds specified by the user in the “Poke Hold” box. Then the next actuator is poked. The user can specify to skip a single actuator (typically that corresponding to the mirror surface) by specifying that actuator in the “Skip Actuator” box and checking the check box adjacent to the label.
The “File Poking (Playback)” group box allows the user to select a series of DM files and apply them in a continuous loop. The user can specify the time delay between the application of the next file data in the Poke Hold box in milliseconds.

The “Function Generator” group box allows the user to apply a signal to a single actuator or group of actuators. In this mode, all actuators except one (or the group) are set to zero. The “Poke Value %” box allows the user to specify the peak-to-valley amplitude of this signal. The “Freq (Hz)” box allows the user to specify the desired output frequency of the signal. The maximum attainable frequency depends on the output rate of the drive electronics hardware being controlled. The actuator to apply the signal to is defined in the Actuator box.

The Signal group box allows the user to select a sinusoidal voltage signal or a square-wave signal. There is no control available for the offset and all voltages commanded will be in the range of 0 to “Poke Value %.” The Fast Oscillation mode creates a square wave with the highest possible frequency (Command High → Command Low → Command High → etc…). The frequency setting has no effect when Fast Oscillation mode is selected.

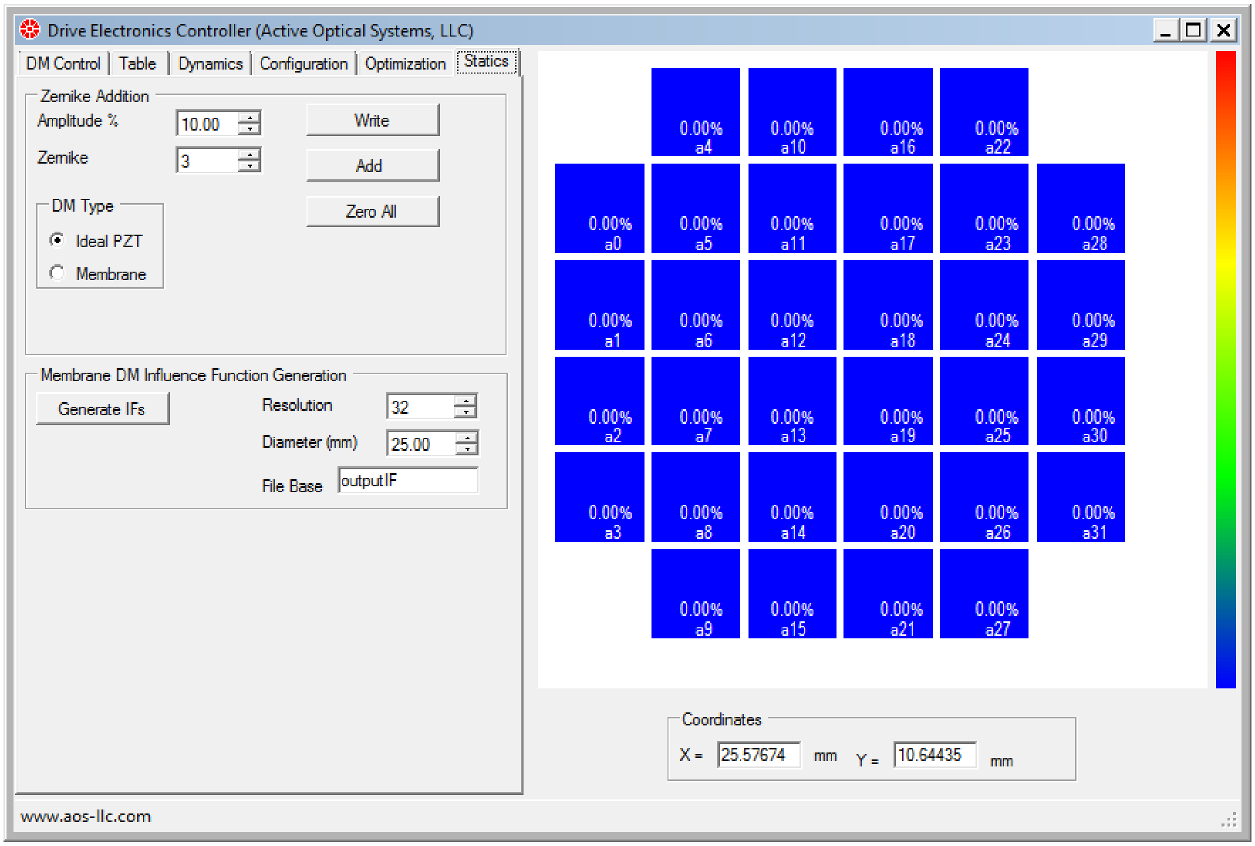
The fast playback group box functions much the same as the playback mode just at higher rates. This mode allows the user to load a CSV file containing DM voltage command vectors in each column, with each row corresponding to actuators. The software will then load all the commands. It will send them one at a time to check them for inter-actuator and general limiting while respecting the limiting settings specified by the user and report if there were any violations. It will then disable the GUI and playback the commands. The delay number up down box will allow the user to specify a delay between subsequent commands. The speed and jitter in this mode is dependent on the operating system, computer specs, and specified delay.

4.5 Optimization Tab



The Optimization tab is used to perform metric-based adaptive optics using drive electronics that is equipped with a photodiode input. The Photodiode Read button is used to test the value read from the photodiode. If the photodiode is attached and setup so that it is giving meaningful feedback to the system, the user can engage either brute force or SPGD optimization for a specified number of iterations. The Delay control is for specifying a settling time on the device. The eta and delta commands are the SPGD gain and step size respectively.

4.6 Statics Tab



The Statics tab is used to either write out Zernikes open-loop or generate membrane DM influence functions. The Membrane DM Influence Function Generation group box is used to generate comma-separated value (CSV) files containing 2D mappings of the DM surface during the actuation of a single actuator. The File Base text box is used to specify a prefix for the file names. The Resolution box specifies the one-axis resolution of the output influence function. The DM diameter is specified in the Diameter input. For high resolution outputs (128+), this can be very time consuming.

The Zernike Addition group box allows the user to either write out a static Zernike term or add a Zernike Term to the existing commands. The user can specify the amplitude in counts and the Zernike term number. The Zernikes are numbered using (n, l) ordering such that the first term (0) is the tilt term in x. The following terms are y tilt, 90-degree astigmatism, focus, 45-degree astigmatism, x trefoil, x coma, y coma, y trefoil, etc. For a PZT-actuator DM, this is very fast because it assumes no crosstalk between actuators, but for a membrane DM, the program actually generates the phase-space poke matrix and inverts it to determine an effective control matrix. This generation can be very time consuming, but it only needs to happen once if the user does not change the configuration file.

5 Adaptive Optics Form

When the user selects the Start Adaptive Optics menu option from the Windows menu in the Hartmann sensor software (Figure 5.1), the Adaptive Optics (AO) form will launch with the DM Controller. The figure below shows all the tabs of the Adaptive Optics tab.

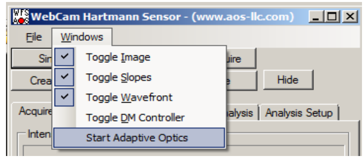


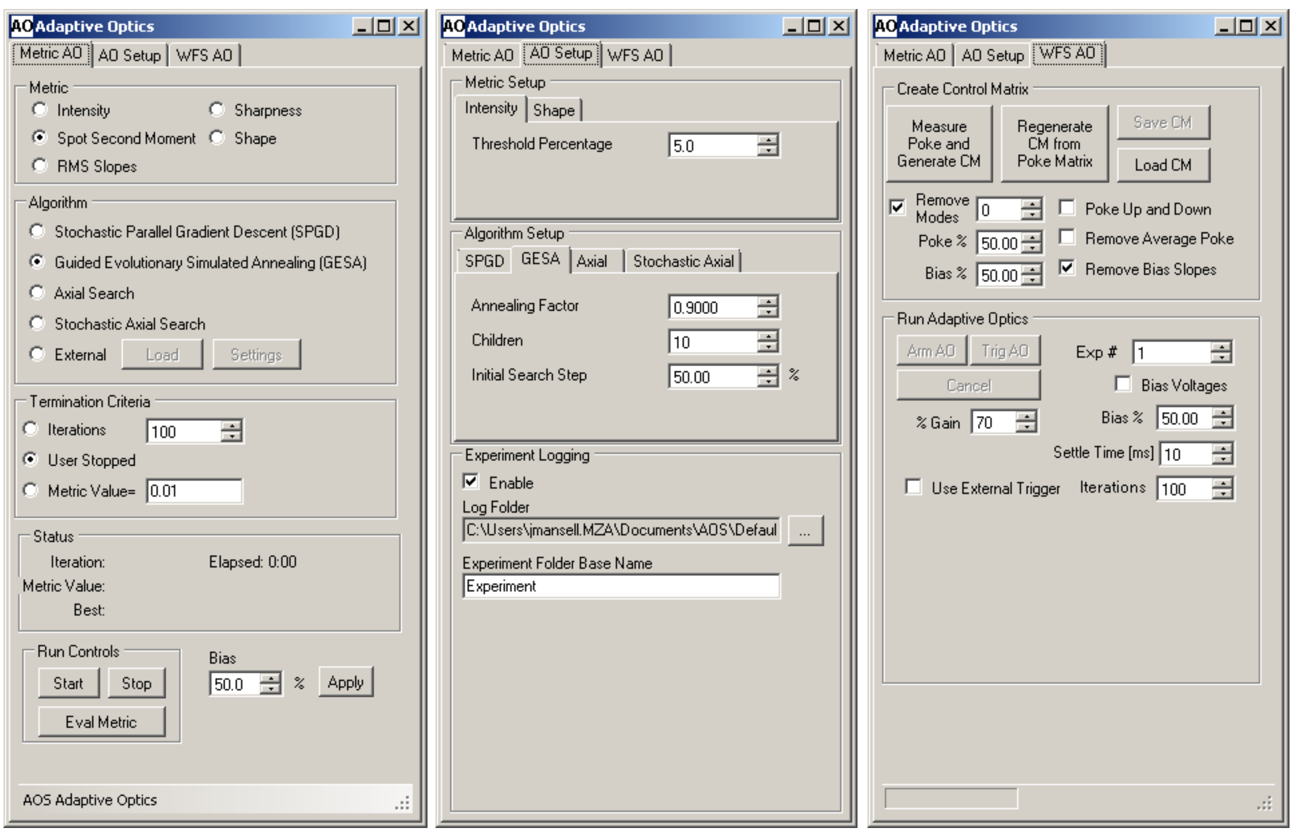
Figure 5.1 - Menu option for starting the adaptive optics form.

5.1 Metric AO Tab

5.1.1 Metric Selection

The first tab is for setting-up and running metric adaptive optics. The first group-box is for establishing the metric. Most of these metrics are designed to be used with an intensity sensor instead of a wavefront sensor. When Intensity is selected, the algorithm maximizes the intensity on the camera.

When Spot Second Moment is selected, the AO system will minimize the second moment of the intensity profile on the camera. When Sharpness is selected, the square of the intensity gradient is maximized. The RMS Slopes option is for use with a Hartmann sensor. When selected, the AO system will minimize the RMS Wavefront Slopes (a.k.a. wavefront gradient). The Shape metric allows the user to define an



Adaptive Optics Form

intensity profile as a difference of two super-Gaussians. We will show the user interface for defining an intensity profile shape in the next section.

5.1.2 Algorithm Selection

There are several different algorithms implemented for performing metric AO. First is stochastic parallel gradient descent (SPGD). In this algorithm, a random direction is established in multi-dimensional error space where each actuator or group of actuator represents a dimension of this space. A single step is taken a user-specified amount and then a larger scaled jump is taken based on the results of that step. The gain associated with the jump (alpha) and the search step is specified by the user in the Metric AO Setup tab.

The guided evolutionary simulated annealing (GESA) algorithm is a genetic-type search algorithm. A family of solutions (children) is generated in a random radius from a starting point, which is referred to as the family's parent or base. The best solution from a generation of solutions becomes the next generation's parent, which is the center of the next generation's search. The radius of each subsequent generation's search is reduced by a simulated annealing factor. The number of children per generation, the initial search radius, and the simulated annealing factor are all user-specified in the Metric AO Setup tab.

The axial search chooses each actuator or group of actuators as a dimension or axis and does a bifurcating search in that axis to determine the best position of each control point. This search algorithm assumes that there is no benefit in searching in multiple axes in parallel, which is often not the case, but is very good at fine-tuning a given solution. The user can specify the initial search step and the minimum step size required before moving to a new axis.

The stochastic axial search is a modification of the SPGD algorithm. A random direction is generated and then a bifurcating search is performed in that direction. The user can specify the initial search step and the minimum step size required before moving to a new axis.

The external algorithm selection will be available in future versions of the AOS software and will allow a user to define an external DLL for their own optimization algorithm.

5.1.3 Termination Criteria & Status

There are three possible termination criteria for the adaptive optics control loop: a user-specified number of iterations, a user-specified metric value, or a user determined stop. The Termination Criteria tab allows the user to select any of these for loop termination. During a loop, the Status box shows the user the iteration number, the elapsed time, the current metric value, and the best metric value obtained so far. The buttons at the bottom of the form allow the user to start and stop the metric AO loop and evaluate the metric for a given out-of-loop condition.

5.2 Metric AO Setup

The Metric Setup box allows the user to specify an intensity threshold for the Second Moment metric or specify a shape by clicking the Define Shape button in the Shape tab.

5.2.1 Intensity Profile Shape Metric Definition

When the user selects the Define Shape button, they can use the parameters of the two super-Gaussian profiles to create a desired intensity profile. The profile can be previewed in the preview window. The current intensity profile on the camera is also displayed for convenience.

5.2.2 Algorithm Setup and Experiment Logging

The Algorithm Setup box allows the user to specify parameters of the search algorithms described above. The Experiment Logging box allows the user to specify a logging file and location and to turn logging on and off.

5.3 WFS AO

The final tab in the AO form is the wavefront sensor (WFS) adaptive optics (AO) tab for setting up and performing traditional control matrix adaptive optics.

5.3.1 Create Control Matrix

This box enables the user to create or load a control matrix(CM). The Generate CM button starts a poke matrix generation by applying voltage to each actuator or group of actuators and measuring the wavefront sensor response. The user can specify the bias and poke value in counts. The Poke Up and Down checkbox allows the user to poke in both directions about a bias condition and use the difference for the measured influence function in order to eliminate any static aberrations in the system.

Once the poke matrix is generated, the control matrix is generated by doing a pseudo-least-squares inverse using single value decomposition (SVD). The user can remove a variable number of modes from the SVD inverse of the poke matrix.

Once the CM is generated it and the poke matrix can be saved as comma separated value (CSV) files using the Save CM button. If the user generated their own control matrix, it can be loaded using the Load CM button. It is essential to get the size of this matrix right before loading it.

The generation of the control matrix triggers the display of a window with tabs showing the control matrix, poke matrix, wavefront SVD modes, DM SVD modes, the SVD gains, and the poke matrix slope vector amplitudes plotted with the measured RMS waverfront sensor slope noise.

5.3.2 Run Adaptive Optics

Once a valid CM is generated or loaded, the user can perform adaptive optics. The AO loop multiplies the vector of slopes by the control matrix to get a vector of delta commands. The DM commands are then calculated as the sum of the existing commands and a user-specified gain factor times the delta commands. The user can specify the bias condition for the DM in counts or chose to use the existing DM shape to start the AO loop (Bias Voltages checkbox).

To setup the system for AO, the user needs to press the Arm AO button. Once pressed, the user can trigger the AO by pressing the Trig AO button. The AO loop will run with a user-specified number of iterations. With some sets of the AOS electronics, the user can trigger the AO or turn the AO gain on and off externally using a TTL input to the drive electronics and checking the Use External Trigger and Exo DM checkboxes respectively.

The Settle Time box allows the user to specify a wait time between commanding the DM and reading the WFS in order to ensure that the DM has reached its command position before reading the WFS again. Experiments are automatically logged in the program directory. The experiment number can be user specified on the form.

6 Hardware Keys

As of 12/19/2012, Active Optical Systems is providing hardware keys with its hardware to enable usage of the hardware and software outside of its software. The EULA now contains the text:

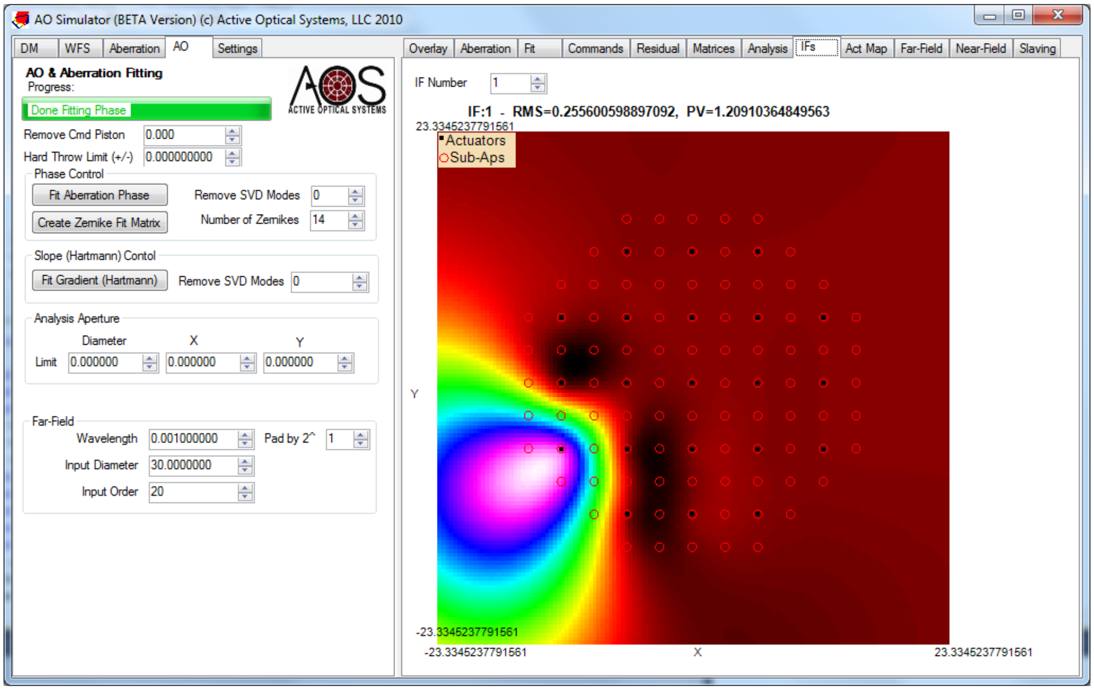
> The user agrees not to modify or attempt to modify or reprogram the hardware key except as instructed by Active Optical Systems, LLC. Any attempts to modify the key will result in termination of the software license and termination of support. Any and all damage to the key is the responsibility of the user. Damaged keys may be replaced after return of the original key for a reasonable cost. Lost keys will require the user to purchase a new software license.

7 Other Software

As of version 2.0.3, AOS software customers will begin seeing new software tools for modeling and simulating AO systems and propagation. These tools are being released in BETA format for free with the AOS software. AOS encourages any and all feedback on the new software.

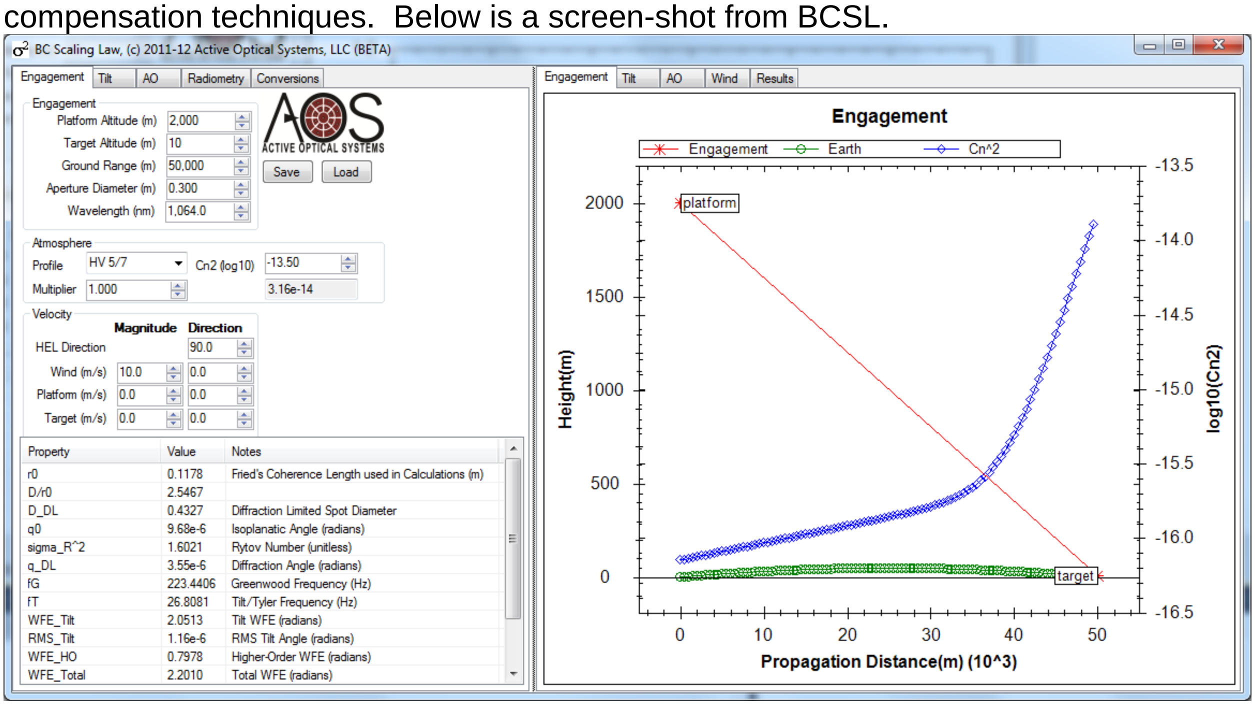
7.1 AO Simulator

The AO Simulator allows users to analyze compensation of wavefronts against a variety of different kinds of deformable mirrors and with various wavefront sensors. The output is compatible with WaveTrain, a full wave-optics simulation package available from our affiliate company, MZA Associates Corporation (www.mza.com). Below is a screen shot from the AO Simulator program.



7.2 *Beam Control Scaling Law*

The Beam Control Scaling Law (BCSL) program allows users to model simple engagements and propagations through the atmosphere and analyze different compensation techniques. Below is a screen-shot from BCSL.



8 Appendix A: File Formats

8.1 WFS File Format

The AOS HWFS code stores the data as text to facilitate easy file reading. Here is the format

Line 1:
Line Format: “Version: [Version Number]”
[Version Number] is currently 1.2

Line 2:
A comma delimited list of:
1. Hartmann Array to Camera Separation (m)
2. Absolute Threshold (counts)
3. X Pixel Size (m)
4. Y Pixel Size (m)]
5. Unused Parameter

Line 3-end:
A comma delimited list of:
1. Sub-aperture minimum pixel in X (pixels)
2. Sub-aperture minimum pixel in Y (pixels)
3. Sub-aperture maximum pixel in X (pixels)
4. Sub-aperture maximum pixel in Y (pixels)
5. Reference Centroid X Location (pixels)
6. Reference Centroid Y Location (pixels)
7. X Sub-Aperture Index on a 2D grid (unitless)
8. Y Sub-Aperture Index on a 2D grid (unitless)
9. Measured Centroid X Location (pixels)
10. Measured Centroid Y Location (pixels)
11. Slope in X (radians)
12. Slope in Y (radians)
13. Relative Intensity (sum in counts above threshold)
14. Z position (m)

8.2 DM File Format

The DM file format provides the DMController with information about the deformable mirror and how to adjust the user interface. The first character if each line specifies the type of information that is on that line. The type specifers are as follows:

A = Actuator Definition
V = Voltage Information

G = Grouping Information

C = Communication Specification

Each line is comma delimited and terminated with a newline character.

“A” = Actuator Definitions

Following the A in this order is:

1. the number of points in the shape definition,
2. the channel number of the electronics that controls this actuator, and
3. a series of x and y coordinates for the shape representing that actuator.

The index number in the AOS DM Controller software is given by the order in which these actuators are read into the program.

“V” = Voltage Information

Following the V is a comma delimited list of the voltage on each of the actuators in counts on the DAC.

“G”=Grouping Information

Following the G is a comma delimited list of each actuator’s group.

“C” = Communication Specification

Communication with the AOS Drive Electronics is done using a virtual COM port. The specifications for this communication is stored in the “C” section. The first line specifies the baud rate. Subsequent lines specify the parity, data-bits, and stop bits. Following lines are no longer used.

Example DM File

Here is an example DM file for a 31 actuator hex-grid DM:

```
A,7,1,3.034615,1.559956,2.965385,1.559956,2.930769,1.500000,2.965385,1.440044,3.034615,1.440044,3.069231,1.500000,3.034
615,1.559956,
A,7,2,3.034615,1.688527,2.965385,1.688527,2.930769,1.628571,2.965385,1.568616,3.034615,1.568616,3.069231,1.628571,3.034
615,1.688527,
A,7,3,3.145961,1.624241,3.076731,1.624241,3.042115,1.564286,3.076731,1.504330,3.145961,1.504330,3.180577,1.564286,3.145
961,1.624241,
A,7,4,3.145961,1.495670,3.076731,1.495670,3.042115,1.435714,3.076731,1.375759,3.145961,1.375759,3.180577,1.435714,3.145
961,1.495670,
A,7,5,3.034615,1.431384,2.965385,1.431384,2.930769,1.371429,2.965385,1.311473,3.034615,1.311473,3.069231,1.371429,3.034
615,1.431384,
A,7,6,2.923269,1.495670,2.854039,1.495670,2.819423,1.435714,2.854039,1.375759,2.923269,1.375759,2.957885,1.435714,2.923
269,1.495670,
A,7,7,2.923269,1.624241,2.854039,1.624241,2.819423,1.564286,2.854039,1.504330,2.923269,1.504330,2.957885,1.564286,2.923
269,1.624241,
A,7,8,3.034615,1.817098,2.965385,1.817098,2.930769,1.757143,2.965385,1.697187,3.034615,1.697187,3.069231,1.757143,3.034
615,1.817098,
A,7,9,3.145961,1.752813,3.076731,1.752813,3.042115,1.692857,3.076731,1.632902,3.145961,1.632902,3.180577,1.692857,3.145
961,1.752813,
A,7,10,3.257308,1.688527,3.188077,1.688527,3.153461,1.628571,3.188077,1.568616,3.257308,1.568616,3.291923,1.628571,3.25
7308,1.688527,
A,7,11,3.257308,1.559956,3.188077,1.559956,3.153461,1.500000,3.188077,1.440044,3.257308,1.440044,3.291923,1.500000,3.25
7308,1.559956,
A,7,12,3.257308,1.431384,3.188077,1.431384,3.153461,1.371429,3.188077,1.311473,3.257308,1.311473,3.291923,1.371429,3.25
7308,1.431384,
A,7,13,3.145961,1.367098,3.076731,1.367098,3.042115,1.307143,3.076731,1.247187,3.145961,1.247187,3.180577,1.307143,3.14
5961,1.367098,
A,7,14,3.034615,1.302813,2.965385,1.302813,2.930769,1.242857,2.965385,1.182902,3.034615,1.182902,3.069231,1.242857,3.03
4615,1.302813,
```

A,7,15,2.923269,1.367098,2.854039,1.367098,2.819423,1.307143,2.854039,1.247187,2.923269,1.247187,2.957885,1.307143,2.923269,1.367098,
A,7,16,2.811923,1.431384,2.742692,1.431384,2.708077,1.371429,2.742692,1.311473,2.811923,1.311473,2.846539,1.371429,2.811923,1.431384,
A,7,17,2.811923,1.559956,2.742692,1.559956,2.708077,1.500000,2.742692,1.440044,2.811923,1.440044,2.846539,1.500000,2.811923,1.559956,
A,7,18,2.811923,1.688527,2.742692,1.688527,2.708077,1.628571,2.742692,1.568616,2.811923,1.568616,2.846539,1.628571,2.811923,1.688527,
A,7,19,2.923269,1.752813,2.854039,1.752813,2.819423,1.692857,2.854039,1.632902,2.923269,1.632902,2.957885,1.692857,2.923269,1.752813,
A,7,20,3.145961,1.881384,3.076731,1.881384,3.042115,1.821429,3.076731,1.761473,3.145961,1.761473,3.180577,1.821429,3.145961,1.881384,
A,7,21,3.257308,1.817098,3.188077,1.817098,3.153461,1.757143,3.188077,1.697187,3.257308,1.697187,3.291923,1.757143,3.257308,1.817098,
A,7,22,3.368654,1.624241,3.299423,1.624241,3.264808,1.564286,3.299423,1.504330,3.368654,1.504330,3.403269,1.564286,3.368654,1.624241,
A,7,23,3.368654,1.495670,3.299423,1.495670,3.264808,1.435714,3.299423,1.375759,3.368654,1.375759,3.403269,1.435714,3.368654,1.495670,
A,7,24,3.257308,1.302813,3.188077,1.302813,3.153461,1.242857,3.188077,1.182902,3.257308,1.182902,3.291923,1.242857,3.257308,1.302813,
A,7,25,3.145961,1.238527,3.076731,1.238527,3.042115,1.178571,3.076731,1.118616,3.145961,1.118616,3.180577,1.178571,3.145961,1.238527,
A,7,26,2.923269,1.238527,2.854039,1.238527,2.819423,1.178571,2.854039,1.118616,2.923269,1.118616,2.957885,1.178571,2.923269,1.238527,
A,7,27,2.811923,1.302813,2.742692,1.302813,2.708077,1.242857,2.742692,1.182902,2.811923,1.182902,2.846539,1.242857,2.811923,1.302813,
A,7,28,2.700577,1.495670,2.631346,1.495670,2.596731,1.435714,2.631346,1.375759,2.700577,1.375759,2.735192,1.435714,2.700577,1.495670,
A,7,29,2.700577,1.624241,2.631346,1.624241,2.596731,1.564286,2.631346,1.504330,2.700577,1.504330,2.735192,1.564286,2.700577,1.624241,
A,7,30,2.811923,1.817098,2.742692,1.817098,2.708077,1.757143,2.742692,1.697187,2.811923,1.697187,2.846539,1.757143,2.811923,1.817098,
A,7,31,2.923269,1.881384,2.854039,1.881384,2.819423,1.821429,2.854039,1.761473,2.923269,1.761473,2.957885,1.821429,2.923269,1.881384,
V,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
G,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
C,9600,
C,None,
C,8,
C,1,
C,1,
C,255,

8.3 Analysis Log File Format

The analysis log file format is based on a comma separated value (CSV) file format. The first line is the description of each of the columns. Each subsequent line is the data from an individual acquisition. The first column is an index number associated with the frame number in a sequence. The second column is the time stamp in milliseconds from 1/1/1970.

8.4 DAT File Format

The DAT file format was invented at AOS to be a very lightweight easy format for recording a series of images. The latest version of this file (3.0) has the following header:

Data Element	**Offset**	**Bytes**	**Data Type**
Version	0	8	Double
Bit Depth (8 or 16)	8	4	Int32

Each frame has the following header:

Data Element	Offset (After Header)	Bytes	Data Type
FrameID	0	8	Uint64
Time	8	8	Double
Width	16	4	Uint32
Height	20	4	Uint32
Buffer Size	24	4	Uint32

After the frame header each pixel irradiance is then recorded as a 8 or 16 bit unsigned integer until the last frame has been recorded.

Below is example MATLAB code to read a DAT file:

```
fp = fopen(fname, 'rb');
if (fp<=0)
   disp('Could not open file');
   return;
end;

version = fread(fp, 1, 'double', 'ieee-le');
v=2.0;
if (version~=2.0 && version~=3.0)
   v=1.0;
   fseek(fp,0,'bof');
else
   v=version;
end;

bitdepth=8;
if (v>=3)
   bitdepth = fread(fp, 1, 'int32', 'ieee-le');
end;

cnt = 1;
while ~feof(fp)
   position = ftell(fp);
   frameID = fread(fp, 1, 'uint64', 'ieee-le');
   if(isempty(frameID))% || frameID == 0)
      break;
   end
   if (v>1.5)
      timeIn(cnt) = fread(fp,1,'double','ieee-le');
   end;
   width = fread(fp, 1, 'uint32', 'ieee-le');
   height = fread(fp, 1, 'uint32', 'ieee-le');
   buf_size = fread(fp, 1, 'uint32', 'ieee-le');
   bytesPerPixel = buf_size / (width*height);
```

```
    if (bytesPerPixel==1)
        data = fread(fp, buf_size, 'uint8', 'ieee-le');
    else
        data = fread(fp, width*height, 'uint16', 'ieee-le');
    end;

    if (length(index)==1 && index~=1)
        headersize = 20;
        if (v>1.5) headersize = headersize + 8; end;

        startOffset = 0;
        if (v > 1.5) startOffset = 8; end; % static offset of 8 for version number double
        if (v > 2.5) startOffset = 12; end;  % additional offset of 4 for bit depth
        fileOffset = ((buf_size+headersize)*index) + startOffset;
        fseek(fp,fileOffset,'bof');
%       fseek(fp,(headersize+buf_size)*(index-1),'cof');

        frameID = fread(fp, 1, 'uint64', 'ieee-le');
        if(isempty(frameID))% || frameID == 0)
            break;
        end
        if (v>1.5)
            timeIn(cnt) = fread(fp,1,'double','ieee-le');
        end;
        width = fread(fp, 1, 'uint32', 'ieee-le');
        height = fread(fp, 1, 'uint32', 'ieee-le');
        buf_size = fread(fp, 1, 'uint32', 'ieee-le');
        data = fread(fp, buf_size, 'uint8', 'ieee-le');
    end;

    frame_ids(cnt+1) = frameID;

    if (width*height~=length(data))
        warning('Data lengths do not match');
        return;
    end
    imgTotal = reshape(data,width,height)';

    img{cnt}=imgTotal;

    cnt = cnt + 1;

    if (length(index)==1)
        img = img{1};
        break;
    end;

    if (cnt>max(index))
        break;
    end;
end
```

```
fclose(fp);
if (length(index)==1)
   return;
end;
if (length(index)>0 && v>1.5)
   img = img{index};
   timeIn = timeIn(index);
end;
return;
```

9 Appendix B: External Software Interfaces

We are currently developing interfaces to our products with an interface so that users can use programs like Matlab to communicate with and command our hardware. Please contact AOS for more details on this interface if you are interested in becoming a beta-tester.

9.1 Matlab Interface

9.1.1 Scripts & Functions (Original)

We are now providing some support functions in Matlab for reading, writing, and displaying AOS data files. Matlab version 2009a provides users with the ability to call .NET assemblies directly from inside Matlab, so we have some scripts now that also allow users to interface with the AOS hardware directly in Matlab. The scripts each have internal documentation on their usage that can be accessed with the help or doc command in Matlab, but their function is summarized in the table below.

Function	Description
DM Files	
LoadDM.m	Load a .DM file.
SaveDM.m	Save a .DM file.
ShowDM.m	Display a .DM file.
WFS Files	
LoadWFS.m	Load a .WFS file.
SaveWFS.m	Save a .WFS file.
ShowWFS.m	Display a .WFS file.
Example External Usage	
RunHWFS.m	Loads a reference file and does a user-terminated loop of acquisition and display of the slopes.
TestRunHWFS.m	Example script for RunHWFS function.
MakePokeAndControl.m	Pokes on each of the actuators and makes a poke and control matrix.
TestMakePokeAndControl.m	Example script for MakePokeAndControl function.
RunAO.m	Loads a reference file and uses a user-created control matrix in a user-terminated loop of adaptive optics.
TestRunAO.m	Example script for RunAO function.
Support Functions	
Centroid.m	Performs a centroid on a sub-aperture image.
ProcessHWFSSlopes.m	Calculates the slopes for a HWFS.

9.1.2 Class Library

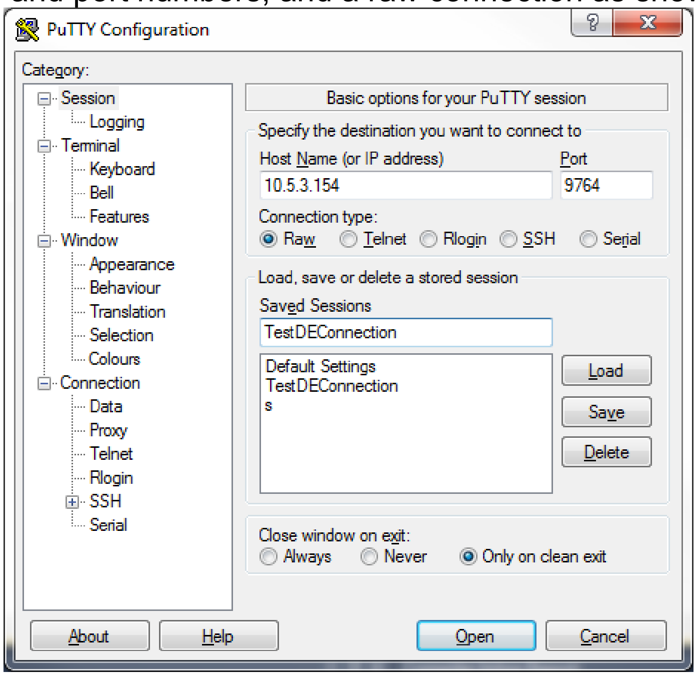
The AOS Matlab functions have now been wrapped into Matlab classes. The classes themselves are documented internally. This documentation can be accessed using the Matlab "doc" command with the name of the class. Below is a list of the key Matlab .m files that come with the AOS Software. NOTE: It is a violation of the license agreement to use these scripts to process anything except data taken using AOS hardware.

Here are some key files that comprise the class library:
Matlab\Files\DM\FileInterfaceDM.m
Matlab\Files\WFS\FileInterfaceWFS.m
Matlab\Hardware\DM\AOSDM.m
Matlab\Hardware\SLM\SLM.m
Matlab\Hardware\Shutter\Shutter.m
Matlab\Hardware\Shutter\TestShutter.m
Matlab\Hardware\Camera\1394\Cam1394.m
Matlab\Hardware\Camera\NICameraLink\CamLinkCam.m
Matlab\Processing\CWFS\CWFSProcessor.m
Matlab\Processing\HWFS\HWFSProcessing.m
Matlab\Processing\Kolmogorov\kolmogorovTurbulenceScreen.m
Matlab\Processing\StrehlCamera\StrehlCameraProcessing.m
Matlab\Systems\HWFS_AO\HWFS_AO.m

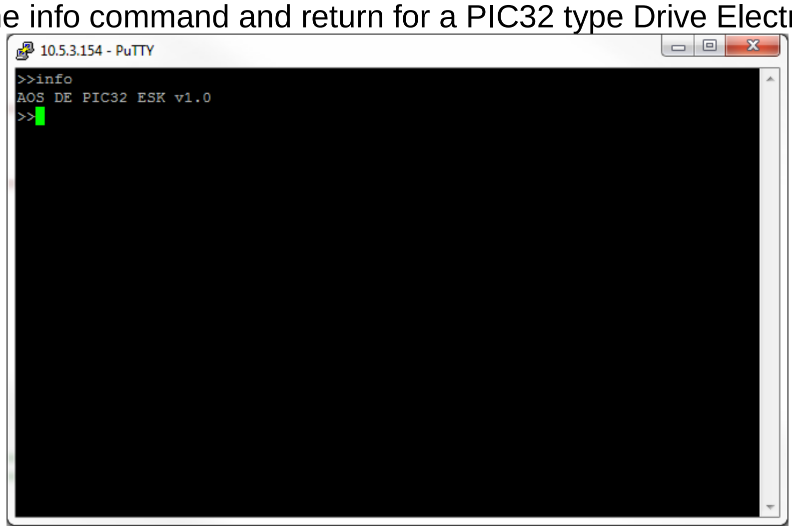
Here are some example scripts for the class library:
Matlab\Hardware\DM\TestAOSDM.m
Matlab\Hardware\SLM\SLMAlign.m
Matlab\Hardware\SLM\SLMCalibrationProcess.m
Matlab\Hardware\SLM\SLM_HighSpeedVersion.m
Matlab\Hardware\SLM\TestSLM_HighSpeedVersion.m
Matlab\Hardware\SLM\WriteConcentricCircleOnSLM.m
Matlab\Hardware\SLM\WriteSLM.m
Matlab\Hardware\SLM\WriteSpiralToSLM.m
Matlab\Hardware\SLM\WriteTurbulenceToSLM.m
Matlab\Hardware\Camera\1394\AutoExposureDemo.m
Matlab\Hardware\Camera\1394\TestCam1394.m
Matlab\Hardware\Camera\NICameraLink\DisplayCamLinkCams.m
Matlab\Hardware\Camera\NICameraLink\TestCamLinkCam.m
Matlab\Processing\CWFS\testCWFSProcessor.m
Matlab\Processing\HWFS\TestHWFSProcessing.m
Matlab\Processing\Kolmogorov\testKolmogorovTurbulenceScreen.m
Matlab\Processing\Kolmogorov\testKolmogorovTurbulenceScreen2.m
Matlab\Processing\StrehlCamera\testStrehlCameraProcessing.m
Matlab\Systems\HWFS_AO\TestHWFS_AO.m

9.2 *General Ethernet Interface*

AOS provides the user with the ability to communicate directly with the Ethernet interfaces of the drive electronics. Depending on the drive electronics type this process will be different. It is important to note that there will be no inter-actuator limiting performed when interfacing this way, and must be used at the user's own risk and with extreme caution. Below are the two general interfaces for drive electronics. If you are unsure of the version of your drive electronics, you can open a putty session with your drive electronics' IP and port numbers, and a raw connection as shown below.



The user can then use the command info to find out the Drive Electronics type, they will return "AOS DEPIC32 ESK v1.0" for a PIC32 and "V1" for a MicroZed type. Below is a screenshot of the info command and return for a PIC32 type Drive Electronics.



9.2.1 Netburner Drive Electronics

The process to interface with these drive electronics is as follows

- Open a TCP stream socket
- Connect to the drive electronics using the DHCP assigned IP address as found in the AOS software find devices method, or using a fixed IP set through the AOS software
- Send properly formatted commands to update actuator voltages

Below is pseudocode describing this process.

```
//There are several things we will need in order to communicate directly with the Ethernet
interface
// Firstly, we will need to create a map that will allow us to order command voltages appropriately
//We can read in the .dm file for our current DM and use the 3rd column to determine this
mapping
// say we had a 5 actuator dm with the values 0,16,22,4,9 read down the third column
NumberofActuators = 5;
int*DMfilevec;
DMfilevec = new int[NumberofActuators];
DMfilevec = [0,16,22,4,9];
//We also need the max port number for our dm. this is the largest value in this same vector +1. In
our case 22+1 = 23
int MaxPort = 23;
int* portToActMap;
portToActMap = new int[MaxPort];
// set all values in our mapping vector to -1. This will allow us to skip channels as needed
for (int i = 0; i < MaxPort + 1; i++)
        portToActMap[i] = -1;
// Loop over the number of actuators using our DMfilevec to index our mapping vector
for(int i = 0; i < NumberofActuators; i++)
{
        portToActMap[DMfilevec[i]]=1; // This gives a vector of length 32(MaxPort) with all but
5(NumberofActuators) values being -1
}


//To Communicate with the drive electronics we must open a TCP stream socket
//Create a Socket with desired attributes
m_clientSocket = new Socket(AddressFamily.InterNetwork, SocketType.Stream,
        ProtocolType.Tcp);
// Set the remote IP address
IPAddress ip = IPAddress.Parse(IPAddr);
// Create the IP end point
IPEndPoint ipEnd = new IPEndPoint(ip, port);
// Connect to the remote host
m_clientSocket.Connect(ipEnd);

//Now we are ready so send commands
// Determine byte command length. This is bytes per channel(MaxPort) plus a static 6 byte
header
// depending on your language and platform interface this will be different. it is only important that
each element in the command vector is byte
uint8_t* cmd;
cmd = new uint8_t[MaxPort * 2 + 6];
cmd[0] = (byte)'<'; // 60
cmd[1] = (byte)'M'; // 77
cmd[2] = (byte)'>'; // 62
```

```
length = (UInt16)(MaxPort * 2); // ensure the MaxPort is bytes in memory
uint8_t* lbytes;
lbytes = new uint8_t[2];// allocate memory to encode our one length value into bytes
lbytes[0]=(byte)(length & 0xff); // do the encoding
lbytes[1]=(byte)(length >> 8);

cmd[3] = lbytes[0]; // assign each byte. In our case lbytes[0] = 32 and lbytes[1]=0
cmd[4] = lbytes[1];

// we can now send our values. We have used the first 5 entries in our array, and so we will index
beginning with 5
int cnt = 5;

// our command array is numbered just like our actuators in the DM file and as displayed on the
AOS DM controller GUI
// these are 16 bit values from 0 for 0% to 65535 for 100%
uint16_t* actuatorCommands;
actuatorCommands = new uint16_t[NumberofActuators];
actuatorCommands = [0,65535,33000,23,42000]; // this demonstrates that there is absolutely no
check on limiting values when interfacing this way
for (int i = 0; i < MaxPort; i++)
  {
    if (portToActMap[i] < 0) // this checks that we have an active actuator on a given port
      val = 0; // if we do not we write nothing to it
    else
      val = actuatorCommands[portToActMap[i]]; // if we do we write the desired voltage


    valBytes[0] = (byte)(val & 0xff); // separate 16 bit command into bytes
    valBytes[1] = (byte)(val >> 8);

    cmd[cnt++] = valBytes[0]; // set bytes in command vector
    cmd[cnt++] = valBytes[1];
  }
 // now we have a properly formatted command that can be sent

if (m_clientSocket != null) // if we still have an open TCP stream
  {
    m_clientSocket.Send(cmd); // send the command vector

  }
```

9.2.2 MicroZed, PIC32, and ML605 Drive Electronics

The process to interface with these drive electronics is as follows

- Open a TCP stream socket
- Connect to the drive electronics using the DHCP assigned IP address as found in the AOS software find devices method, or using a fixed IP set through the AOS software

- Enable the device
- Send properly formatted commands to update actuator voltages
- Read back from the device to check for success

Below is pseudocode describing this process.

```
//There are several things we will need in order to communicate directly with the Ethernet interface
// Firstly, we will need to create a map that will allow us to order command voltages appropriately
//We can read in the dm file for our current DM and use the 3rd column to determine this mapping
// say we had a 5 actuator dm with the values 0,16,22,4,9 read down the third column
NumberofActuators = 5;
int*DMfilevec;
DMfilevec = new int[NumberofActuators];
DMfilevec = [0,16,22,4,9];
//We also need the max port number for our dm. this is the largest value in this same vector +1. In our case 22+1 = 23
int MaxPort = 23;
int* portToActMap;
portToActMap = new int[MaxPort];
// set all values in our mapping vector to -1. This will allow us to skip channels as needed
for (int i = 0; i < MaxPort + 1; i++)
        portToActMap[i] = -1;
// Loop over the number of actuators using our DMfilevec to index our mapping vector
for(int i = 0; i < NumberofActuators; i++)
{
        portToActMap[DMfilevec[i]]=1; // This gives a vector of length 32(MaxPort) with all but 5(NumberofActuators) values being -1
}

//To connect to a ML605 or MicroZed device we must open a TCP stream, get the stream, flush it, and then read back what the device sends us
m_client = new TcpClient(IPAddr, port);
m_client.NoDelay = true;
m_stream = m_client.GetStream();
m_stream.Flush();


//Read back 2 byte prompt. We don't care about what it returns for this call
ReadPrompt();
// Send a command to ensure that the device is a drive electronics, and to enable output
byte[] retdata = new byte[0];
SendCommandML605("HVEnable\r\n", ref retdata, 0);  // again we are not interested in this return

// We are now connected, active, and ready to send voltage commands to the DE

// Determine byte command length. This is 2 bytes per channel(MaxPort) plus a static 11 byte header
// depending on your language and platform interface this will be different. it is only important that each element in the command vector is 1 byte
byte[] cmd = new byte[portToActMap.Length * 2 + 2 + 7 + 2];
cmd[0] = (byte)'m';// 109
cmd[1] = (byte)'w';// 119
cmd[2] = (byte)'r';// 114
cmd[3] = (byte)'i';// 105
```

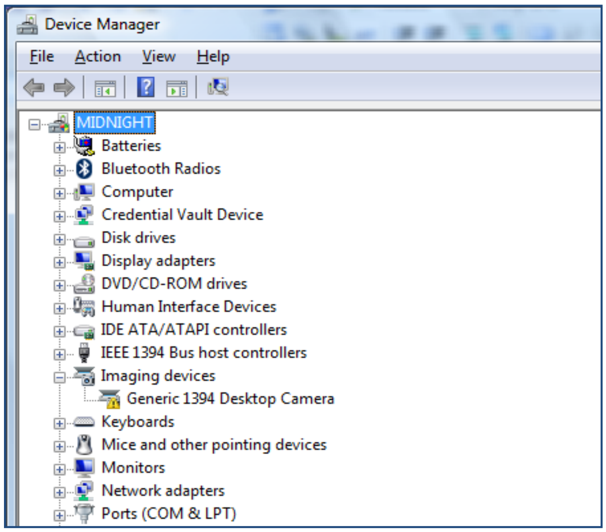
```
cmd[4] = (byte)'t';// 116
cmd[5] = (byte)'e';// 101
cmd[6] = (byte)' ';// 32

UInt16 length = (UInt16)(portToActMap.Length * 2); // length is two bytes in command vector
byte[] lbytes = BitConverter.GetBytes(length); // convert to bytes
cmd[7] = lbytes[0]; // set individual bytes
cmd[8] = lbytes[1];

// we can now send our values. We have used the first 9 entries in our array, and so we will index
beginning with 5
int cnt = 9;

UInt16 val;
uint16_t* actuatorCommands;
actuatorCommands = new uint16_t[NumberofActuators];
actuatorCommands = [0,65535,33000,23,42000]; // NO LIMITING IS PERFORMED. THIS
WOULD BE A BAD COMMAND
for (int i = 0; i < MaxPort; i++)
{
   if (portToActMap[i] < 0)
      val = 0; // if there is no actuator for this channel we write 0
   else
      val = actuatorCommands[portToActMap[i]]; // otherwise we write out our command;


   valBytes = BitConverter.GetBytes(val);
   cmd[cnt++] = valBytes[0];
   cmd[cnt++] = valBytes[1];
}

cmd[cnt++] = (byte)'\r'; // we finish the command with \r\n for carriage return and new line
cmd[cnt++] = (byte)'\n';
byte[] retData = new byte[4];

return SendCommandML605(cmd, ref retData, 0); // we check this return to see if it was '>>'=
success


// Read back function
DEWriteRet ReadPrompt()
{
        uint8_t* data;
        data = new int[2];
        int size = 2;
        int offset = 0;
        int nRead;
        while (size > 0)
        {
           nRead = m_stream.Read(data, offset, size);
           offset += nRead;
           size -= nRead;
        }
        if (promptBuf[0] == '>' && promptBuf[1] == '>')
           ret = DEWriteRet.Success; // command was successfully sent to the HVAs
```

```
            else if (promptBuf[0] == '>' && promptBuf[1] == '1')
               ret= -2; // this applies to microZed only and indicates that there was an IA violation and
the command was rejected
            else if (promptBuf[0] == '>' && promptBuf[1] == '2')
            {
               ret= -1; // improper command formatting
               WriteNetworkStatus("Error in command word" + Environment.NewLine);
            }
            else
            {
               ret= -1;
               WriteNetworkStatus("Error reading prompt!" + Environment.NewLine);
            }

}

//Send command function
 DEWriteRet SendCommandML605(byte[] cmd, ref byte[] ReturnData, int ReturnLength)
{
   DEWriteRet ret = DEWriteRet.DEError;

   if (m_stream == null)
   {
      WriteNetworkStatus("ERROR: No active connection");
   }
   else if (!m_stream.CanWrite)
   {
      WriteNetworkStatus("ERROR: No active connection");
   }
   else
   {
      lock (locker)
      {
         byte[] cmd_b = cmd;
         byte[] crnl = new byte[2];
         m_stream.Write(cmd_b, 0, cmd_b.Length);

         if (ReturnLength != 0)
         {
            ReadBytes(ref ReturnData, ReturnLength);
            ReadBytes(ref crnl, 2);
         }
         ret = ReadPrompt();
      }
   }

   return ret;
}
```

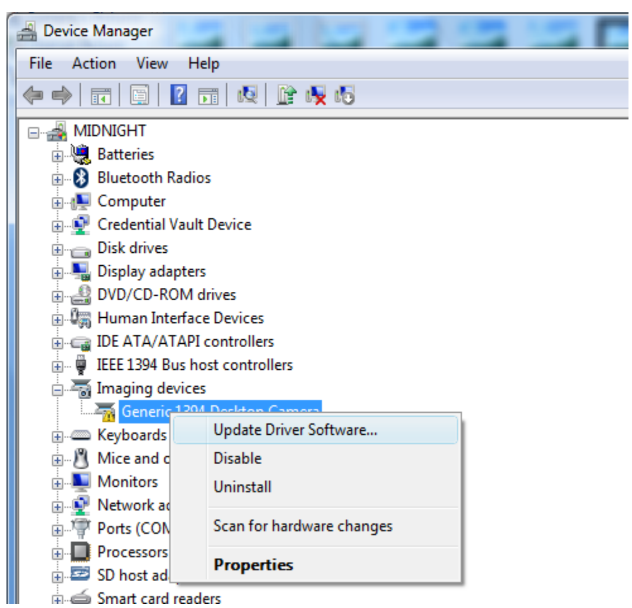
10 Appendix C: CMU Camera Driver Configuration

NOTE: The CMU FireWire camera driver is no longer actively supported by AOS primarily due to lack of 64 bit platform support. While the CMU interface currently still works with the AOS software, support for it may be dropped in future versions. Therefore, it is recommended that users switch to the Unibrain driver. For instructions on configuring the Unibrain driver, please refer to Section 2.2.1.

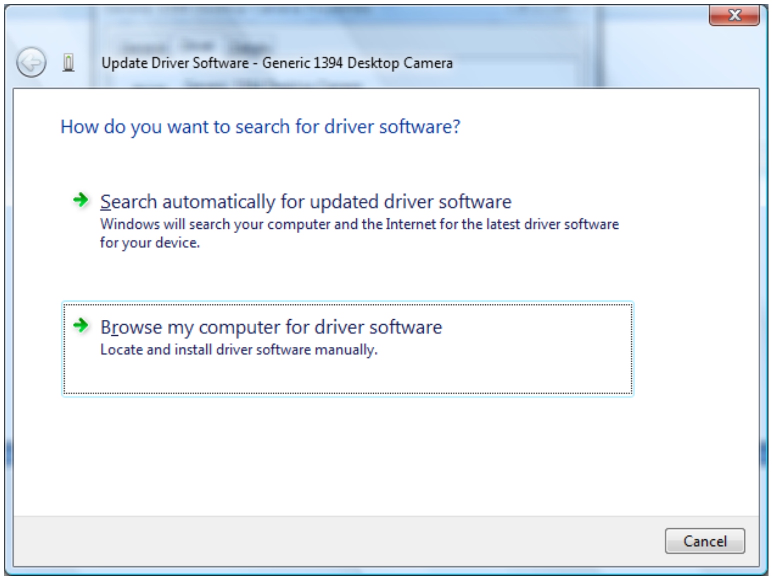
Connect the Wavefront sensor to the computer and open Device Manger.



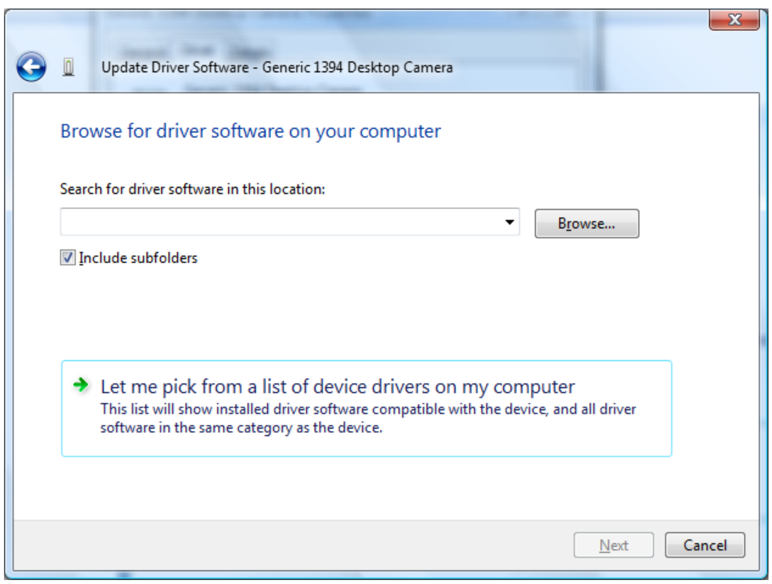
Right click on the camera and select “Update Driver Software.”



Select “Browse my computer for driver software”

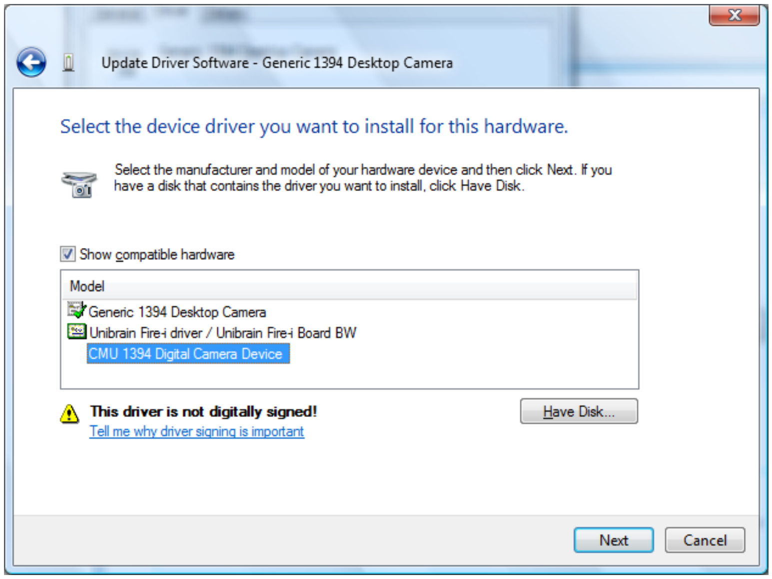


Click “Let me pick from a list…”



Select CMU 1394 Digital Camera Device. If this driver is not present, click “Have Disk” and browse to the drivers folder under the AOS software installation directory:

<AOS_DIR>\Drivers\Cameras\1394



After the driver installation is complete, the camera should be displayed as a CMU 1394 Digital Camera Device.