OASIS Application Note

Video Processing Using the OASIS-AF Module

The OASIS-AF module is a plug-in daughter board for the OASIS-4i Four Axis Motion Controller on the PCI bus. The OASIS-AF module accepts standard analogue video signals, offers a variety of built-in automatic focus operations, and is fully programmable to provide a range of video processing capabilities for high-speed and fully automatic image analysis applications.

The OASIS-AF module fits directly onto the OASIS-4i controller so that no extra power supplies or expansion slots are required. Video input and output is achieved via the primary connector interface of the OASIS-4i controller.

This tightly coupled internal design provides a variety of advantages for the system integrator:

- Internal to the control computer; no bulky, space-wasting external control boxes are required
- All power is obtained via the 5V PCI bus and 12V internal PC power supply
- High-speed interface between motion controller and video processor creates a fully independent automation and video analysis sub-system for maximum performance

This Application Note describes the use of the OASIS-AF module in detail.

What You Need

The following items are required to use the OASIS-AF module:

1. An OASIS-4i controller installed in a PCI expansion slot.
2. A control computer running Microsoft Windows 95/98, Windows NT 4.0, or Windows 2000.
3. A standard, analogue video camera.
4. A connecting cable for the camera.

If automatic focus is required, you will also need to have a microscope fitted with a motorised focus mechanism using a stepper motor that can be controlled by the OASIS-4i hardware.

To access the OASIS hardware, you will need the OASIS4I Dynamic Link Library (DLL). This is installed automatically for you whenever the OASIS-4i controller is installed using Windows Plug-and Play. Also included with the OASIS-4i system is the OASIS4I DLL Software Developer Kit, which includes the manual describing all of the software functions used to interface to the OASIS hardware. All the specific DLL functions mentioned in this document are described in further detail there.

Video Input

The OASIS-AF module accepts video from standard analogue sources, according to the following video formats:

- PAL (colour) and CCIR (mono)
- NTSC (colour) and RS-170 (mono)
- SECAM (colour)

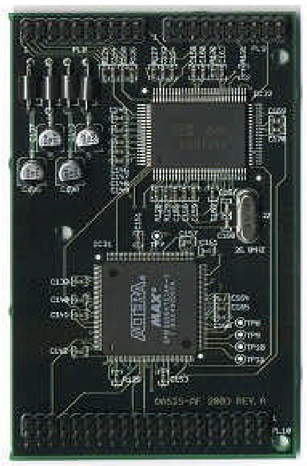


Figure 1. OASIS-AF module

The PAL / CCIR formats are popular in European countries, while NSTC / RS-170 are used in North America and Japan. SECAM is used in France and Russia.

The OASIS-AF module accepts monochrome, composite colour, and Y/C colour video signals. The particular video format in use is detected automatically by the OASIS-AF module.

A brief review of the characteristics of standard video will be helpful and particularly useful later when we discuss how to position and size of the OASIS-AF's video window. It will also come in handy when we discuss how area and chord size measurements are set up.

Standard analogue video is composed of a number of full video *frames* representing the image detected on the video camera's sensor. For PAL and CCIR video, 25 video frames are updated every second. This format uses 625 video lines, with each line corresponding to a row formed by the raster sweep of the video signal across the image. Some of the vertical signal is composed of non-visual information such as blanking, so the total number of visual lines is actually about 576 lines.

Each video line in this format can be decomposed into 768 discrete picture elements ("pixels"). Therefore the PAL and CCIR video formats provide an effective resolution up to 768 x 576 pixels per frame.

For NTSC and RS-170 video, 30 video frames are updated every second. Each frame contains 525 video lines, of which about 480 contain visual information. Each line in this video format can be decomposed into 640 pixels, giving a total resolution of 640 x 480 pixels per frame.

In all of these formats, each full video frame uses a 2:1 interlacing scheme, where the full frame is divided into two separate scans. One scan defines the odd video lines and another defines the even video lines. Each of these odd/even scans is called a video *field*.

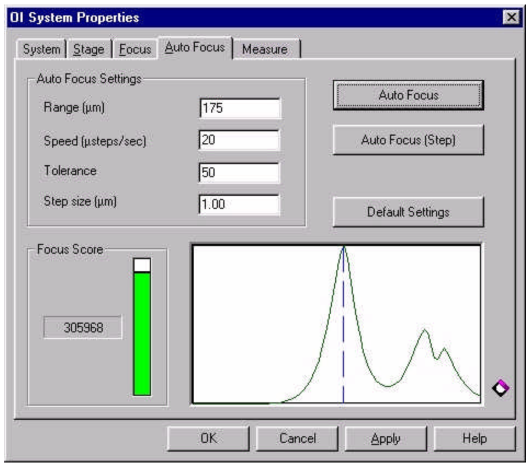
Since the full frame is composed from these two successive scans, the video field rate is double that of the full video frame. So the video field rate for PAL and CCIR is 50 fields per second, while NTSC and RS-170 yield 60 fields per second. However, the vertical resolution is half the full frame resolution. For instance, the vertical resolution of the odd and even video fields in CCIR is effectively 288 lines, i.e., half of the 576 lines visible in the full frame.

For our purposes, the SECAM video format may be considered the same as the PAL format.

Now let's look at a primary application of the OASIS-AF module: Automatic Focussing.

Automatic Focus

Scanning applications in microscopy most often employ three automated axes. The first two drive the X and Y directions of a motorised stage, allowing a specimen to be translated under the primary objective lens. As the specimen is moved, very often the focus of the specimen is lost. This could be due to the stage not being perfectly orthogonal to the optical axis, or the sample itself may not be flat. In any case, this practical problem means that if fully automatic scanning of the specimen is required, the focus mechanism of the microscope needs to be motorised. In addition to this Z axis motorisation, we also need some process by which the optimised focus position is determined automatically.



Auto Focus Settings

Range (µm)	175
Speed (µsteps/sec)	20
Tolerance	50
Step size (µm)	1.00

Focus Score: 305968

Figure 2. Focus Score Plot.

The OASIS-AF module provides this by analysing the incoming video signal and calculating the degree of focus automatically. This is done by looking at the sharpness of the edges in the image to generate a *focus score* that indicates the relative degree of focus. Higher focus score values correspond to more and/or sharper edges present in the video signal.

When the Z-axis is motorised, the OASIS-AF module and OASIS-4i controller work together to move the Z-axis and check the focus score until an optimal focus position is found. The standard OASIS autofocus algorithm moves the focus over a pre-defined range of Z-axis travel, while continuously and simultaneously sampling the focus score during the movement. Figure 2 shows a plot of the focus score versus Z position data that results from an automatic focus operation. The optimal focus is defined as the centre of the highest peak in that graph, as indicated by the dashed vertical line.

DLL Functions for Autofocus

Several functions in the OASIS4I DLL allow you to perform and configure the automatic focus operation performed by the OASIS hardware. The most straightforward is the function that initiates an autofocus using the current settings:

OI_AutoFocus()

This function tells the OASIS hardware to begin the autofocus. The function returns immediately, i.e. it does not wait for the autofocus operation to finish. This is by design, because it allows an application to start the hardware autofocus while the application can continue on doing some other tasks asynchronously, such as analysing a previously acquired image or writing results to disk.

Since of course at some point you will want to ensure that the autofocus is indeed finished, the OASIS4I DLL provides this function:

OI_WaitForAutoFocus()

This function continuously checks the status of the autofocus and returns only when the autofocus is complete, has timed out, or has been aborted by the user pressing the ESCAPE or CTRL-C keys. The return value of the function indicates which of these three options occurred.

You can also check on the status of the autofocus at any time by calling:

```
OI_RequestAutoFocusStatus(
        LPWORD pwFinished, // finished flag
        LPWORD pwSuccess,  // did it work?
        LPWORD pwNumSamples   // # samples taken
        )
```

See Listing 1 at the end of this document for an example of XY scanning with automatic focus using the OASIS-4i Motion Controller and OASIS-AF Autofocus module.

Autofocus Settings

The operation of the autofocus is determined by four parameters:

- The Z range used in the search
- The speed at which the Z axis is moved over that range
- The tolerance value used to define a peak
- The threshold value used when calculating the focus score

A couple of DLL functions are used to set these up. The first allows you to define the default values for the Z axis range to be searched (in microns), the speed at which the Z axis is moved, and a tolerance value that helps minimize the effects of erroneous sub-peaks in the focus score data:

```
OI_SetAutoFocus(
        double dRange, // the z range
        int wSpeed,  // the cruise speed
        int wTolerance // peak tolerance
        )
```

If you want to pass these parameters directly to an autofocus function, use:

```
OI_AutoFocusEx(
        double dRange,
        int nSpeed,
        int nTolerance
        )
```

Since the video frame rate is fixed, if you choose a higher speed for the autofocus, then you will get fewer focus score samples for a given range. This may lead to gaps in the data and may affect the accuracy of the autofocus. Conversely, a slower speed for a given range will result in a larger number of focus score samples, giving more precise results but at the price of a longer overall autofocus time.

The proper setting for each of these parameters depends on the magnification in use and the type of specimen being viewed. For instance, higher magnifications may require a smaller range of travel and a slower speed due to the smaller depth of field in high power lenses. On the other hand, the larger depth of field of lower power lenses means a larger range of travel and a faster speed may be used.

Another setting that affects the performance of the autofocus is the focus score threshold. This value is applied to the incoming video so as to exclude noise from the video signal. To set the autofocus threshold use:

```
OI_SetAutoFocusThreshold(
        WORD wThresh  // can be 0-255
        )
```

Higher threshold values cause globally smaller focus score values, so that noisy variations are minimized.

Reading the Focus Score

The OASIS-AF module continuously calculates the focus score on the incoming video signal for each frame. To read the current focus score, call:

```
OI_ReadFocusScore(
        double* pdScore  // the returned score
        )
```

You may also wish to read the Z position value that corresponds to the focus score, by calling:

```
OI_ReadFocusScoreEx(
        double* pdScore,  // the returned score
        double* pdZPos   // the Z position
        )
```

The asynchronous nature of the OASIS system means that you can call these functions at virtually any time. For instance you may wish to initiate a movement of the Z axis then continuously monitor the focus score and Z position using the **OI_ReadFocusScoreEx** function.

Later on, in the Measurements section, we describe how you can combine multiple axis movements with focus score and other measurements for an extremely flexible system for automatic focus and rare event scanning applications.

Reading the Focus Score Profile

After an autofocus operation has been completed, the focus score profile—giving the focus score values for each Z position sampled during the autofocus—may be read using this function:

```
OI_ReadFocusProfile(
        double* pdArray,  // the focus score values
        double* pdZPos,   // the Z position values
        int nSize,   // the size of the arrays passed
        int* pnSamples  // the actual # samples taken
        )
```

This function fills in a couple of arrays with the focus score values and corresponding Z position from the previous Autofocus command.

A possible application making use of this data could be to automatically measure the thickness of a sample based on the shape of the focus profile curve. Another situation where this data could be used is to handle special cases where more than one focus peak is generated by the sample and you wish to choose not the highest peak (i.e. the peak chosen by the default autofocus algorithm), but perhaps instead wish to use the relation of one peak to another.

More specialized variations of automatic focus are described below in the Measurements section.

Measurements

The OASIS-AF provides a very powerful set of features for high-performance measurements based on the incoming video fields. In particular, the OASIS-AF module produces real-time measurements of total detected area, maximum chord length, and maximum gradient value. These measurements are made directly on the incoming video signal, allowing applications to check these results prior to acquiring an image using a digitising frame grabber. This increases the efficiency of scanning and therefore can significantly improve the overall throughput of the analysis process.

Applications for these measurements include:

- Specialised automatic focus algorithms
- Rare event detection
- High speed scanning using blank field bypass

Before we get into the details of how the OASIS-AF's measurements are set up and the results are read back, let's look at the video window in which all measurements occur.

The Video Window

Earlier we discussed how standard video is defined in terms of *frames* of interlaced video *fields*. The full video frame has either of two main resolutions: the 768x576 size of PAL/CCIR/SECAM or the 640 x 480 size of NTSC/RS-170. These values represent the maximum extent of the full video frame, from which we derive information such as focus scores or other measurements.

The OASIS-AF module allows us to further refine the area in the video we are dealing with by using a *Video Window*. The Video Window is a rectangle that specifies exactly where the calculations occur. For instance, an application may wish to restrict focus score calculation to a smaller area near the centre of the video frame and exclude information near the edges. The Video Window allows us to achieve that.

The OASIS4I DLL function that sets up the Video Window is:

```
OI_SetVideoWindow(
        int nXStart,  // left edge of window
        int nXStop,   // right edge of window
        int nYStart,  // top edge of window
        int nYStop    // bottom edge of window
        )
```

The Video Window is defined by the pixel locations of the edges of the rectangle. These values should be in the range appropriate to the video standard in use. For instance, if you're using PAL/CCIR video, then the X values should be in the range of 0 to 767 and the Y values should be in the range of 0 to 575. For NTSC/RS-170 video, these ranges are 0-639 and 0-479, respectively.

There are some restrictions on the size and position of the Video Window imposed by the OASIS-AF hardware. The main restriction is that the top edge of the Video Window cannot be greater than 254, roughly halfway down the video field. Also, the width of the Video Window should be a multiple of 4 pixels (0, 4, 8, etc).

Bearing these restrictions in mind, the placement of the Video Window allows us to hone in on regions of the image that are of interest to us, and ignore areas that we wish to avoid.

OASIS-AF Real-Time Measurements

In addition to the focus score values described earlier, the OASIS-AF module also provides these measurements:

- Total number of detected pixels (i.e., area)
- Maximum chord length of detected pixels
- Maximum gradient value of detected pixels

The OASIS-AF module makes the measurements continuously—at video rates--on each video *field*.

In order to identify the features of interest that are to be measured, two settings must be defined. The first is the *Phase* of interest. This can be set to detect either the darker or the lighter parts of the image.

The second setting is the *Threshold* level to be used to identify the desired phase. The OASIS-AF module digitises the incoming video signal's brightness into a scale of 0 to 255, where 0 corresponds to low intensity (dark) and 255 indicates high intensity (light). Note for monochrome cameras, the brightness is simply the video signal intensity, whereas for colour cameras, the brightness is the intensity component of the composite or the Y/C colour signal.

The combination of the Phase and Threshold tells the OASIS-AF module what to detect.

For instance, a Threshold value of ***50*** and a Phase of ***Dark*** will cause the OASIS-AF to detect all pixels in the Video Window that are from intensity 0 (black) to 50 (a dark grey). The measurement results will then tell you the total number of pixels that are detected with these settings, the pixel length of the longest continuously detected horizontal chord in the image, and the maximum gradient value found in the detected pixels.

The DLL function used to specify these settings is:

```
OI_SetVideoSettings(
    int nEvenPhase,  // Even interlace phase
    int nEvenThreshold,  // Even interlace threshold
    int nOddPhase,   // Odd interlace phase
    int nOddThreshold  // Odd interlace threshold
    )
```

You'll notice that the **OI_SetVideoSettings** function requires two sets of Phase and Threshold values. This is because the OASIS-AF module actually allows us to specify separate Phase and Threshold values for each of the two interlaced video *fields*. In effect, the OASIS-AF module simultaneously measures two separate phases for each video frame.

This flexibility allows more complex detection and analysis schemes. However, if only one phase is required, then the Even and Odd video field values can be set to the same values.

Reading the Video Measurements

Like the focus score values, the video measurements are being continuously made by the OASIS-AF hardware. To read the results, you simply "tap" into the results at a given moment. You can do this using this DLL function:

```
OI_ReadVideoData(
    LPWORD nStatus, // Results code
    LPDWORD pdwArea,  // Total # pixels
    LPWORD pwMaxChord,   // Longest chord
    LPWORD pwMaxGradient  // Highest gradient
    )
```

The *nStatus* parameter indicates whether the results are for the Odd video field (*nStatus*=1), the Even video field (*nStatus*=2), or the results are not valid (*nStatus*=0).

To read the results from a specific video field, or for the full video frame, use this function:

```
OI_ReadVideoResultsEx(
    int nMode,  // Results type indicator
    LPDWORD pdwArea,   // Total # pixels
    LPWORD pwMaxChord,   // Longest chord
    LPWORD pwMaxGradient  // Highest gradient
    )
```

The *nMode* parameter indicates whether you desire the Odd field results (*nMode*=1), the Even field results (*nMode*=2), or you want the combined results for the full video frame (*nMode*=3).

The OASIS4I DLL also offers some further variations for reading the video results. For instance, you may wish to read the video results along with the focus score and Z position in order to create a customised automatic focus algorithm. This function may be used:

```
OI_ReadVideoResultsZ(
    int nMode,  // Results type indicator
    double* pdResults,  // Results destination array
    double* pdZPos   // Z position value
    )
```

Here the *pdResults* parameter is an array of 4 values passed into the function where the focus score, area, maximum chord and gradient data are to be copied. The *pdZPos* parameter returns the Z position corresponding to those values.

For instance, if an application wishes to create a customised autofocus based on maximizing the detected area, the **OI_ReadVideoResultsZ** function could be used to read how the size of a detected region in the image changes as the Z axis is continuously moved. An example of this is shown in Listing 2 at the end of this document.

Summary

Reliable, accurate automatic focus is an essential component wherever the automated scanning of microscopic specimens is used. The OASIS-AF Autofocus module, when coupled with the OASIS-4i Four Axis Motion Controller, provides a powerful set of built-in functions for the automatic determination of best focus, using any of the supported motorised microscope focus mechanisms. A variety of programmable settings ensure the hardware-based video autofocus of the OASIS-AF is tailored easily to specific specimens and imaging conditions.

In addition to these built-in automatic focus facilities, the OASIS-AF's real-time video measurement functions ensure maximum flexibility and performance for specialized applications in high-speed scanning and fully automatic microscopy solutions. The incorporation of video-rate image analysis into the OASIS-AF and OASIS-4i motion control system represents an important step to the next level of automation control for microscopy and imaging applications.

For assistance in finding out how the OASIS line of automated microscopy and imaging products can help with your specific requirements, please contact Objective Imaging for further information.

Listing 1. Scan with AutoFocus Example.

```
#include "oasis4i.h"
void ScanRectWithAutoFocus(
   int iXFields, int iYFields, // # of fields in X,Y
   double dXStep, double dYStep  // step size in X,Y
)
{
   // read where we are now
   double dXStartPos, dYStartPos;
   OI_ReadXY( &dXStartPos, &dYStartPos );

   // start scanning
   for(int iY=0; iY<iYFields; iY++)
   {
      for(int iX=0; iX<iXFields; iX++)
      {
         // move to the next XY position, no waiting
         OI_MoveToXY(dXStartPos + iX * dXStep,
                        dYStartPos + iY * dYStep, 0 );

         // some application processing could go here…

         // now wait for stage to get where it's going
         OI_WaitForStoppedXYZ( 1, 1, 0 );

         // autofocus the current field
         OI_AutoFocusEx( 250, 200, 10 );

         // some more application processing could go here…

         // now wait until we've got the focus
         OI_WaitForAutoFocus();

      } // for iX
   } // for iY
   return;
}
```

Listing 2. Custom AutoFocus Based on Video Area Measurement.

```
#include "oasis4i.h"
void CustomAutoFocus(
   double dZRange // the range to look over, in microns
)
{
   double dResults[4];
   double dZPos;
   WORD wStatus;

   double dZPosAtMax;
   double dAreaMax = 0;

   // Read where we are, so if we don't find a max,
   // come back here
   OI_ReadZ( &dZPosAtMax );

   // Move to one end of range, wait until we arrive
   OI_StepZ( -dZRange/2, 1 );

   // Start the move to the other end of the range,  no waiting
   OI_StepZ( dZRange/2, 0 );
   do
   {
      OI_ReadVideoResultsZ( 3, dResults, &dZPos );
      if( dResults[1] > dAreaMax )
      {
         dAreaMax = dResults[1];
         dZPosAtMax = dZPos;
      }
      OI_ReadStatusZ( &wStatus );
   } while( wStatus & S_MOVING );

   // Move back to where we found the max
   OI_MoveToZ( dZPosAtMax, 1 );
   return;
}
```

Innovative Solutions for Automated Microscopy

Objective Imaging Ltd   Tel (Europe) +44 (0)1223 842471
Tel (U.S.A.)   +1 (847) 376-8148   www.objectiveimaging.com