R.Coles

DESI CI Metrology Software

Centroiding Offsets for CCDs and FIFs

Goal: To find the CS5 location of an illuminated object that is imaged using the SBIG ST-I and STXL-6303 cameras that is attached to the DMM, while the DMM is attached to a CMM.

Necessary Offsets: Offsets that are needed to connect the position of an illuminated object in an SBIG ST-I or STXL-6303 image with the objects physical location in CS5 coordinates. The illuminated object will usually be either a pinhole on a FIF, or the light from a pinhole that is projected onto a camera's CCD. The necessary offsets are:

1. **Light from 100um DMM Pinhole in ST-I Image (Needed for CCD Metrology Measurements)**
   When we project light from the DMM's 100um pinhole onto a surface (usually a CCD), and image it using the DMM's ST-I camera, we need to know the distance from where the projected pinhole shows up on a ST-I full frame image (in units of rows and columns) to the corner of that image (rows = 0, column = 0). The image used to find this off set is created by focusing the light from the 100um pinhole on the DMM onto a diffuse surface and analyzing the resulting ST-I image. This measurement is completed before the DMM is attached to the CMM, and the resulting distance is a static variable in the DESI CI Metrology Software.

   Example: The user tells the CS5 calibrated CMM to move to CS5 position (X=25, Y=25) over a CCD. We then project light from the 100um pinhole on the DMM onto the CCD and take an image with the ST-I camera. We then record the (rows, column) location of the pinhole light from the ST-I image.

   How we use this offset: We use this offset, in combination with the offset described in Necessary Offsets bullet point #2, to find where the light from the 100um pinhole shows up in a ST-I full frame image relative to the location of the desired CS5 position in the same image, as described in Necessary Offsets bullet point #3.

2. **CS5 Point on CI Dowel as shown in ST-I Image (Needed for FIF and CCD Metrology Measurements)**
   When we tell the CS5 calibrated CMM to move to a given CS5 location, we will need to know where the specified point will show up in the associated ST-I image taken with the DMM for that point. To find this offset in units of (rows, columns), we perform the following measurements prior to mounting the DMM on the CMM:
   1. The CI will have two dowels placed at know locations. Each dowel will have a divot machined on top. Prior to mounting the DMM on the CS5 calibrated CMM, we use the CMM's touch probe to find the exact location of the divots in CS5 coordinates.
   2. After attaching the DMM to the CMM, we tell the CMM to move to the CS5 location of the divot.
   3. We take an image of the divot using the ST-I camera on the DMM.
   4. We input the image into the DESI CI Metrology software, and it located the location of the divot in units of (rows, column).

How we use this offset: this offset tells us where an object that we image with the ST-I (usually a FIF pinhole or light from the 100um pinhole on the DMM), at a given CS5 location, will appear in the image. Put simply, the ST-I is not mounted on the DMM such that an imaged object appears directly in the center of the image, and this offset tells us where the imaged-object origin is in ST-I images.

3. **CMM Offset for 100um DMM Pinhole Projection (Needed for CCD Metrology Measurements)**
This offset combines the offsets that were measured in in Necessary Offsets bullet points #1 and #2. This offset accounts for the distance of projected light from the 100um pinhole in the ST-I images to the desired location that the CMM was told to move to (as shown in the image).

Example: While in the lab, we look at an image taken with the intent of calculating where light that is projected from the 100um pinhole on the DMM appears in the frame in units of (rows, columns) as described in Necessary Offsets bullet point #1. We centroid the image of the pinhole light and find that it is at (24, 25). Later, after having followed the procedure explained in in Necessary Offsets bullet point #2, we have a second image that shows where a desired CS5 location, as given to the CMM, shows up in an image in (rows, columns). We find that the desired CS5 location shows up at (32, 34) in the ST-I image. The difference between these two points is:
Rows: 32-24 = 8
Columns: 34-25 = 9
This means that the projected light from the 100um pinhole is eight rows and 9 columns away from where it needs to be if we want it to project onto the CCD at a given CS5 location. Therefore, the CMM Offset for 100um DMM Pinhole Projection is (8, 9). Since we know the size of the camera pixels,

How we use this offset: we use this offset to ensure that the light from the 100um pinhole on the DMM falls onto the CCDs where we would like. In other words, if we would like the projected light from the 100um pinhole on the DMM to fall on CS5 location (X = 100 mm, Y = 100mm), but this offset tells us that telling the CMM to move that location would cause the projected light to fall a known distance away from that location, so we can tell the CMM to move to CS5 location (X = 100mm + offset mm, Y = 100mm + offset mm) to have the projected light fall on the CCD at CS5 location (X = 100 mm, Y = 100mm).

Centroid Software

Class:
centroidPinHoleImage:
This module holds a series of functions used to find the center of an illuminated pin hole from an image using code that was adapted from the IDL Astronomy Users Library. The algorithm computes the centroid of a star using a derivative search (adapted for IDL from DAOPHOT, then translated from IDL to Python). It uses an early DAOPHOT "FIND" centroid algorithm by locating the position where the X and Y derivatives go to zero. The original python conversion was performed by David Jones.

Module:
findCentroid:

This module takes a Numpy array of an image and centroid the pinholes within. The function computes the centroid of a star using a derivative search. The maximum pixel within distance from input pixel is determined from the FHWM and is found and used as the center of a square, within which the centroid is computed as the value (XCEN,YCEN) at which the derivatives of the partial sums of the input image with respect to (x,y) = 0. In order to minimize contamination from neighboring stars, a weighting factor W is defined as unity in center, 0.5 at end, and linear in between. Values for xcen and ycen will not be computed if the computed centroid falls outside of the box, or if the computed derivatives are non-decreasing. If the centroid cannot be computed, then a xcen and ycen are set to -1 and a message is displayed.