|  |
| --- |
| Method of Estimating Beam Size from VELA Screen Images |
| M.S. Toplis & D.J. Scott  ASTeC: Accelerator Physics Group  July 2016 |
|  |

|  |
| --- |
|  |

**Abstract**

A method is presented for characterising beam distributions on YAG screens on VELA. The procedure has been developed as a starting point for ‘on the fly’ image analysis and thus the aims are to robustly and automatically analyse as many images as possible. Mean vectors and covariance matrices of the beam distributions are calculated to give estimates of the beam centroid and size, after pre-processing, using two methods: least squares fitting of a bivariate normal distribution and by calculating the 1st and 2nd moments of the distribution which are the maximum likelihood estimators of a normal distribution. Both methods give comparable results and the advantages and disadvantages of both are discussed. Images taken as part of the Model Validation Project [1] have been used to develop this procedure and all images taken were analysed successfully.

**Introduction**

The goal of this project is to create a robust, automated procedure to characterise beam distributions recorded YAG screens to be used for online image analysis on VELA.

The procedure is split into 3 sections (steps outlined below):

* Pre-Processing (steps 1-3)
  + Cutting an image down to just the beam signal
* Processing (steps 4 & 5)
  + Fitting a Bivariate Normal distribution to the data (BVN method)
  + Directly calculating the 1st and 2nd moments of the distribution which are the maximum likelihood estimators of a normal distribution (MLE method)
* Post-Processing (steps 6 & 7)
  + Rescaling the values of the beam centroid position
  + Converting from pixels to millimetres
  + Present centroids relative to machine axis (this has not been done as of yet, although it is a future goal to determine the positions of the screens relative to the machine axis using a HeNe laser, this has been done using the electron beam for the Model Validation Project [1])

This report will explain the procedure using an example image recorded on YAG-1 at 50 pC. The routine has been prototyped in Mathematica and there is an accompanying Mathematica notebook with annotations [2].

It will also show some examples of analysis of more challenging images for the procedure such as low charge images and images where the beam is not fully on the screen.

Figure 1 shows the example image and its projections in the x and y axes with some common image artefacts labelled. The image projections are the sum of pixel intensity values along each row and column of the image data array, giving a histogram of intensities in both the x and y directions.

Beam Signal

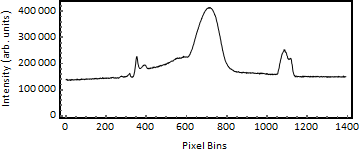
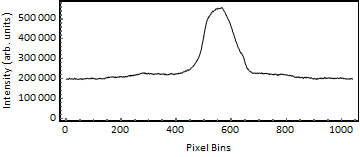
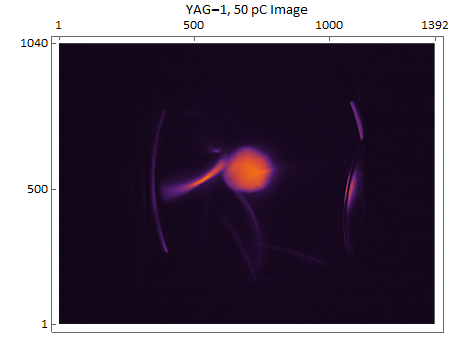


Figure : The example image (taken on YAG-1 at 50pC) and its image projections with common image artefacts labelled in red. The beam pipe reflections are caused by light emitted from the screen being reflected from the edges of the beam pipe and into the camera.

Dark   
Current

Beam  
Pipe

Reflections

**Theory**

To characterise a beam distribution on a YAG screen we estimate the beam position, beam size and x,y correlation by calculating the mean vector ***μ*** and covariance matrixof the distribution:

|  |  |  |
| --- | --- | --- |
|  |  | ( 1 ) |

where μx is the mean in the x direction and μy is the mean in the y direction.

|  |  |  |
| --- | --- | --- |
|  | , | ( 2 ) |

where σxx and σyy are the variances along the x and y axes and σxy = σyx is the covariance.

The distribution recorded on the screen can be considered as a Bivariate Normal Distribution with probability density (x,y) as given in equation ( 3 ) :

|  |  |  |
| --- | --- | --- |
|  |  | ( 3 ) |

where B is the background level, A is a proportionality factor and are the mean vector and covariance matrix respectively.

Figure 2 shows an illustration of a contour plot of a Bivariate Normal Distribution with labelled:

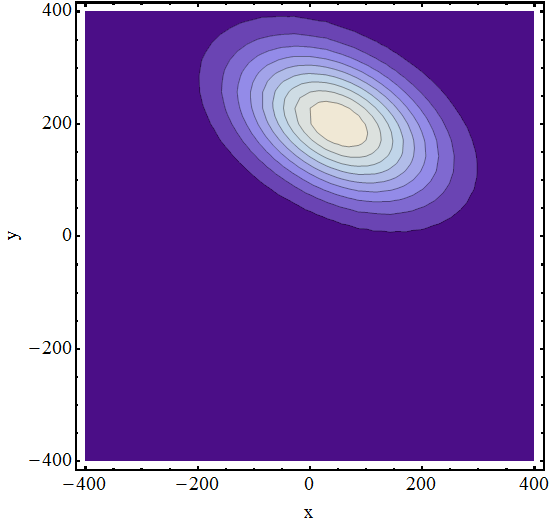
σyy

σxx

Figure : Illustration of the components of *μ* and on a contour plot of a Bivariate Normal Distribution.

μx

μy



Estimates for andare calculated by fitting equation ( 3 ) to the distribution (BVN) as well as by directly calculating the 1st and 2nd moments of the distribution (MLE) using equations ( 4 ) and ( 5 ).

|  |  |  |
| --- | --- | --- |
|  |  | ( 4 )  ( 5 ) |

where *X* is the distribution and E is the expectation value.

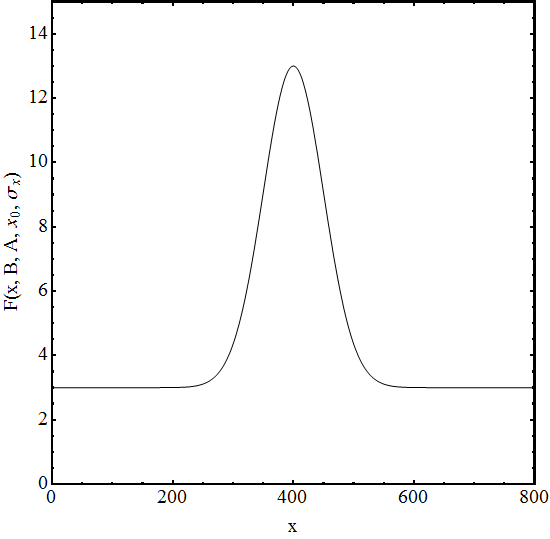
This gives two sets of values for the mean vector and covariance matrix, one from the BVN method and one from the MLE method.

To improve fitting speed, estimates of are calculated by fitting the equation of a 1D Normal Distribution, given in equation ( 6 ), to the image projections shown in Figure 1 and extracting the mean position μ0 and standard deviation σ0:

|  |  |  |
| --- | --- | --- |
|  | , | ( 6 ) |

where x is a coordinate in 1D, μ0 is the mean position, σ0 is the standard deviation, B0 is the background level and A0 is a proportionality factor.

A diagram of a 1D Normal Distribution is shown in Figure 3:



*B0*

*A0*

*μ0*

σ0

Figure : Diagram of a 1D Normal Distribution that is fit to the x and y image projections, the fit parameters from equation ( 6 ) are labelled in red.

**Image Analysis Procedure**

The full procedure consists of 7 steps:

1. Cropping the image to the screen

Applying a mask to the image to cut away the screen holder and beam pipe, leaving only YAG screen.

1. Subtracting a background
   1. If a background image has been taken then this is subtracted from the image. If not then estimate the background level and subtract from the image.
   2. Apply filters to the image projections to reduce noise.
2. Cutting the image down to just the beam signal
   1. Fit 1D Normal Distributions to the image projections
      1. Calculate estimates and use them to fit the 1D Normal distributions
      2. Use the estimates to decide which projection filter to use.
   2. Cut the image based on the 1D Normal fit parameters.
   3. Fit 1D Normal to the cut image projections.
3. BVN Method
   1. Fit the 2D Normal distribution to the cut image data using the fitting parameters from the 1D Normal fits of the cut image projections as estimates.
   2. Extract values of ***μ*** and from the fit.
4. MLE Method
   1. Directly calculate the 1st and 2nd moments of the cut image distribution which are the maximum likelihood estimators of ***μ*** and of a normal distribution.
5. Rescaling the centroid position

Rescale the values of ***μ*** to account for cutting and cropping the image.

1. Converting to millimetres

Convert ***μ*** and from pixels to millimetres.

8 Centroids relative to machine axis (not included in this report)

1. **Cropping The Image To The Screen**

The mask is a data array that describes the shape of an ellipse, every element inside the ellipse is equal to 1, everything outside is equal to 0, as shown in Figure 4 and equation ( 7 ):

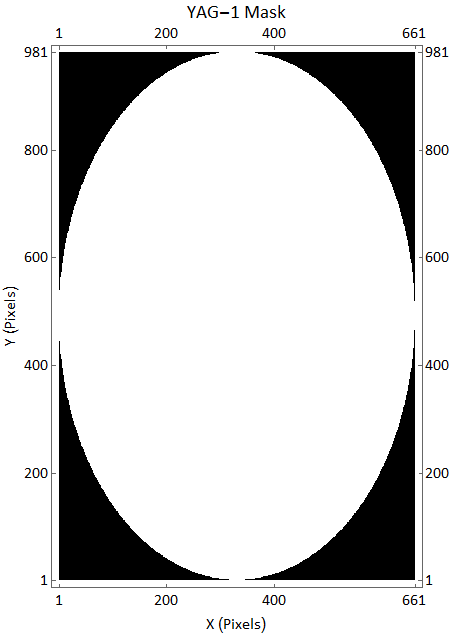


Figure : The mask that is applied to YAG-1, black = 0 white = 1.

|  |  |  |
| --- | --- | --- |
| , |  | ( 7 ) |

where x and y are the Cartesian coordinates, x0 and y0 are the centre of the mask and rx and ry are the radii of the ellipse in the x and y directions respectively.

When multiplying an image data array by this mask, everything outside the white region will be multiplied by 0 and everything inside the white region will be multiplied by 1. This has the effect of cropping the image down to just the YAG screen.

The projections of the mask are shown in Figure 5:

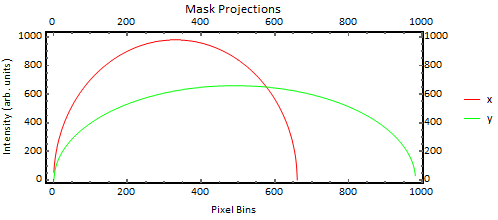


Figure : X (red) and Y (green) projections of the mask applied to the image. The curving towards the edges of the projections is a consequence of the mask being elliptical, the projections are made by summing the intensity values along each row/column, therefore the rows/columns towards the edge of the ellipse will contain some elements with 0 intensity so their total will be much lower than rows/columns closer to the centre.

The masks are adjusted to fit each screen offline using an image of the screen to set the dimensions of the ellipse. The mask for YAG-1 is shown in Figure 6, the red ellipse shows the dimensions of the ellipse in the mask array.

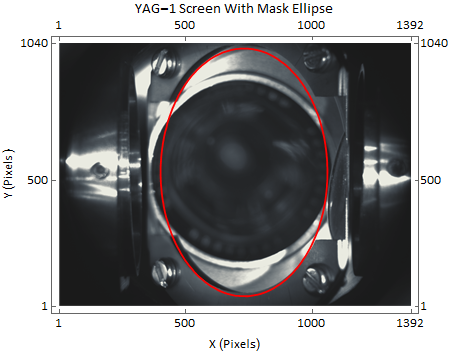


Figure : Image of YAG-1 with the mask ellipse (red) overlaid, {x0,y0} is the centre of the ellipse and rx and ry are the ellipse radii in the x and y direction respectively. The blue lines show where the image could be cut using a step function; note that much more of the screen is included if the area cropped to is extended to include the blue lines.

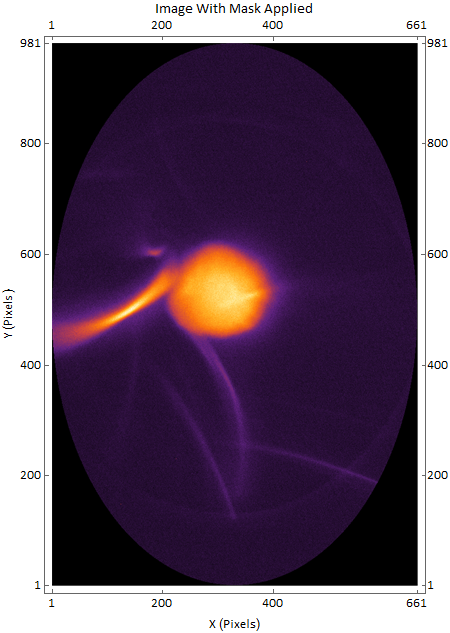
{x0,y0}

rx

ry

Using an elliptical mask gives a good estimation for the shape of the screen in the holder, however a small amount of YAG screen is still cut away. An improvement that could be made to the design of the mask is using a step function to fit the mask exactly to the edges of the YAG screen, as shown by the light blue lines in Figure 6.

Figure 7 and Figure 8 show the example image and its projections after the mask has been applied:



**Figure 7: Example image after the mask has been applied, the black corners of the image are where the intensity of the pixels is zero.**

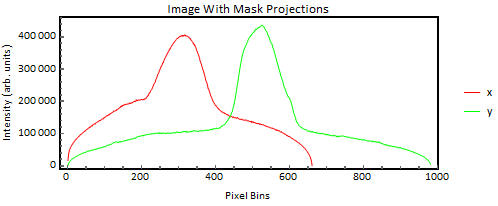


Figure : X (red) and Y (green) projections of the image after the mask has been applied, the curving towards the edges are a result of the projections of the mask, as explained in Figure 5.

Figure : Example image after the mask has been applied, the black corners of the image are where the intensity of the pixels is zero.

1. **Background Subtraction**

Subtract a background to reduce noise in the image data. If a background image has been taken then subtract this from the image, if not then we must estimate the background as shown in section 2.1.2.

**2.1.1. Subtracting a Background Image**

If a background image exists then this is subtracted from the original image before the mask is applied in step 1 , this should get rid of most of the artefacts. Figure 9 shows the example image after a background image has been subtracted.

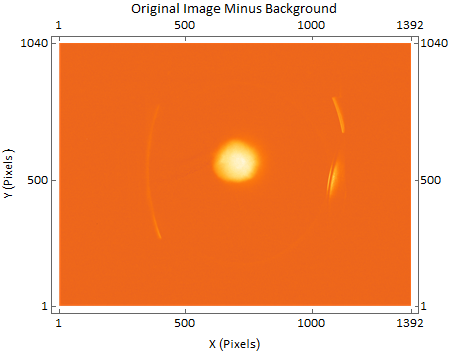


Figure : The example image after a background image has been subtracted, the reflections caused by the beam (labelled in red) are still visible as these don’t exist in the background image so will not be subtracted.

Reflections

**2.1.2 Estimating The Background**

If a background image does not exist then the background intensity level is estimated and subtracted. This is done after the mask has been applied using two methods, the Scaled Mask Subtraction method and the N Point Scaling method.

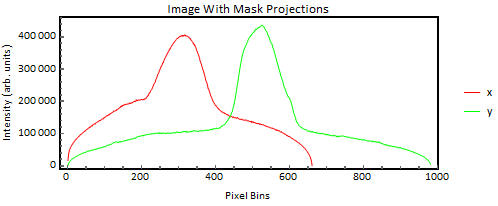
The aim is to subtract the shaded areas in Figure 10 from the image projections leaving just the beam signal:

Figure : Projections of the image with the mask applied, the shaded areas at the bottom of the peaks show what is removed using the two background subtraction methods.

Scaled Mask Subtraction

Subtract a multiple (λ) of the mask projections Figure 5 from the image projections (Figure 8) this allows us to subtract a given value from the intensity of each pixel bin.

Subtract several multiples of the mask projections from the image projections and plot the mean intensity value of the image vs the multiple λ, Figure 11 shows a plot of mean intensity value vs. λ:

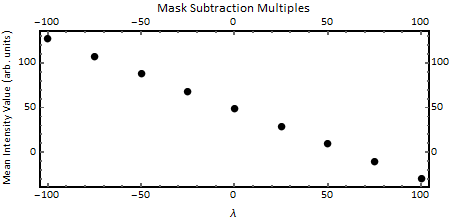


Figure : Plot of the mean intensity value of the image vs multiple of mask subtracted. Extracting the value of λ that gives a mean intensity of zero from a linear fit gives us the scaling factor λ0.

Fit linearly to Figure 11 and extract the value of λ that gives a mean intensity value of 0, use this as the scaling factor (λ0).

Subtract λ0 multiples of the mask data from the image data as shown in Figure 12:

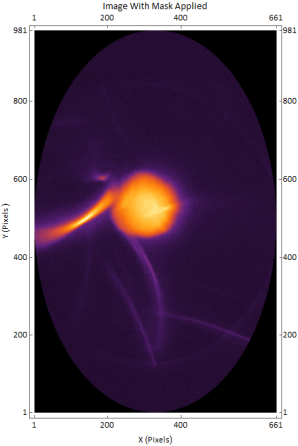
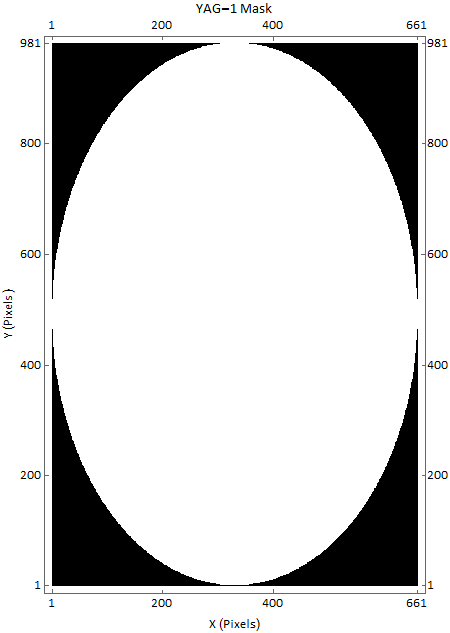


Figure : Illustration of the Scaled Mask background subtraction method. Subtract λ0 times the mask image (Figure 4) from the beam signal image (Figure 8.).

- λ0



=

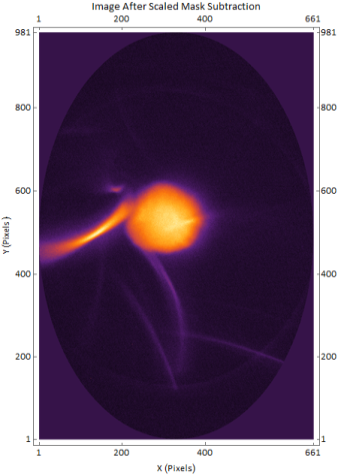
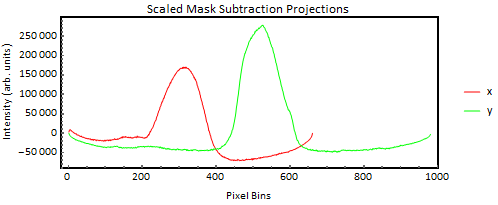


Figure 13 and Figure 14 show the image and projections after the Scaled Mask Subtraction.



Figure : Resulting image from the Scaled Mask subtraction, note how the background is now much darker than before the background subtraction (Figure 7).

Figure : X and Y projections of the example image after the Scaled Mask subtraction method. The edges of the projections no longer curve downwards as in Figure 8.. The upward curve of the edges is a result of the Scaled Mask subtraction overestimating the background level at the edges.



N Point Scaling

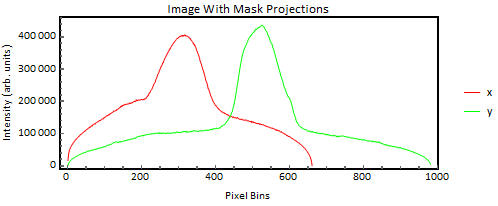
Assume that the first and last N points in the x and y directions of the image projections are just background and not beam, dark current or other artefacts. Through testing several values of N we chose N = 10 for this procedure as this gave enough points of background without picking up any beam or dark current. Figure 15 shows where these points are in the projections:

Figure : Illustration of the first and last ten points of the projections used in the N Point Scaling method, the red (x) and green (y) double headed arrows give a rough idea of where the N points are in the projections.

First 10 X Points

Last 10 Y Points

Last 10 X Points

First 10 Y Points

Subtract several multiples (λ) of the mask projection points from these four sets of points - the same as in the Scaled Mask Subtraction.

|  |  |  |
| --- | --- | --- |
|  | **Minimum rms Intensity** | **λ** |
| **Units** | Arbitrary Units | Dimensionless |
| **First 10 X Points** |  |  |
| **Last 10 X Points** |  |  |
| **First 10 Y Points** |  |  |
| **Last 10 Y Points** |  |  |

For each of the four sets take the value of λ that gives the minimum rms pixel intensity value of the ten points (i.e. for the first 10 x points, take the λ that gives the lowest rms of those first 10 x points). Table 1 shows the minimum rms and corresponding λ for each of the four sets of points.

Table 1: Minimum rms values and the corresponding λ value for each of the four sets of points in the N Point Scaling background subtraction of the example image.

Then choose the set of points that has the lowest rms pixel intensity and use the corresponding λ as the scaling factor (λ0) - in the example the first 10 y points have the lowest rms intensity so λ0 = 146.96 - and subtract λ0 times the entire mask from the entire image as in the Scaled Mask Subtraction, see Figure 12.

Figure 16 and Figure 17 show the resulting image and its projections:

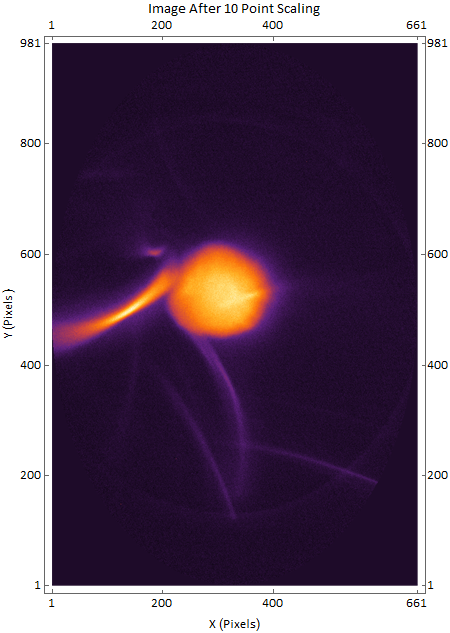
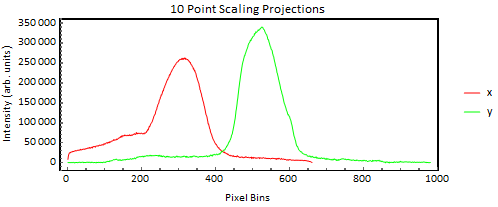


Figure : Resultant image from the N Point Scaling background subtraction method, again note that the background is much darker compared to the beam signal image (Figure 7).



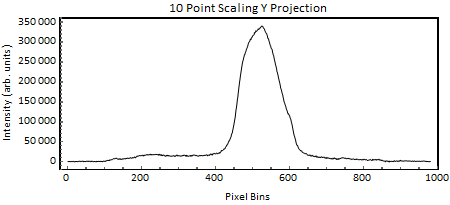
**Figure 17: X (red) and Y (green) projections of the example image after the 10 point scaling.**

Comparison of Scaled Mask Subtraction & N Point Scaling

By looking at the projections of the images after both the Scaled Mask Subtraction and the N Point Scaling we can see that while both methods give a good estimate for the background, the N Point Scaling method gives the better estimate of the two.

This can be seen by looking at the off peak sections of the projections, they are much flatter with the N Point Scaling than Scaled Mask Subtraction, which shows curving towards the edges.

To compare the two methods automatically, we can take the variance of the off peak sections of the projections. To do this, remove the peak data from the projections by finding the maximum value and remove all pixels with intensity greater than the maximum divided by 4 from the projection; do this for both the x and y projections. Figure 18 shows an illustration of this.



Maximum Value

Divided by 4

Figure 17: Illustration of how to remove the peak data from the projections using the 10 Point Scaling y projection as an example.

Removed Points

Maximum Value

Of Projection

This leaves roughly the flat parts of the projection; calculate the variance of these flat parts of both the x and y projections (as shown in Figure 19) and add them together in quadrature; do this for both the N Point and Scaled Mask methods.

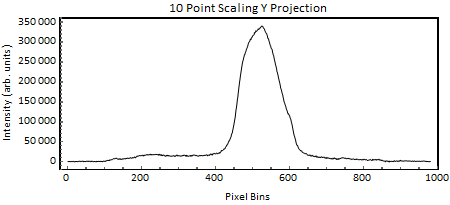


Figure : Illustration of the calculation of the variance of the off peak parts of the y projection for the 10 Point Scaling and Scaled Mask Subtraction images.

Variance

The background subtraction method that gives the lowest total quadrature variance in both x and y will be the one that has the best estimation of the background and the image and projections from this method are the ones that are used for the rest of the procedure.

Variance

Table 2 shows the results of the variance calculation:

|  |  |  |
| --- | --- | --- |
|  | **Scaled Mask Subtraction** | **10 Point Scaling** |
| **Units** | Pixels2 | Pixels2 |
| **X-Projection Variance** |  |  |
| **Y-Projection Variance** |  |  |
| **Total Quadrature Variance** |  |  |

Table 2: The variances of the off peak sections of the x and y projections of the scaled mask subtraction and N point scaling background subtraction methods.

In the case of the example image, the N Point Scaling has the lowest total variance so the image and projections from this method are used in the rest of the procedure.

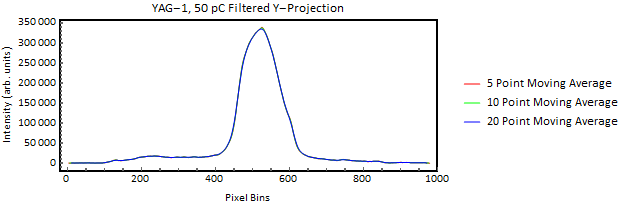
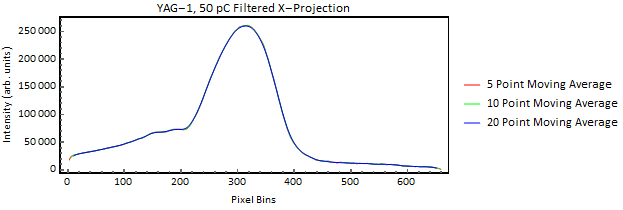
An image that has had a background image subtracted is still run through the estimating a background step of the procedure, doing this does not alter the image as the background level will already be low.

This report will use the worked example without a background image being subtracted and the background being estimated, to give a better idea of how the procedure works for ‘on the fly’ image analysis where a background image is unlikely to be taken.

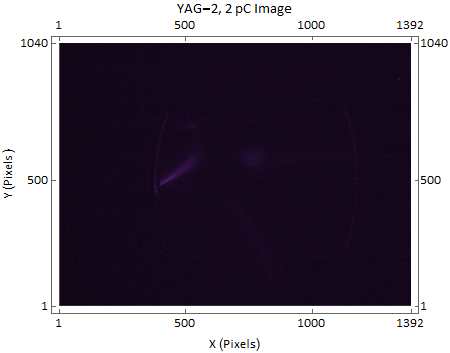
**2.2 Applying Filters**

To further reduce noise the image projections can be made smoother by applying filters. Apply 5, 10 and 20 point moving averages to the x and y projections as shown for the example images in Figure 20 :

Figure : X (top) and Y (bottom) projections of the example image after the 5, 10 and 20 point moving average filters have been applied. As these projections have very little noise all of the filtered projections appear exactly the same and are hidden under the blue 20 point moving average line.



The projections of the example image have very little noise as the beam is at relatively high charge, however for a lower charge beam such as the 2 pC image taken at YAG-2 shown in Figure 21 , the projections are much noisier as shown in Figure 22:

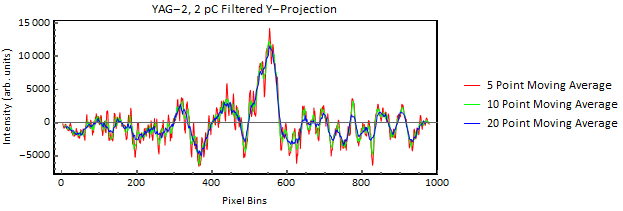
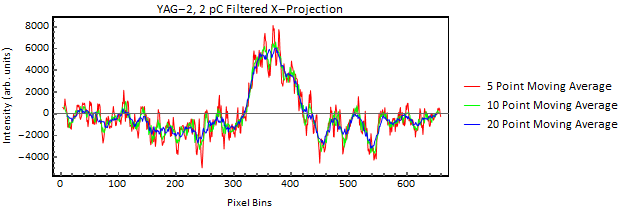


Dark Current

Figure : 2 pC image taken on YAG-2, this is an example of a low charge image with much noisier projections than the example image.

Beam Signal

Figure : X (top) and Y (bottom) projections of the 2 pC image taken on YAG-2, the lower charge results in much noisier projections and it is easier to see the effects of the moving point averaging.



From Figure 22, it can be seen that for lower charge images the filtering smoothens the projections considerably; the 20 point moving average filters (blue) are a lot smoother than the 5 point moving average filters (red). The smoother curves are much easier to fit with the 1D Normal Distributions.

Which filter to use is decided in step 3.

1. **Cutting The Image Down To Just The Beam Signal**

The next step of pre-processing is to further cut the image down to just the beam signal; this is done by fitting 1D Normal distributions to the image projections, extracting the values of μ0 (the mean position) and σ (the standard deviation) and using these to cut the image in the x and y directions.

3.1. Fit 1D Normal Distributions To The Image Projections

Fit the 1D Normal distribution equation ( 6 ), illustrated in Figure 3, to the x and y projections of the image:

To fit a 1D Normal distribution accurately and quickly to a set of data, values of the fitting parameters μ0, σ0, B0 and A0 must be estimated for both the x and y projections before fitting.

3.1.1. Calculating Estimates For the 1D Normal Fits

The parameters B0 and A0 are simple to estimate:

An estimate for B0 is found by taking the mean intensity of the projection.

An estimate for A0 is found by taking the maximum of the intensity values.

Values shown in Table 3:

|  |  |  |
| --- | --- | --- |
|  | **B0** | **A0** |
| **Units** | Arbitrary Units | Arbitrary Units |
| **X-Projection** |  |  |
| **Y-Projection** |  |  |

Table 3: Estimates for the B and A of the a and y image projections.

Two methods of estimating μ0 and σ are used in this procedure:

* Full Width Half Maximum Method
* Moments Method

Full Width Half Maximum (FWHM) Method

Firstly, find the position in the projection of the maximum intensity value, this is assumed to be the mean position μ0. This works on the assumption that there is no dark current or beam artefact more intense than the beam signal.

Then divide the maximum intensity value by 2 and iterate along the projections in both the positive and negative directions until this value is found. The sum of pixels iterated over in both the positive and negative directions will be the width at half maximum σ.

This method is very similar to calculating the variance of the peaks in section 2.1, however instead of removing the points from the projection they are counted to give an estimate of the width.

Table 4 shows the results of the Full Width Half Maximum Method for the example image projections.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **μx** | **σx** | **μy** | **σy** |
| **Units** | Pixels | Pixels | Pixels | Pixels |
| **FWHM Estimates** |  |  |  |  |

Table 4: Estimates of μ0 and σ for the x and y projections of the example image using the Full Width Half Maximum method, μx and σx are the estimate for the x projection and μy and σy are estimates for the y projection.

Moments Method

Calculate the first and second moments of the projection distributions to give estimates of μ0 and σ.

μ0 in this case is the expectation value of the distribution, this gives a ‘balance point’ where 50% of the distribution has a higher value than μ0 and 50% has a lower value.

The second moment – the variance – of the data is calculated using Equation ( 5 ). Square rooting the variance gives an estimate of the standard deviation σ.

In order to calculate the moments, all intensity values in the projections must be positive. Estimating the background in step 2.2 can cause the projections to have negative intensity values. Therefore to compensate for this, if the minimum value of the projection is less than zero, then the minimum is added on to all values in the projection to shift the data above the zero line.

Table 5 shows the results of the Moments method for the example image projections.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **μx** | **σx** | **μy** | **σy** |
| **Units** | Pixels | Pixels | Pixels | Pixels |
| **Moments Estimates** |  |  |  |  |

Table 5: Estimates of μ0 and σ for the x and y projections of the example image using the Moments method, μx and σx are the estimate for the x projection and μy and σy are estimates for the y projection.

The Moments method of estimating the fitting parameters is extremely fast, taking only a few microseconds to calculate for a typical image, however it is very susceptible to noise; also the entire distribution must be captured in the image, this is not the case for the FWHM method.

From analysing several images we found that fits using the FWHM estimates are more accurate than fits using the Moments estimates. Therefore the FWHM method is used to calculate the estimates for fitting the 1D Normal Distributions to the projections.

However, both the FWHM and Moments methods are used to decide which of the filters applied in step 2.2 to use, this is explained in step 3.1.2.

3.1.2. Choosing The Best Filter For The Projections

To decide which of the 5, 10 or 20 point moving average filters to use we compare the results of the FWHM and Moments estimates for each of the filtered projections.

The reasoning behind this is that the projections that have the most agreement between the FWHM and Moments estimates will be the closest representation of a Normal distribution and will therefore be the easiest to fit a 1D Normal to.

To compare both sets of estimates take the proportional difference between the values of μ0 from the FWHM and Moments methods and add on the proportional difference between the values of σ from the FWHM and Moments method for each of the filters on each projection. This gives a value ω that takes into account the difference between μ0 and σ that can be used to compare the filters:

|  |  |  |
| --- | --- | --- |
|  | , | ( 8 ) |

where and are the estimates from the FWHM method and and are the estimates from the Moments method for one of the filtered projections and ω is the comparison factor for that projection.

Table 6 shows the values of for the projections of the example image.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **- No Filter** | **- 5 Point Moving Average Filter** | **- 10 Point Moving Average Filter** | **- 20 Point Moving Average Filter** |
| **Units** | Dimensionless | | | |
| **X-Projection** |  |  |  |  |
| **Y-Projection** |  |  |  |  |

Table 6: The values of for each of the filtered x and y projections for the example image.

Use the filters that have the smallest values of , from Table 6 the x-projection will have no filter and the y projection will have a 5 point moving average filter applied.

Results of the 1D Normal Fits

Fit the 1D Normal distribution ( 3 ) to the chosen filtered projections using the Full Width Half Maximum estimates.

The fits of the example image are shown in Figure 24:

Figure : Plots of the 1D Normal fits for the x projection (top) and y projection (bottom) of the example image. The thick black lines are the projections themselves and the thinner red lines are the fits.

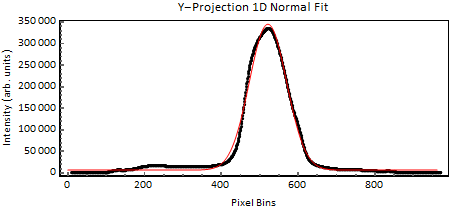
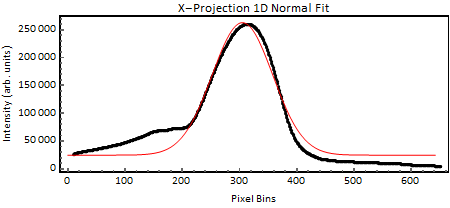


Table 7 shows the fitting parameters from the 1D Normal fits of the image projections

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **B** | **A** | **μ0** | **σ** |
| **Units** | Arbitrary Units | Arbitrary Units | Pixels | Pixels |
| **Cut X-Projection** |  |  |  |  |
| **Cut Y-Projection** |  |  |  |  |

Table 7: Fitting parameters from the 1D Normal fits to the image projections.

3.2 Cut The Image Based On The 1D Normal Fit Parameters

Cut the image data at 3σ either side of the μ0 in the x and y directions, use the μ0 and σ from the 1D fits of the projections.

To account for any poor fits in step 3.1 (i.e. fits that get nowhere near the correct μ0 and σ0) use the Coefficient of Determination R2.

The coefficient of determination gives the proportion of the variance in the dependent variable that is predictable from the independent variable. It lies between 0 and 1, with 1 meaning predictable without error and 0 meaning not predictable. [R]

The minimum value of R2 for a fit that can be used is set at 0.4, if R2 is less than this for the x or y fits then the FWHM parameters estimates of μ0 and σ are used to cut the image instead. Table 8 shows the value of R2 for the example image projections:

|  |  |
| --- | --- |
|  | **R2** |
| **Units** | Pixels2 |
| **X-Projection** | 0.97 |
| **Y-Projection** | 0.99 |

Table 8: The values of R2 for the 1D fits of the example image projections.

The example image has very high R2 in both the x and y so is cut using the fit parameters. Figure 25 shows the uncut and cut example images:

Figure : The uncut (left) and cut (right) example images.

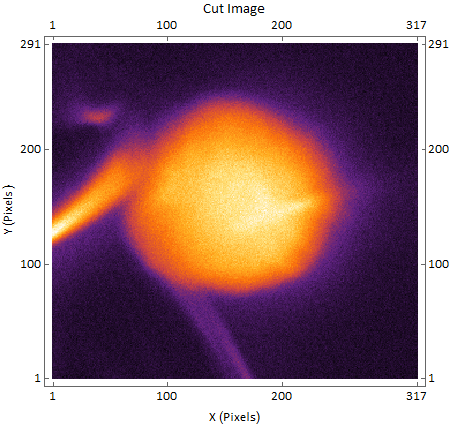
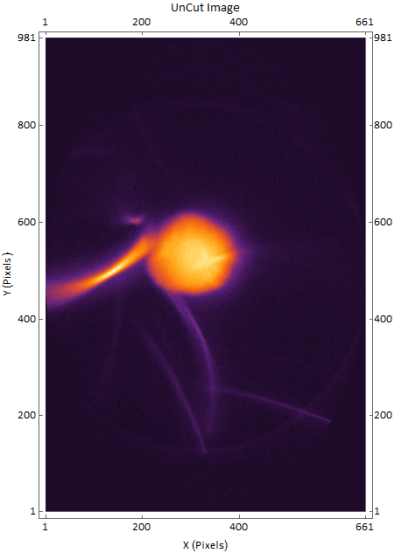
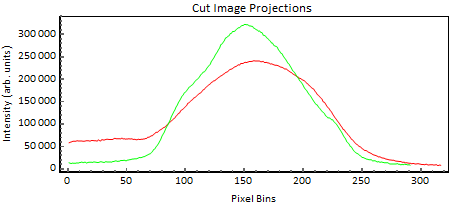


Figure 26 shows the projections of the cut image:

Figure : X (red) and Y (green) projections of the cut example image.



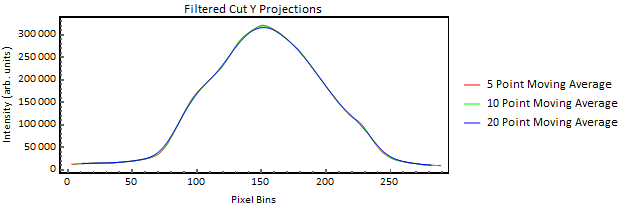
3.3 Fit 1D Normal To The Cut Image Projections

After the image is cut repeat steps 2.2 -3.1:

* Apply Filters
* Get 1D Fit Parameter Estimates
* Choose Best Filter
* Fit 1D Normal Distributions Projections

Figures 27 and 28 and Tables 9, 10, 11 and 12 show the results of each of these steps for the cut image:

Figure : The filtered x (top) and y (bottom) projections of the cut image, again as we are at a relatively high charge of 50 pC the projections have very little noise so there is little difference between the filters.



|  |  |  |
| --- | --- | --- |
|  | **B** | **A** |
| **Units** | Arbitrary Units | Arbitrary Units |
| **X-Projection** |  |  |
| **Y-Projection** |  |  |

Table 9: B and A estimates for the cut image projections, note that the value of B is much higher than it appears on Figure 28, this is because the cut image contains mostly the peak due to the beam signal and very little actual background, this does not matter much as it is only an estimate and the actual fit should get an closer to the actual background level.

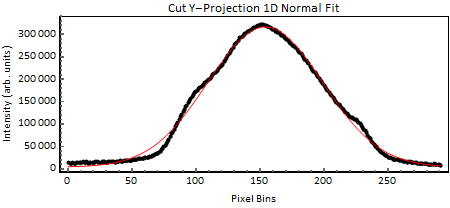
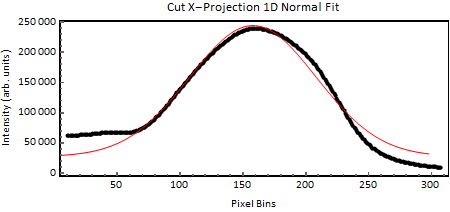
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **μx** | **σx** | **μy** | **σy** |
| **Units** | Pixels | Pixels | Pixels | Pixels |
| **FWHM** |  |  |  |  |
| **Moments** |  |  |  |  |

Table 10: Estimates of μ0 and σ for the x and y projections of the cut example image using the Full Width Half Maximum and Moments methods, μx and σx are the estimate for the x projection and μy and σy are estimates for the y projection.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **- No Filter** | **- 5 Point Moving Average Filter** | **- 10 Point Moving Average Filter** | **- 20 Point Moving Average Filter** |
| **Units** | Dimensionless | | | |
| **Cut X-Projection** |  |  |  |  |
| **Cut Y-Projection** |  |  |  |  |

Table 11 : The values of for each of the filtered x and y projections for the cut example image. The 10 Point moving Average was chosen for the x projection and no filter was chosen for the y projection.

Figure : Plots of the 1D Normal fits for the x projection (top) and y projection (bottom) of the cut example image. The thick black lines are the projections themselves and the thinner red lines are the fits.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **B** | **A** | **μ0** | **σ** |
| **Units** | Arbitrary Units | Arbitrary Units | Pixels | Pixels |
| **Cut X-Projection** |  |  |  |  |
| **Cut Y-Projection** |  |  |  |  |

Table 12: Fitting parameters from the 1D Normal fits to the cut image projections.

After cropping the image down to just the beam signal we can process the image to calculate values of **.**

1. **BVN Method**

The BVN method involves least squares fitting of equation ( 3 ) to the image data and extracting values of **.**

The data is converted into 3D coordinates based on the intensity of each pixel and the fit parameters from the 1D Normal fit of the cut image projections are used as estimates.

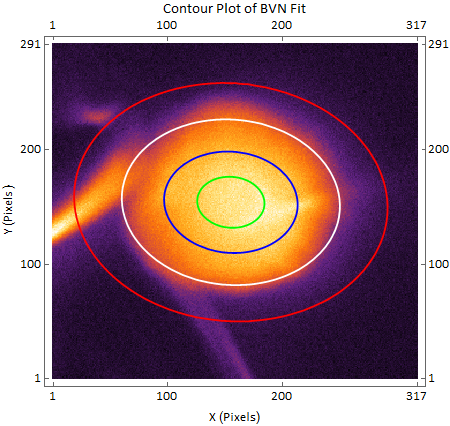
In the case of saturated images, the distribution will plateau at the saturation level, thus making fitting the BVN difficult. In these cases the saturated values are removed from the image data.

Table 13 shows the values taken from the BVN fit of the cut example image.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **μx** | **μy** | **σxx** | **σyy** | **σxy** |
| **Units** | Pixels | Pixels | Pixels2 | Pixels2 | Pixels2 |
| **BVN** |  |  |  |  |  |

Table 13: The values taken from the BVN fit of the cut example image, μx and μy are the values of the centroid position and σxx, σyy, and σxy are the elements of the covariance matrix.

Figure 29 shows a contour plot of the BVN fit of the cut example image:



**Figure 29: Contour plot of the BVN fit of the cut image data, green = 90% peak, blue = 66% peak, white = 33% peak and red = 10% peak.**

1. **MLE Method**

The second method to calculate values of is to directly calculate the 1st and 2nd moments of the 2D cut image data - which are the maximum likelihood estimators of of a normal distribution - using equations ( 4 ) and ( 5 ). The algorithm used in the prototype Mathematica notebook is given in the appendix.

This method is much faster than fitting the BVN and doesn’t require any estimates. However, it is very sensitive to noise and high levels of background, therefore will only work if the image contains only beam signal, which is why it is performed after the cutting of the image.

Table 14 shows the results of directly calculating the moments of the 2D cut image data:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **μx** | **μy** | **σxx** | **σyy** | **σxy** |
| **Units** | Pixels | Pixels | Pixels2 | Pixels2 | Pixels2 |
| **MLE** |  |  |  |  |  |

Table 14: The values from the MLE method, μx and μy are the values of the centroid position and σxx, σyy, and σxy are the elements of the covariance matrix.

1. **Rescale The Centroid Position**

We must correct the values of ***μ*** from the BVN and MLE methodsto account for cropping the image to just the beam signal.

To correct for the cutting in step 3 add on the number of pixels cut from the image in the x and y directions to the values of μx and μy.

To correct for the mask applied in step 1 add on the difference between the radius of the ellipse and the distance from the centre of the image to the origin in both the x and y directions, as shown in Figure 30:

Figure : The example image with the YAG-1 screen mask overlaid (red), the offsets of *μ* due to applying the mask are shown in light blue.

Y-Offset

X-Offset

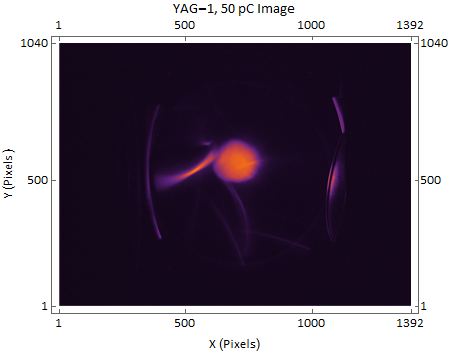


Figure 31 shows the contour plot of the BVN fit with corrected ***μ*** over the original image:

Figure : Contour plot of the BVN fit over the original image, green = 90% peak, blue = 66% peak, white = 33% peak and red = 10% peak. The value of *μ* has been corrected to account for cropping the image to just the beam signal.

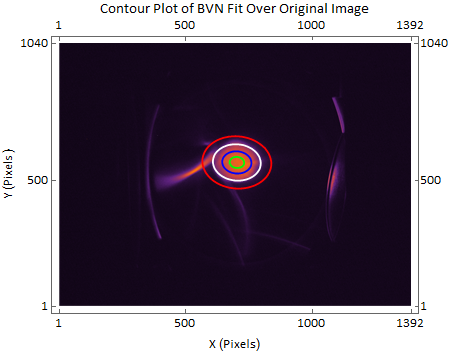


Table 15 shows the values of the BVN fit and moments of the distribution with corrected ***μ***.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **μx** | **μy** | **σxx** | **σyy** | **σxy** |
| **Units** | Pixels | Pixels | Pixels2 | Pixels2 | Pixels2 |
| **BVN** |  |  |  |  |  |
| **MLE** |  |  |  |  |  |

Table : Values of the BVN fit and MLE of method with the corrected *μ*.

1. **Converting To Millimetres**

The values of from the BVN fit and directly calculating the moments are given in pixels; convert to millimetres by multiplying by the pixel to mm conversion factor for each screen.

The conversion factors are calculated using three different methods:

* Calibration A
  + Measured distance from the determined far edge to the near edge of the YAG in pixels divided by the YAG diameter in mm.
* Calibration B
  + Measured diameter of the reflected beam pipe image in pixels, divided by the engineering pipe diameters in mm:
    - For 50mm YAGs internal pipe diameter is 34mm
    - For 100mm YAGs internal pipe diameter is 98mm
  + Assuming perfect vacuum pipes
* Calibration C
  + For 50mm YAGs, distance between lowest point on top YAG clamp, and highest point on bottom YAG clamp is 25mm. Average distance using both near and far points (to give calibration in the middle of the YAG) is taken in pixels and divided by 25mm.
  + For 100mm YAGs, horizontal distance from far edge of clamp to near edge of clamp is 103mm (Equivalent to Calibration A)
  + Similarly for BA2 YAG-02, except horizontal distance is 48mm.

An average of the values from the three methods is taken to get the conversion factor for each screen.

Table 16 shows the conversion factors for each of the three calibrations and the average for YAGs 1, 2 and 3.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Calibration A** | **Calibration B** | **Calibration C** | **Average** |
| **Units** | Pixels/mm | Pixels/mm | Pixels/mm | Pixels/mm |
| **YAG-1** | 21.1 | 21.2 | 22 | 21.43 |
| **YAG-2** | 21.7 | 22.6 | 23.2 | 22.5 |
| **YAG-3** | 21.9 | 22.4 | 22.6 | 22.3 |

Table : The pixel/mm conversion factors from each of the three calibrations and their average for YAGs 1, 2 and 3.

Divide the values from the BVN fit and the MLE method by the conversion factor, the values in mm for the example image are shown in Table 17 :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **μx** | **μy** | **σxx** | **σyy** | **σxy** |
| **Units** | mm | mm | mm2 | mm2 | mm2 |
| **BVN** |  |  |  |  |  |
| **MLE** |  |  |  |  |  |

Table : Values of the BVN fit and moments of the distribution in millimetres.

1. **Centroids Relative To Machine Axis**

This procedure calculates the centroids of the images (μx andμy ) relative to the bottom left corner of each image. As images are taken on different screens using different cameras this is not consistent. Ideally the centroid values would be relative to the point where the machine axis passes through each screen.

As the screens may not be aligned perfectly (i.e. the centre of each screen may not lie on the machine axis) then it would be useful to send the HeNe laser along the axis of the beam line and see where the laser spot appears on each of the screens. Images of the laser on each screen can then be subjected to the analysis described in the report and the centre of each screen determined; this is a future goal.

As part of the Model Validation Project [1] this was done using the electron beam itself – centred on the cathode - instead of the HeNe laser.

**References**

[1] – M.S. Toplis et al. “Comparison of Model vs. Reality for VELA” - TUPOW028, presented at IPAC’16, Busan, South Korea, 2016

[2] – ‘Method of Estimating Beam Size from VELA Screen Images.nb’

**Appendix I – MLE Algorithm**

The MLE method calculates the moments of the distribution using the following algorithm:

* Define the image data array to be **V.**
* Find the number of rows and columns of **V**, defined as **Ny** and **Nx** respectively.
* Normalise **V** by dividing each element by the sum of the elements of **V**. This gives the normalised array **V’.**
* Create a **Nx** x **Ny** array called **yD**. Each row consists of **Nx** copies of the row index, for example row 1 would be {1,1,1,1...**Nx**}, row 2 would be {2,2,2,2…**Nx**}, row 3 would be {3,3,3,3…**Nx**} etc.
* Create a **Ny** x**Nx** array called **xD.** This is the transpose of **yD**, for example each row in this array is a list of the range up to **Nx** (i.e. {1,2,3,4….**Nx**}).

Therefore we have 3 arrays of equal size: **V’**, **xD** and **yD**. The mean vector and covariance matrix of the data are then calculated as follows:

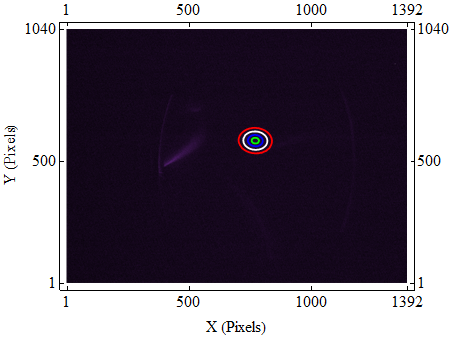
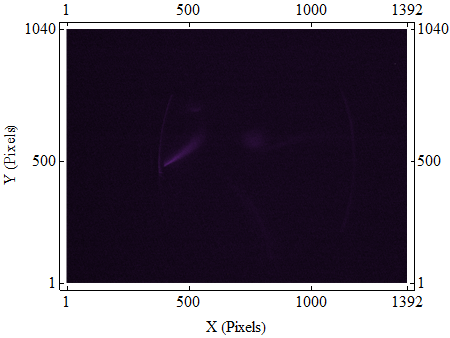
**Appendix 2 – Interesting Images**

As the aim of this procedure is to robustly analyse as many images as possible we have tried to test the analysis on types of images that we thought would be especially challenging for an automated analysis procedure.

Below are several examples of these challenging images, together with the BVN fit contour plots to show how well this procedure has performed in testing.

Low Charge

Figure : Image taken on YAG-02 at 2 pC (left), this is an example of a very faint image; the analysis procedure is still able to accurately fit a contour plot as can be seen in the right hand image.

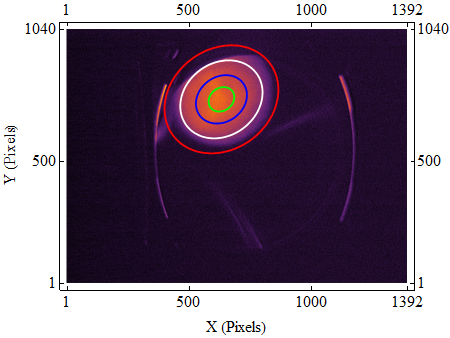
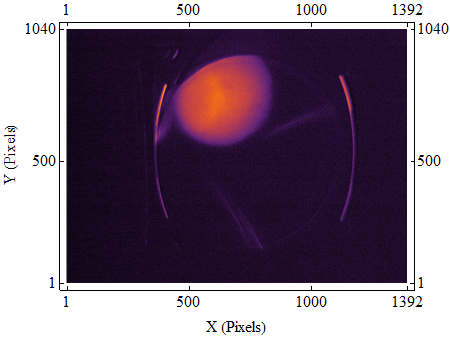


|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **μx** | **μy** | **σxx** | **σyy** | **σxy** |
| **Units** | mm | mm | mm2 | mm2 | mm2 |
| **BVN** |  |  |  |  |  |
| **MLE** |  |  |  |  |  |

Table 18: Centroid and covariance matrix values for an example low charge image taken on YAG-02 at 2 pC.

Cut-Off Beam

Figure : Image taken on YAG-03 at 100 pC (left), this is an example of an image where the beam has been cut off by the edge of the screen; again the analysis method still fits accurately as can be seen by the BVN contour plot in the right hand image.

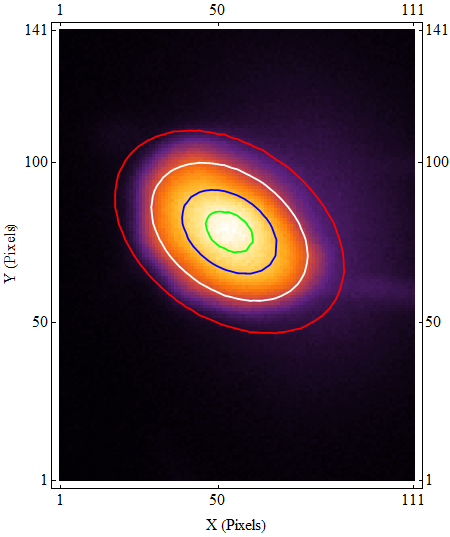
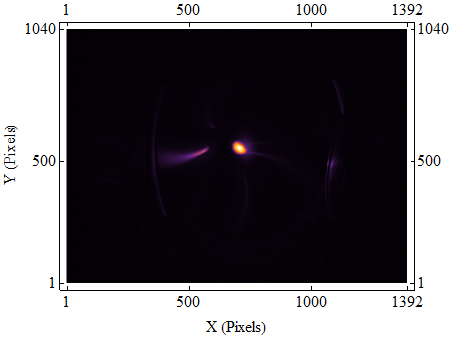


|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **μx** | **μy** | **σxx** | **σyy** | **σxy** |
| **Units** | mm | mm | mm2 | mm2 | mm2 |
| **BVN** |  |  |  |  |  |
| **MLE** |  |  |  |  |  |

Table : Centroid and covariance matrix values for an example image where the beam has been cut off by the screen (YAG-03, 100 pC).

Tilted Beam

Figure : Image taken on YAG-01 at 40 pC (left), this is an example of a tilted elliptical beam; the orientation of the beam is still picked up by the analysis procedure and the contour plots from the BVN fit this tilted shape (right).



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **μx** | **μy** | **σxx** | **σyy** | **σxy** |
| **Units** | mm | mm | mm2 | mm2 | mm2 |
| **BVN** |  |  |  |  |  |
| **MLE** |  |  |  |  |  |

Table : Centroid and covariance matrix values for an example image with a tilted elliptical beam (YAG-01, 40 pC). Note that the values of σxy are much higher as a proportion of σxx andσyy for this image than the previous images, this is due to the tilt of the beam.