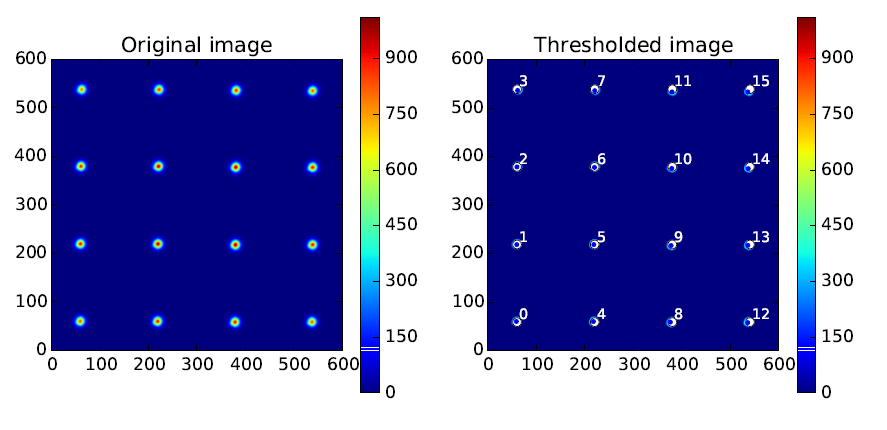
Analysis of Lynx data

Lynx data files are 18 DICOM files (for 18 beam energies) each with 16 spots delivered to known intended coordinates. Current software analyzes the spots to compare them to their intended coordinate. New software should focus more on detecting systematic errors.



Software should analyze all 18 energies, but for simplicity this discussion will focus on a single energy.

The first step is to read in the original DICOM data, 600x600 array.

The next step is to find the centroid (x,y) of all 16 spots. Generally, this is tricky unless you know the approximate position of each spot, which we do. Spot 0 is approximately at the maximum pixel value of all pixels with 0<x<100 and 0<y<100. Ideally the centroid is determined by doing a 2-dimensional least-squares fit to the data in that region (or a tighter region) with a Gaussian. The current software does a reasonable job of finding the centroid for all sixteen spots, but there may be better ways.

For convenience, it helps to create a symmetric coordinate system. The raw data has integer pixel coordinates from 0 to 599. The centroids are real numbers in the same range. Subtract 299.5 from each centroid value and they will now be between -299.5 to 299.5.

The image should be symmetric about the origin, so one can determine the offsets, xoff and yoff, by simply averaging the x centroids and the doing the same for the y-centroids. Create corrected centroids by the offsets and report the offsets as an output.

Next, the image should be corrected for possible rotations. To do this, fit a straight line (y=mx+b) each set of offset-corrected centroids of 4 points: (0,4,8,12)->m1; (1,5,9,13)->m2; (2,6,10,14)->m3; (3,7,11,15)->m4. Average m1, m2, m3 and m4 to find the best slope m. The rotation theta is the inverse tangent of m. Report the value theta. (We don’t need the intercepts b).

To get centroids that have been corrected for rotation, the original x is related to the rotated centroid x’ by x=x’cos(theta). So the new centroid is x’=x/cos(theta).

For y, the equation is y-y’=sin(theta) so y’=y-sin(theta).

After applying first the offsets, and then the rotational corrections we have 16 centroids (x’,y’). The next step is to look for any distortions. For each point divide each of it’s coordinates by it’s expected coordinate.

Look at this ratio for the corners (pts. 0, 3, 12 and 15). The average value would be the overall scale factor, which should be reported.

Look at that ratio for the other edge points. For points 1, 2, 13 and 14 the average x ratio determines “x pin cushioning” if it is different from the overall scale. This average should be reported. For points 4, 7, 8 and 11 the average y ratio determines “y pin cushioning” if it differs from the overall scale. This average should be reported too.

The inner points, 5, 6, 9 and 10 should be consistent with the scaling and pin cushioning results above, but will be less sensitive as they are closer to the origin, so we will ignore them.

Finally, after applying the above corrections, we should note any spots that still deviate from their expected position by more than 2mm. Note: pixels are 0.5mm. This is what our old software did, but without correcting for the systematic errors. Generally I don’t believe we have any significant (i.e. >2mm) random errors.

One other note:

Ideally this analysis would be for all 18 energies simultaneously. The expectation is that the same correction should be applied to all 18 images. The initial offset correction, for example could be due to setting up the detector imperfectly. But it is positioned only once and then all 18 beams are delivered. So if it off by 2mm for energy 1, it should still be off by 2mm for energy 18. Similarly if there is a rotational error positioning the detector, it should be the same for all 18 measurements.

For the sake of software development, I suggest getting a program that works for one energy. When that seems satisfactory, it can be modified to loop over all 18 energies and report the results for all 18 energies individually. We can manually examine the output file to see if the results (offsets, rotations, etc.) are consistent across energies. Ultimately, the program could be made more sophisticated to calculate results for each energy and then report inconsistencies. For example, one my find a trend in the pin cushioning that grows larger with increasing beam energy.

Assuming we have a working program that gives the expected results, another project would be to retrospectively analyze our historical data (8 years x 12 months) to see how reliable our equipment is and what tests might be most important to examine in the future. We could right a paper and publish such results.