McMaster University
Department of Materials Science and Engineering

Summer | 2020

EELs Data Processing Documentation

July 18th, 2020

Ray Lyu

Abstract

This research devotes to analysing the EELs data acquired from the composite materials. The data will be analysed in specific energy levels and plot on a 2D color map. To improve the quality of the map and make the features more evident to the reader, interpolation methods are used to expand the data set. This study also tested several methods of interpolation to find the best method and variable that can follow the raw data and also validate the method by making the comparison with the raw data on the major features. Additionally, some adjustments were also made to the silicon dioxide region to correct the antenna effect of the material.

Data Acquisition

In this study, we are using an EELs spectrometer examining the composite material which consists of aluminum, silicon dioxide, and silicon carbide. The spectrometer will collect EELs spectrum from the detection spot which has fifteen spots vertically and seven horizontally. The vertical spot position can be seen in (Fig 1). In the horizontal direction, the seven lines consist of two lines for aluminum and silicon carbide and three lines for silicon dioxide. Thus, there are a total of 7x16=112 spectrum. However, there are eight detection spots that don't have valid EELs data, so there are 104 EELs data acquired. The EELs data collected from -0.26 eV to 0.63 eV with corresponding Intensity.

The intensity corresponds to the energy loss plot (Fig 2,3,4,5,6) that is unique for every detection spot, but all the plots share some common characteristics. The most obvious one is the two giant peaks near 0 energy Loss and actually there is supposed to have another peak right at 0, which is called the zero loss peak. However, to make it easier for analysing, that peak has been deleted for all detection spots. The log plot(Fig3,5) shows more details on the energy loss side (EEL>0), which shows some major peaks appearing on that side, at energy level =0.1/0.133/0.154/0.295/0.44, respectively. Because the data after 0.5eV are a lot weaker compared to the data before it, which makes it easier to be affected by the noise, we are not going to do the analysis of the data after 0.5eV. Additionally, this research only focuses on the energy loss side, so we are not going to analyze the data in the energy gain region (EEL<0).

Data Extraction

The first step is to extract the data from every spectrum. The purpose of this step is to create a 2D map for intensity at a specific energy level. For example, if we want to create a map for energy level at 0.1 eV, the code will go through all the files, extract the intensity for each detection spot at 0.1eV and put it into a table. Due to the discrete nature of data, there is no data point at 0.1eV. Based on that there are two solutions to pick the data, one is to choose the nearest neighbor, which is easier to operate but the data is more likely to be affected by the noise. The other one is to construct a linear line between the two

data points (one before, one after) and find the corresponding intensity by taking the anticipated energy level into the function, which is more complex but the data is more robust compared to the last method. Therefore, the second method is used for extracting the data points from the data bank.

To keep the originality of the raw data, the missing data are treated as the blank in the table. Although the blank can be filed by using interpolation through the line or using the average value of the surrounding, we found that may cause some unexpected features in the map. For example, if the surrounding data shows strong variance or huge drop which may cause an error for the data point filled in. Therefore, we decide to leave the missing data as the blank in the data table.

2D Map Construction

The second step is to turn the data table to a 2D colour map. Based on the available python library, there are mainly two ways of achieving this. One is turning the values in the data directly to its corresponding colour. The method will assign a colour range, which is based on the cmap, to the data range provided. For example in (fig7), the method will assign the maximum number to the brightest end and minimum to the darkest end and those values in between have each unique colour. The other method is to plot the top view of a 3D plot and the colour will be assigned based on the cmap. The color map should provide the position of each pixel, so the first method will be used for the color map construction.

The cmap (Fig8) is short for colour map in python, which has more than fifty choices classified into six groups. The colour map needs to satisfy several requirements, which the map should be clear and easy to tell the difference between different intensity. Thus, jet and plasma are too colourful and Blue, GnBu, viridis have a small colour range which makes it hard to distinguish different intensity. The twilight has the same colour for maximum and minimum which makes it not appropriate for this case. Based on bare eye observation, afmhot seems more clear to the hot plot, so afmhot was chosen for the colour map for the rest of the colour map construction.

To separate the data in different element regions (Fig9). Two blank lines lie between the aluminum and silicon dioxide and silicon dioxide and silicon carbide respectively. Based on that situation, two different types of raw data plot are made, one shows the separation line and the other doesn't (Fig9). In the end we decided to combine all the regions together, so the plots on the right are used for the raw data plot reference.

The 2D color map result can be seen on (Fig 10,11). The first set doesn't have the same scale among the plot, which means the map is plotted based on it's maximum and minimum. The purpose for this set is used to visualize the variance in the plot, such as the location of the peak and the valley. Especially for the two high energy plots (0.295eV and 0.44eV) has very close maximum and minimum, which makes the difference between the different intensity very small and makes it hard to visualize in the scaled plot. The second set has scaled from 0 to 0.06 among all four plots. The purpose for this set is to show the intensity difference between the low energy and high energy, which is quite obvious for the low energy loss has a relatively higher overall intensity compared to the high energy loss. Additionally, both sets

have deleted the row 3050nm, because that line only has two pixels and was used for validation of the accuracy of the data, making it less important for raw data profile.

Interpolation

The third step is to increase data sampling, so the feature of the map will be more evident to the reader.

After cautious selection, RBF (radial basis function) and Lsq_Bspline (Least Square B-spline) is chosen as the interpolation method. The explanation to those methods can be seen in appendix B. In this case, the radial basis function tends to generate an interpolation line that goes through every raw data and B-spline tends to generate a smooth curve that does not necessarily go through the data points.

First, Lsq_Bspline with highest order K= 2/3/4/5/6 are applied to create the interpolation line. In low energy (0.133eV)(Fig12), the k=2 interpolation line shows huge difference with the raw data, the other interpolation lines showed decent work on following the behavior of the data. However, in high energy (0.44eV)(Fig13), because the vibration of the data is greater than the lower energy, the interpolation line generated by the k=3 fails to follow the tendency of the raw data. k=4/5/6 can show the major tendency of the raw data well, but the k=4 line did worse on the local maximum and minimum part compared to the other two. In order to find the best k value between five and six, we plot the interpolation result at 0.154eV (Fig14). The result shows that the k=6 tends to be affected by the variation of the data compared to k=5, which makes the interpolation line more vibrate. However, it is necessary to consider the noise effect to the raw data, which we should avoid because the interpolation line has too much fluctuation. Therefore, k=5 are chosen for the Lsq_Bspline method variable we used for the vertical direction interpolation.

Then, we apply two methods (B Spline with k value equal to 5) to the horizontal direction in both low (0.133eV) and high(0.44eV) energy. In low energy (Fig17), the result generated by two methods are close, but the RBF method has more fluctuations in the 250nm-750nm region. In high energy(Fig 18), the difference between RBF and B splines is obvious. Due to the RBF method that tends to go through every data point, the interpolation line seems more vibrate compared to the B-spline method. Also, in SiCB plot the RBF method created a deep valley around 2750nm, which caused huge variance compared to the raw data. Hence, B spline method with k equal to five will be used for vertical direction interpolation.

For the horizontal direction, B splines with k equal to two and three are first applied to the raw data (Fig19). Due to the limited raw data available in this direction (seven raw data available), the B spline method can only generate the result with k value less or equal to three. In low energy (0.133eV), the k equals two line shows more vibration compared to the the k equals to three line and the k equals to three lines seems more smoother than the k equals to two in the high energy (0.44eV)(Fig20).

Then we compared the B spline with k equal to three with RBF method. Due to limited raw data, both methods generate close results in both low and high energy, which the interpolation line tends to go

through every raw data point. However, the B spline method tends to create a higher / lower local maximum / minimum, which makes it possible to create some unexpected features on the 2D color map. Therefore, RBF seems more promising for the horizontal direction interpolation.

In summary, in the vertical direction, we use Lsq_Bspline with K value, the highest order, equal to five and in the horizontal direction, we use the RBF.

The sample point we expand also needs consideration. Because the purpose of expanding is to increase the details of the plot, and it is possible for some unexpected features in the extensive number sampling. Based on that, we expand the vertical from 14 to 32 and 62, and we found that there is not too much difference between the 32 points plot and 62 points plot, but compared to 14 points plot 32 points indeed improve the quality and details of the plot. Thus, in the vertical direction, we increase the data point from 14 to by using Lsq_Bspline. Then it is for the horizontal direction, which has seven points in the raw data, and 14/21/28 are tried. Based on the plot we acquired, it is noticed that 21 points shows a lot of unexpected features which does not show on the raw data map and the 28 points map created a massive drop in the 3000nm region. Therefore 14 points are chosen for the horizontal direction expansion.

Hence, after the interpolation method expands the data set from 104 points to (32x14) 448 points, the 2D color map expands from 104 pixels to 448 pixels.

The result of interpolation can be seen on (Fig20,21)

Interpolation Validation

During the discussion, we found that the method used may create some expected feature. Thus, it is necessary to compare the interpolation result with raw data. First we need to confirm that the feature appears in the interpolation data should appear in the raw data too. The features include the color difference, the peak and the valley.

0.133eV are picked for the sample to prove the validation of the method (Fig for row 0.133). By observation we can tell the major features of this energy level are. The dipole distribution and relatively higher intensity in the center compared to the two sides. Meanwhile the left, aluminum side, has a lower valley compared to silicon carbide.

The interpolation result (Fig23) which expands the data point in the vertical direction from 14 to 32 and the horizontal direction from 7 to 14. The result indeed shows the features that describe above for the raw data. However, some unexpected features are shown in the interpolation result, which is the color difference between neighboring column and row. The most evident difference appears in the start and end part of the plot and also the first and the last two columns.

To find the problem caused by the start and the end row color difference, we need to go back to the vertical interpolation result (Fig23). We noticed that the interpolation line follows the raw data properly

and there is a drop in the region between 3000nm and 3100nm in the four columns out of total seven columns, which explains why the obvious color difference.

To find the reason for the color difference between neighboring columns, we need to examine the accuracy of the horizontal interpolation result (Fig24). The raw data marked in the plot isn't actual raw data, those are the vertical interpolation results, but the last step has shown the vertical interpolation result is reliable. Thus, we can assume those "raw data" behaves like the actual raw data. The interpolation line indeed shows the tendency of the raw data, so the interpolation result should be robust. The appropriate explanation for the color difference between columns can be explained by the interpolation itself, in which we double the number of data points. However, we shouldn't say the first two lines correspond to the first line in the raw data, because the interpolation creates a data point that is between the first and second raw data column. Thus, the second line can't technically be the same line as the first one. That's the reason why the color difference between neighboring columns.

Data Adjustments

To show the overall intensity difference between low energy level and high energy level, all the plots will be normalized based on the maximum value of the whole data set, which is around 0.059. However, the high energy plot has a relatively low intensity, making the whole plot dark and hard to visualize the intensity variance inside the plot. Thus, a formula (Fig25) is created for finding the proper multiplication factor to multiply with every energy plot (Fig26). Based on the observation, the intensity difference between high energy and low energy are preserved and the variance in the high energy is clearer compared to the color map without multiplying with the multiplication factor.
Based on the study done by Dr. Maureen Joel Lagos , it is noticed that the silicon dioxide region has relatively higher intensity compared to the other two regions, which is called the antenna effect which during the energy loss experiment, silicon dioxide seems to receive some extra intensity. Thus, to remove the extra intensity, a specific value will be subtracted from the raw data. Due to effect only occurring in the near infrared region (<0.2eV), the energy levels that will be investigated are 0.1 / 0.133 /0.154eV.
After taking several tests, we find the proper number for each energy level to subtracts. For 0.1eV, silicon dioxide line A/B/C will be subtracted 0.008eV. For 0.133eV silicon dioxide line A/C will be subtracted 0.01eV, silicon dioxide line B will be subtracted 0.0115eV, silicon carbide A/B will be subtracted 0.004eV. For 0.154eV, silicon dioxide line A/C will be subtracted 0.009eV, silicon dioxide line B will be subtracted 0.0115eV. All the result can be viewed in Appendix (Fig27)

Video

To show the intensity variance through the energy range that we are interested in (0.05eV - 0.5eV). For each frame, a blank area is added to each side of the plot. To make each frame more smooth, the sampling is increasing to 245 in vertical direction and 65 in horizontal direction. To make most frames easy for the viewer to read, the

energy difference of each frame is 0.001 eV, so there are totally 450 frames in each video. To control the length of the video and make it easy to stop at specific energy (0.1/0.133/0.154/0.295/0.44 eV), the frame rate is 15 frames per second.

Conclusion

In summary, the raw data 2D map made in following steps, first the specific intensity of its energy extracted from the data bank, then the data will be put into a table and multiplied with the multiplication factor and normalized the data. Finally, afm hot color map is applied to the table and transforms the table to the 2D color map. The result can be seen in (Fig28)

For interpolation data, by using the data extracted from the data bank, the interpolation method will be applied to the data in horizontal and vertical direction respectively. Then the interpolation data will be put into the table and multiplied with the multiplication factor and use afm color map to transform it to the 2D color map.

For subtraction map, subtraction will be performed during data extraction. The code can recognize where the data comes from and for example when the intensity data come from silicon dioxide in 0.133 eV, 0.115 will be subtracted before going to the table. The result can be seen in (Fig 29)

This study focuses on constructing the 2D color map for the intensity of specific energy of the data acquired from the EELs spectrometer measuring the compost material. By applying proper methods in the python library and choosing the appropriate parameter we can build a 2D color map that shows the feature of the raw data. Additionally, by using the fittest method of interpolation and parameter we can build a map with more detailed color map and similar behavior with the raw data map. Then by doing some adjustment on data, the variance in the plot can be more clear ro th reader and it is necessary to make regional adjustments to remove the antenna effect in the silicon dioxide region. Finally, the pictures of each energy level will be made into a video to show the intensity changes through a certain energy range.

Appendix A

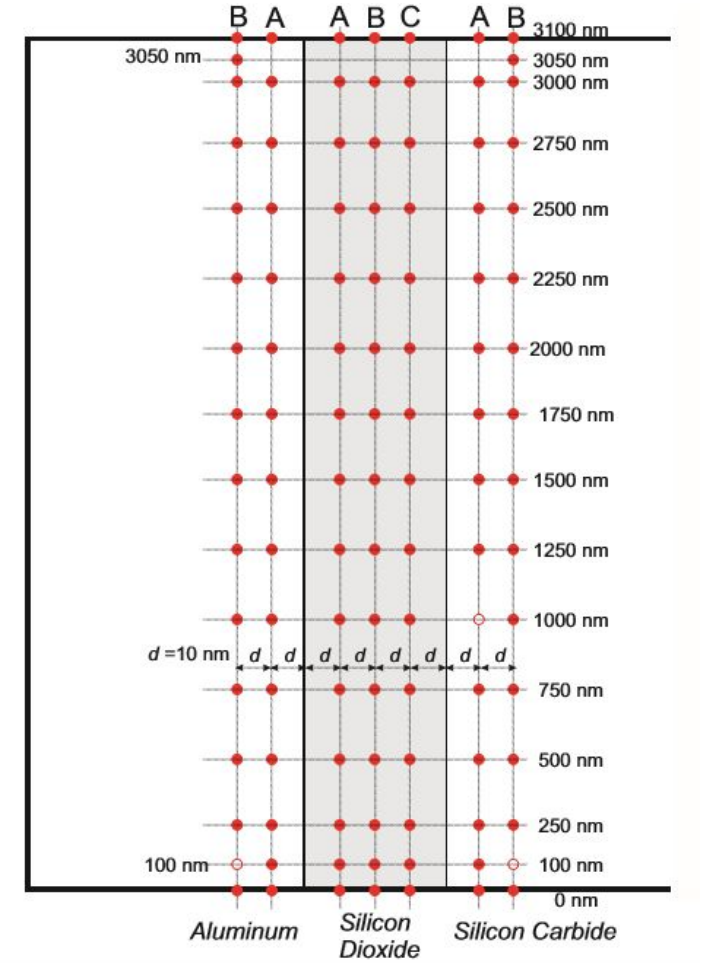

Fig1: The detection spot

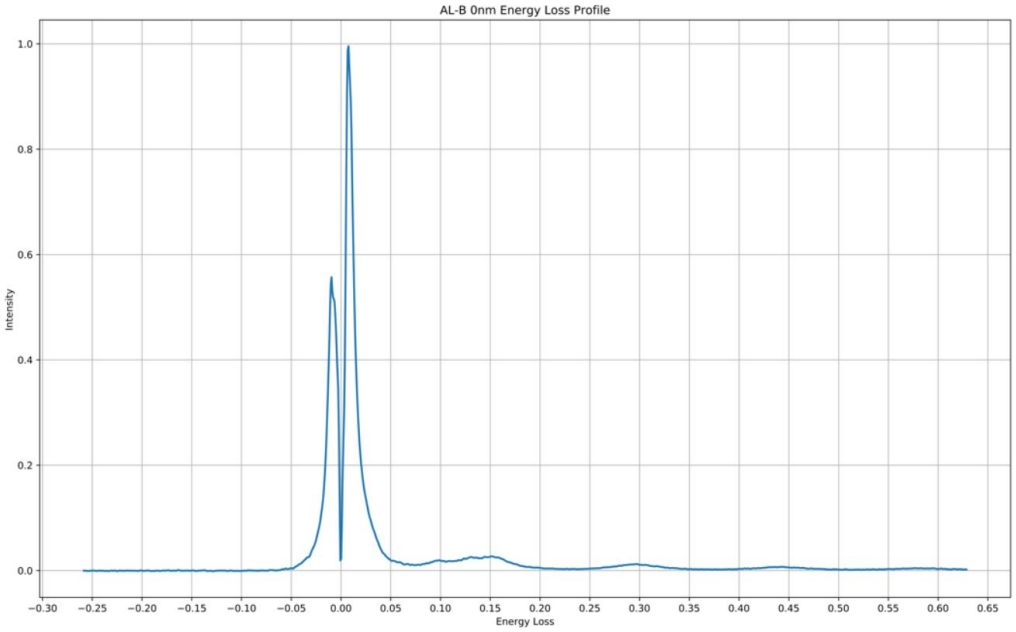

Fig2 Example (Al-B 0nm) energy loss profile

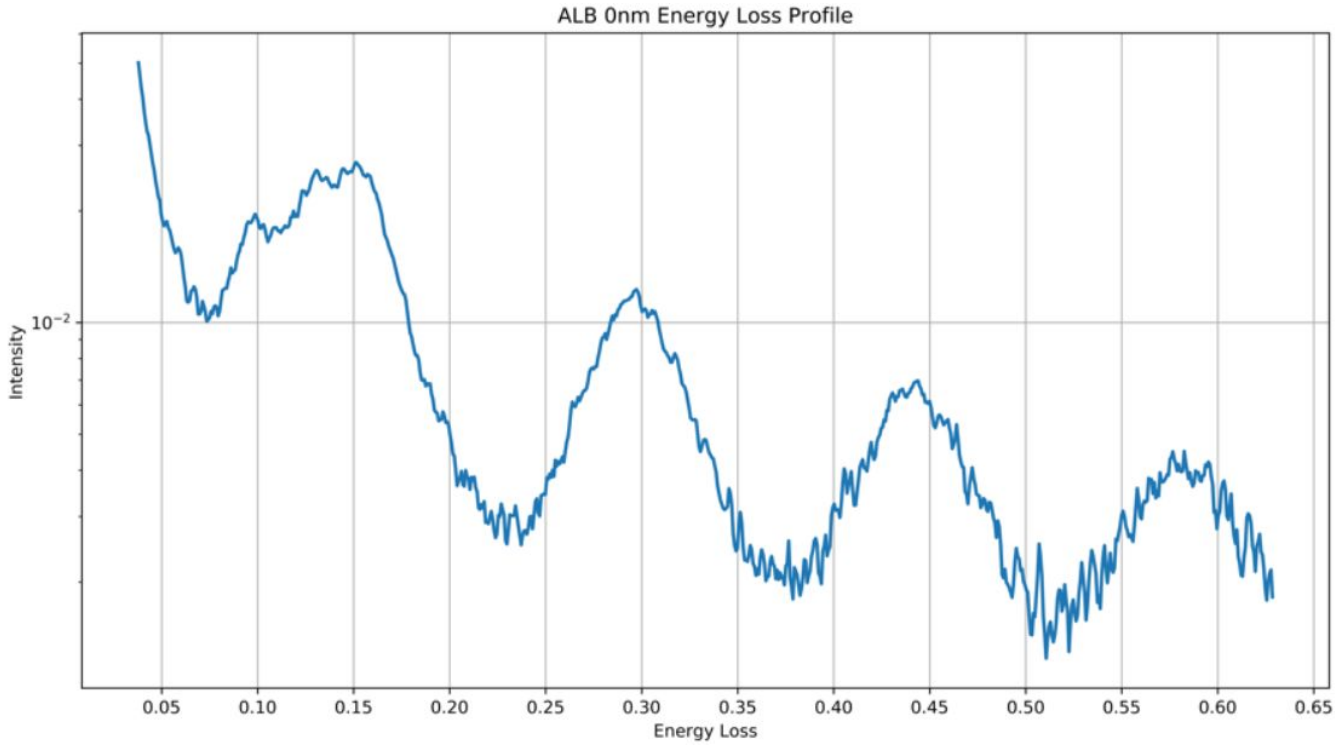

Fig3 Example (Al-B 0nm) energy loss profile in log scale in energy loss side

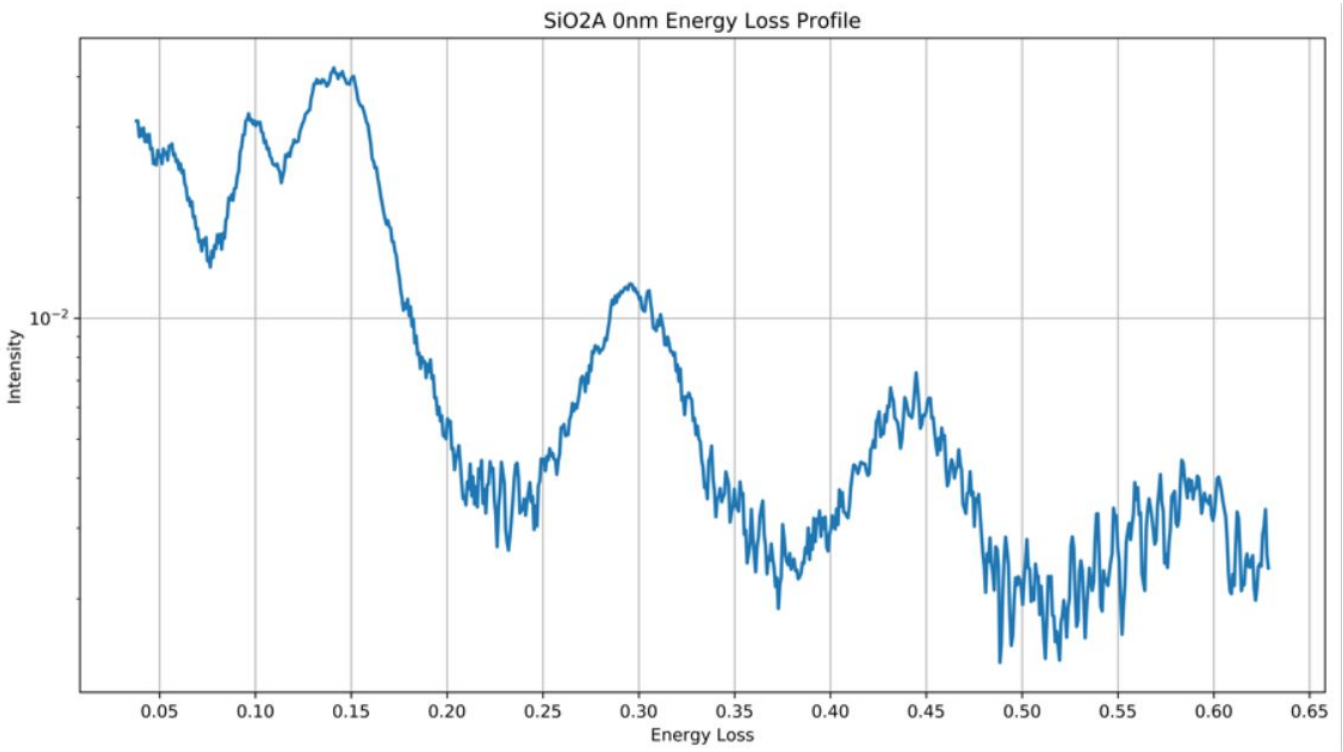

Fig4 Example (SiO2-A 0nm) energy loss profile in log scale in energy loss side

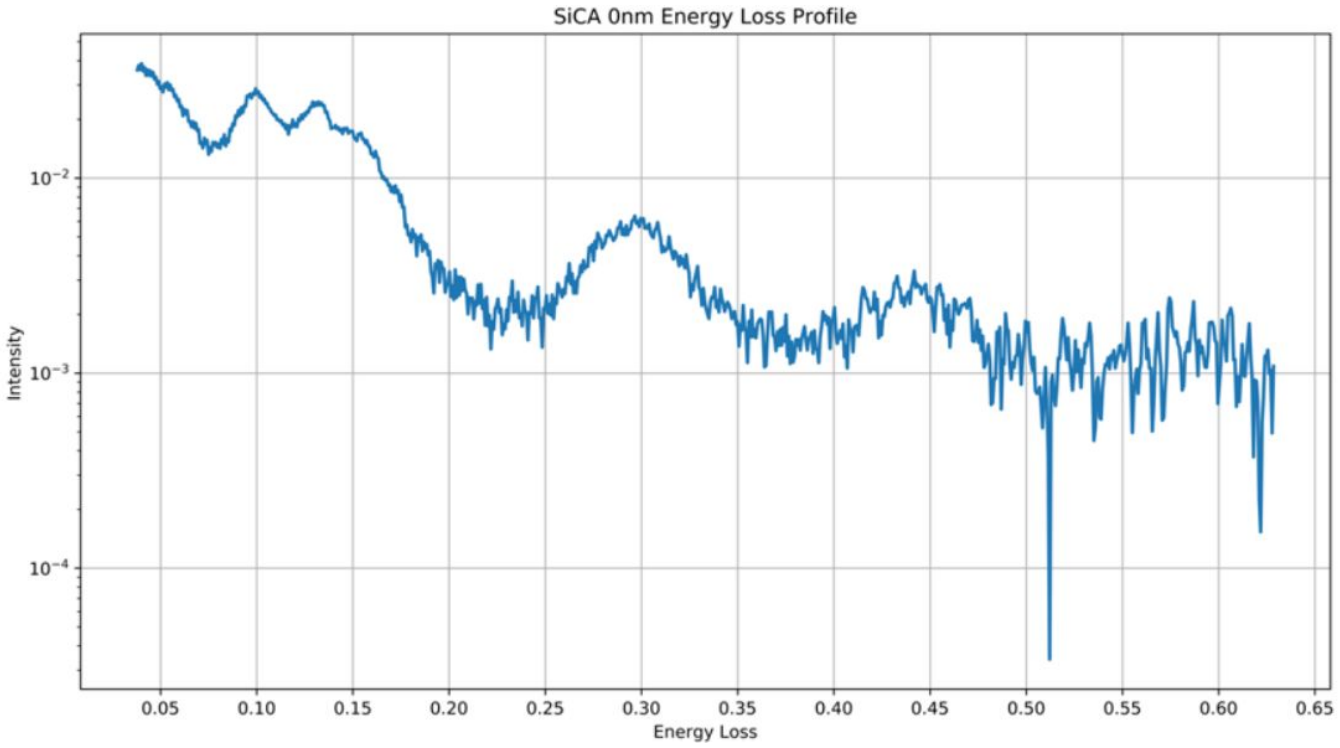

Fig5 Example (SiC-A 0nm) energy loss profile in log scale in energy loss side

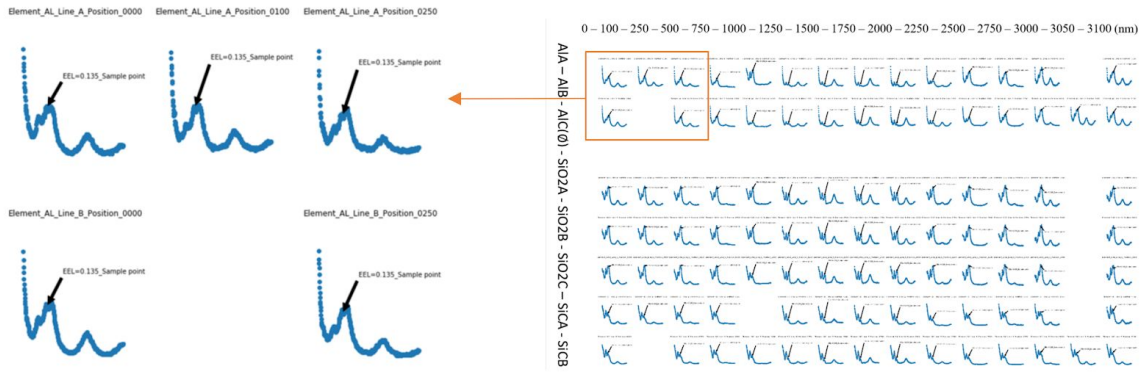

Fig6: The intensity plot corresponds to energy loss.

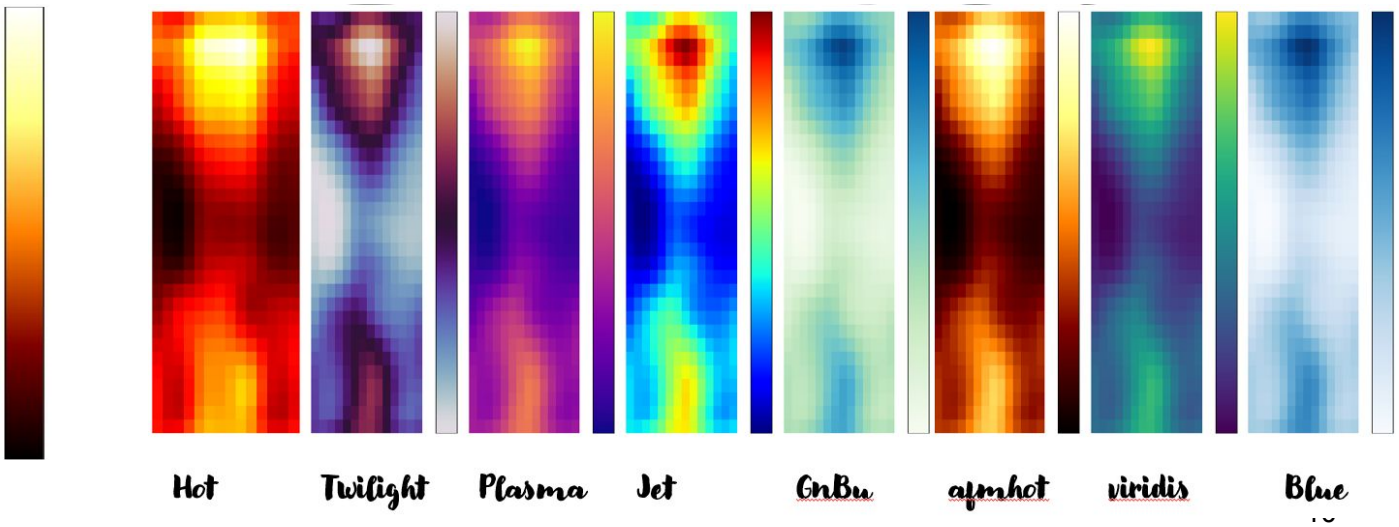

Fig7. color bar

Fig8. Different cmap comparison for energy level at 0.133eV

Fig9 Raw Data plot with/without spacing

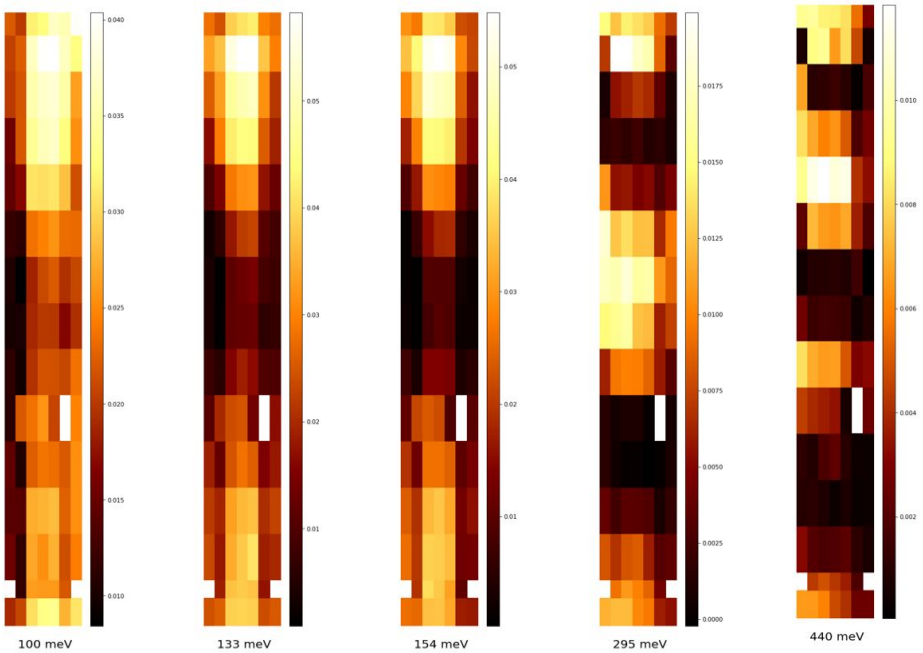

Fig10, Non Scaled Raw Data Plot

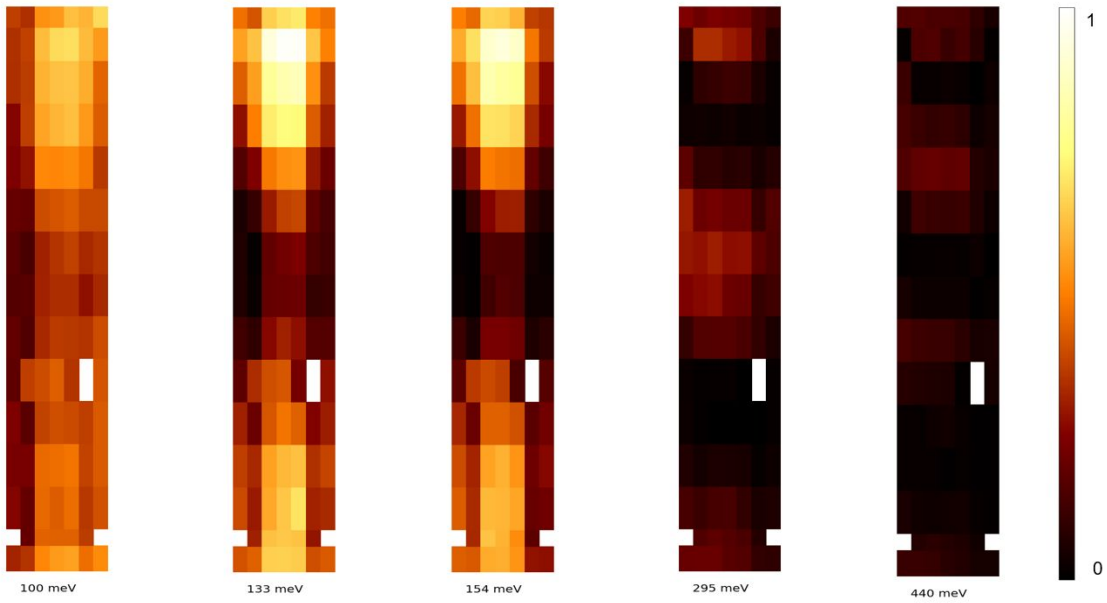

Fig11, Normalized raw data

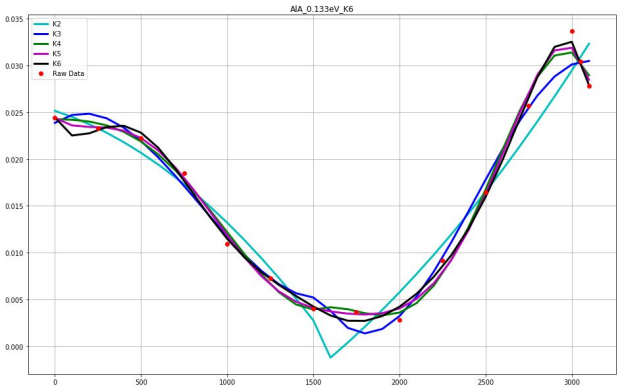

Fig12 Lsq_Bspline different K value comparison in low energy (0.133eV)

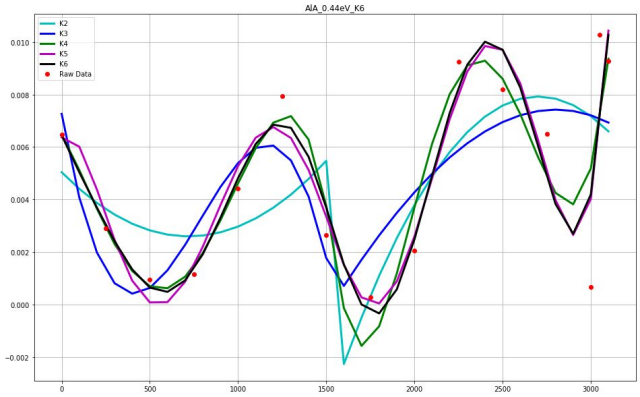

Fig13 Lsq_Bspline different K value comparison in high energy (0.44eV)

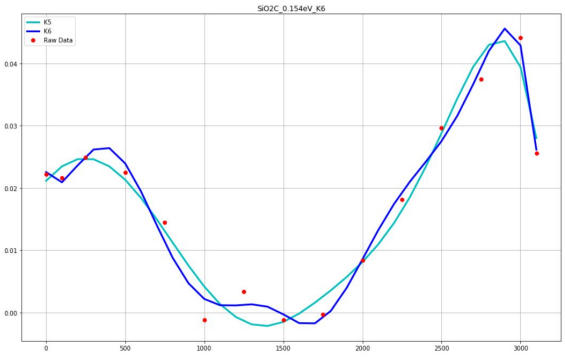

Fig14 K=5 Comparison at 0.154eV.

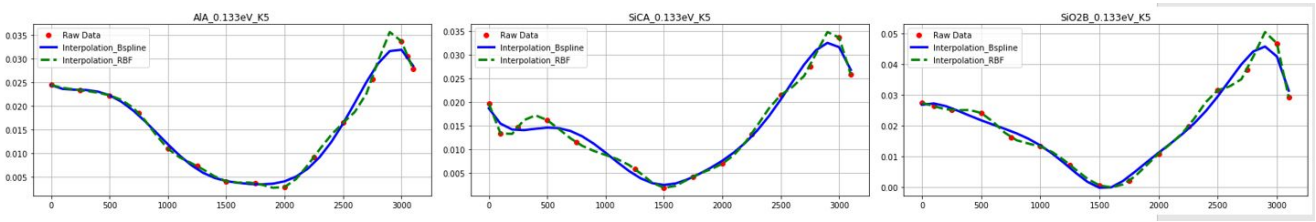

Fig15 Vertical Interpolation method comparison in low energy (0.133eV)

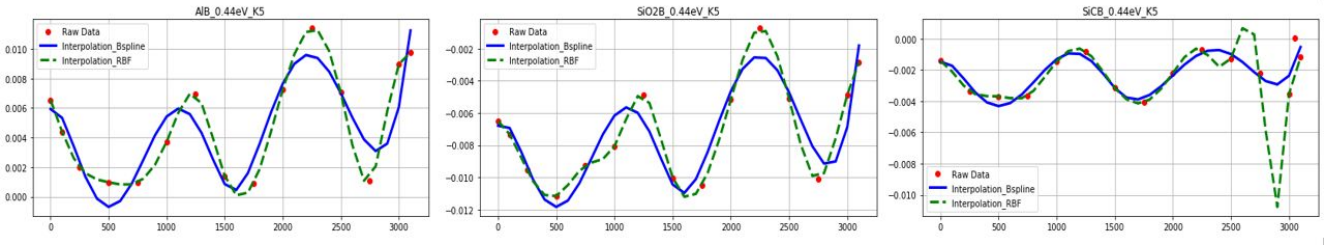

Fig16 Vertical Interpolation method comparison in high energy (0.44eV)

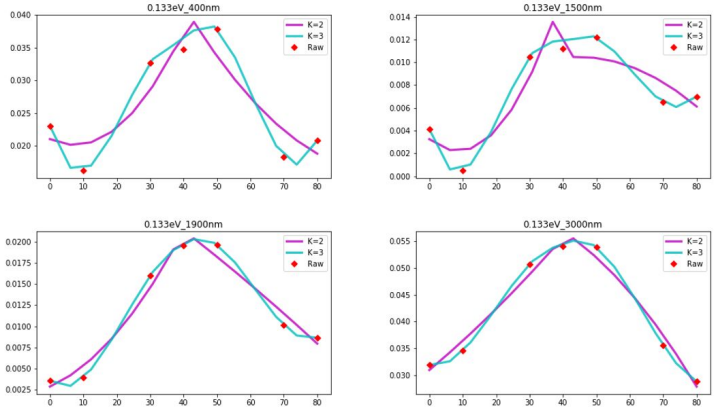

Fig17: Horizontal Interpolation method comparison B spline with k=2 and 3 in low energy (0.133eV)

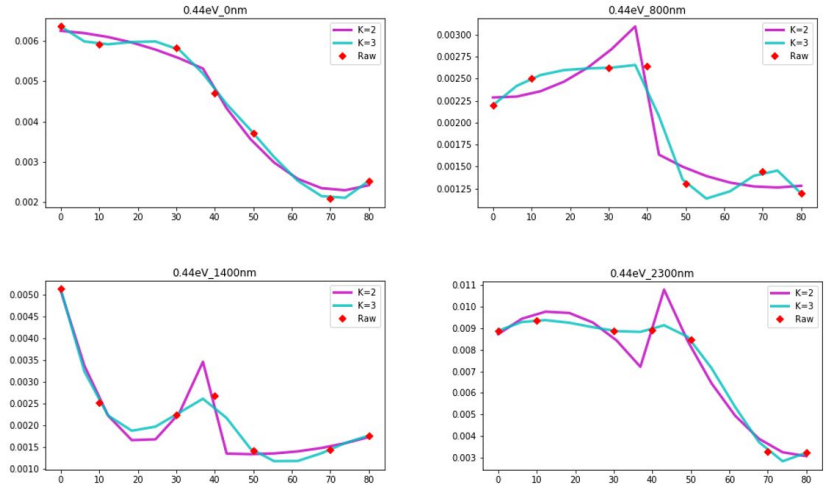

Fig18: Horizontal Interpolation method comparison B spline with k=2 and 3 in high energy (0.44eV)

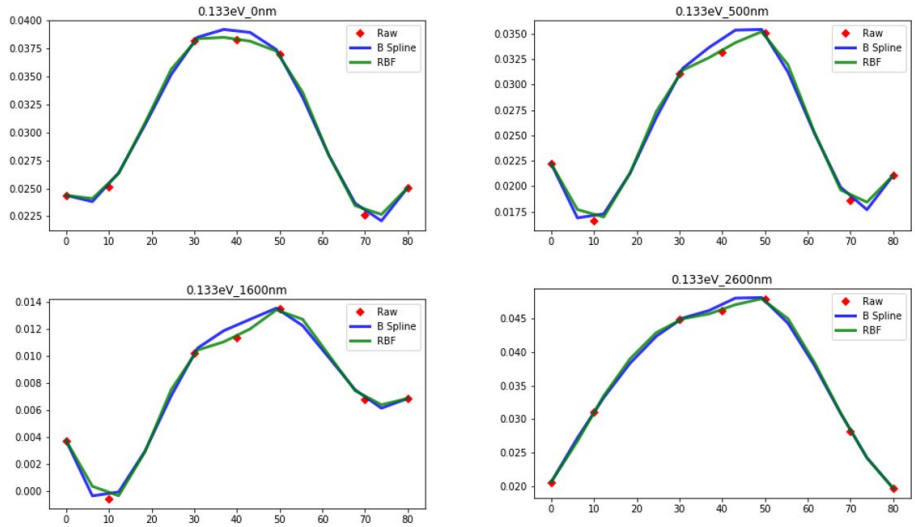

Fig19: Horizontal Interpolation method comparison RBF and B spline with k=3 in low energy (0.133eV)

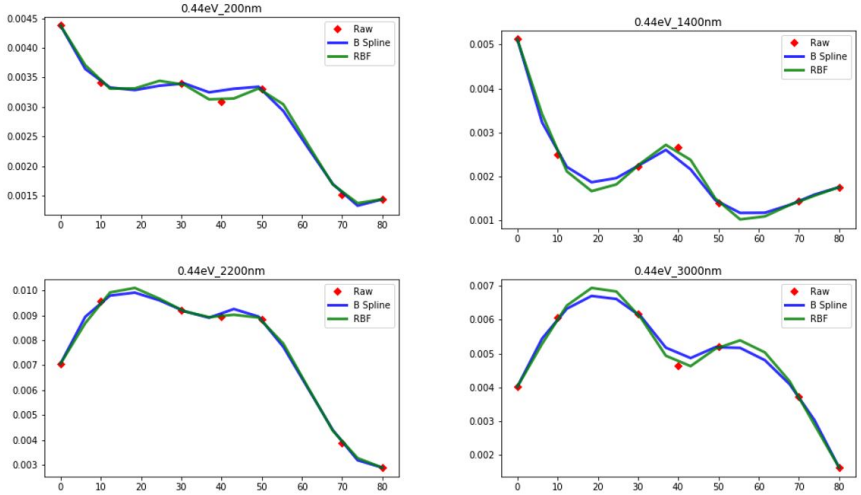

Fig20: Horizontal Interpolation method comparison RBF and B spline with k=3 in high energy (0.44eV)

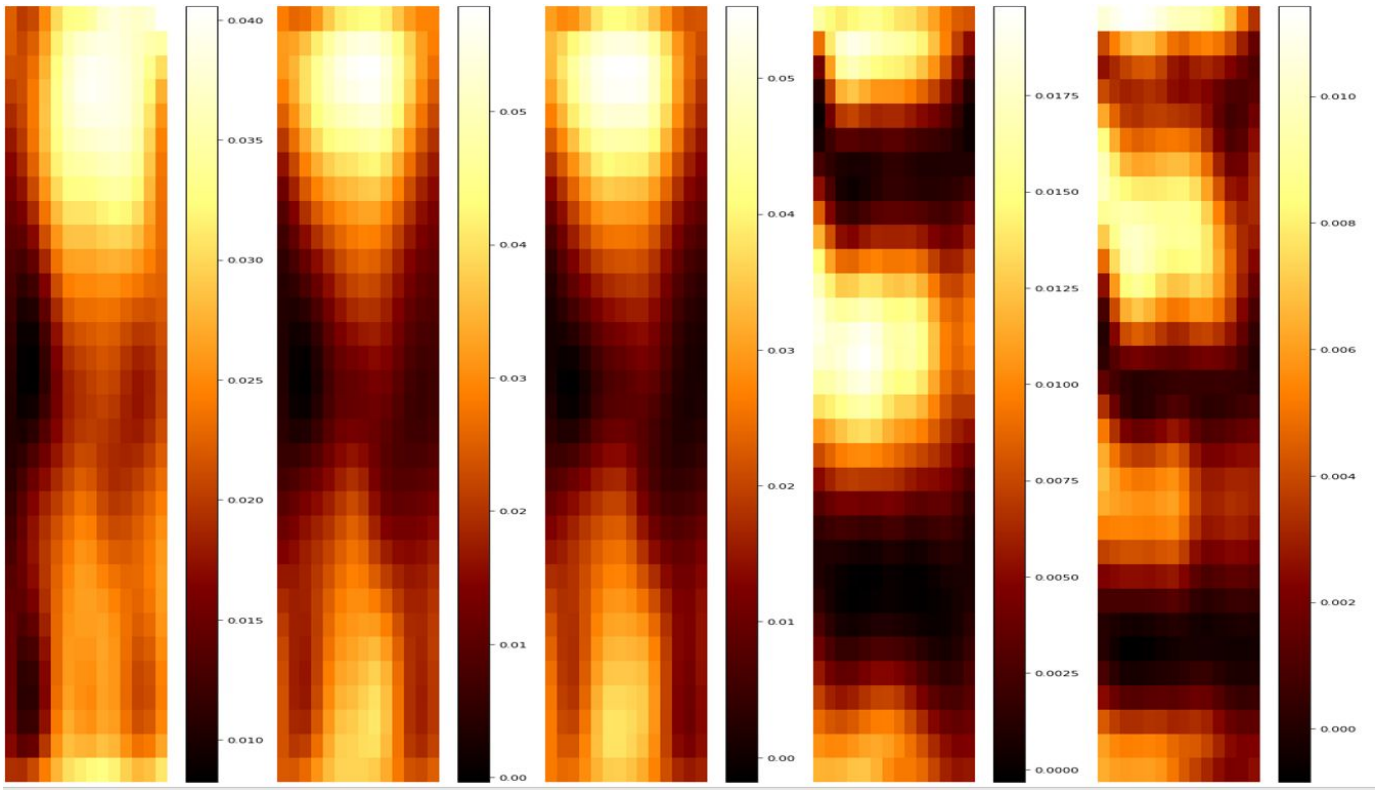

Fig21: Interpolation Result Non Scaled (0.1/0.133/0.154/0.295/0.44eV)

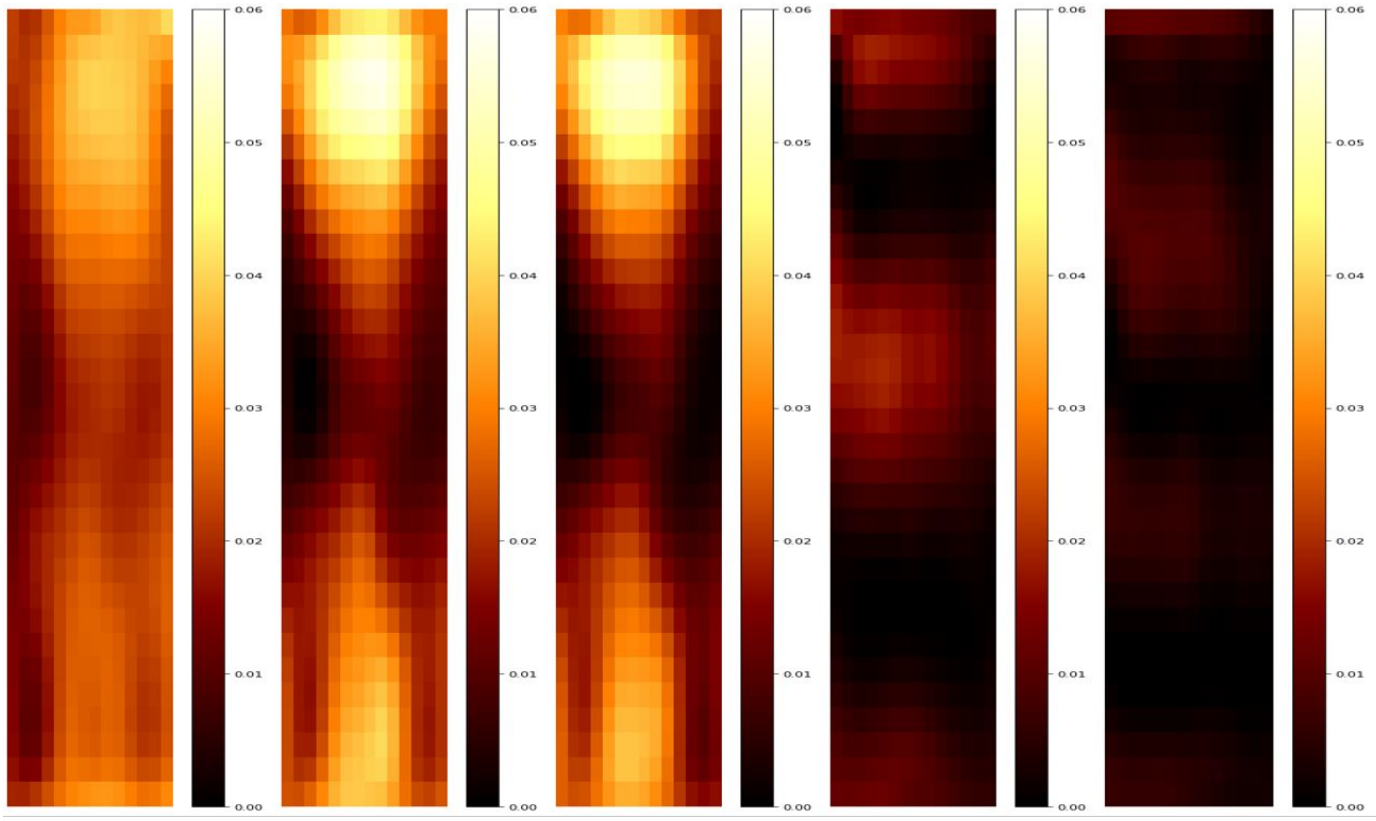

Fig22: Interpolation Result Scaled (0.1/0.133/0.154/0.295/0.44eV) (0-0.06eV)

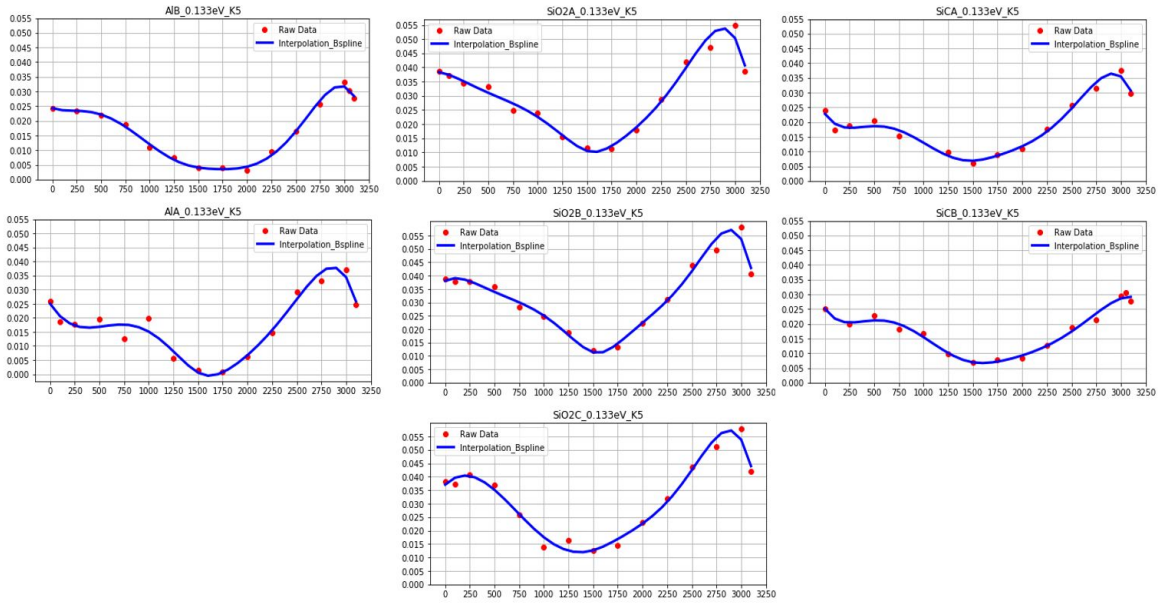

Fig23a. interpolation validation

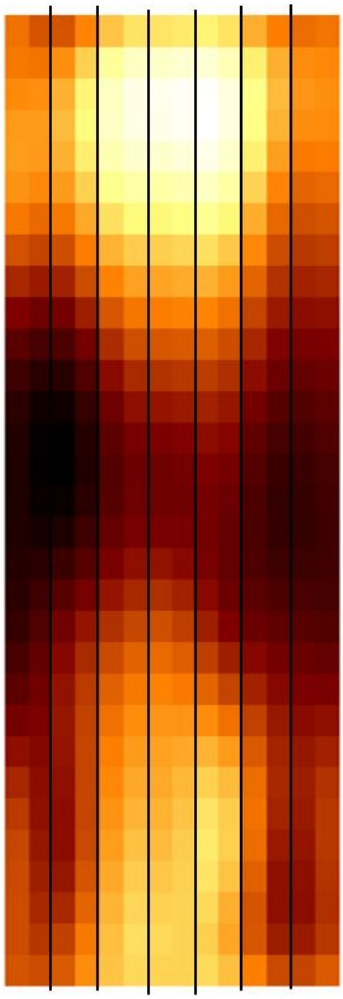

Fig23b, interpolation validation

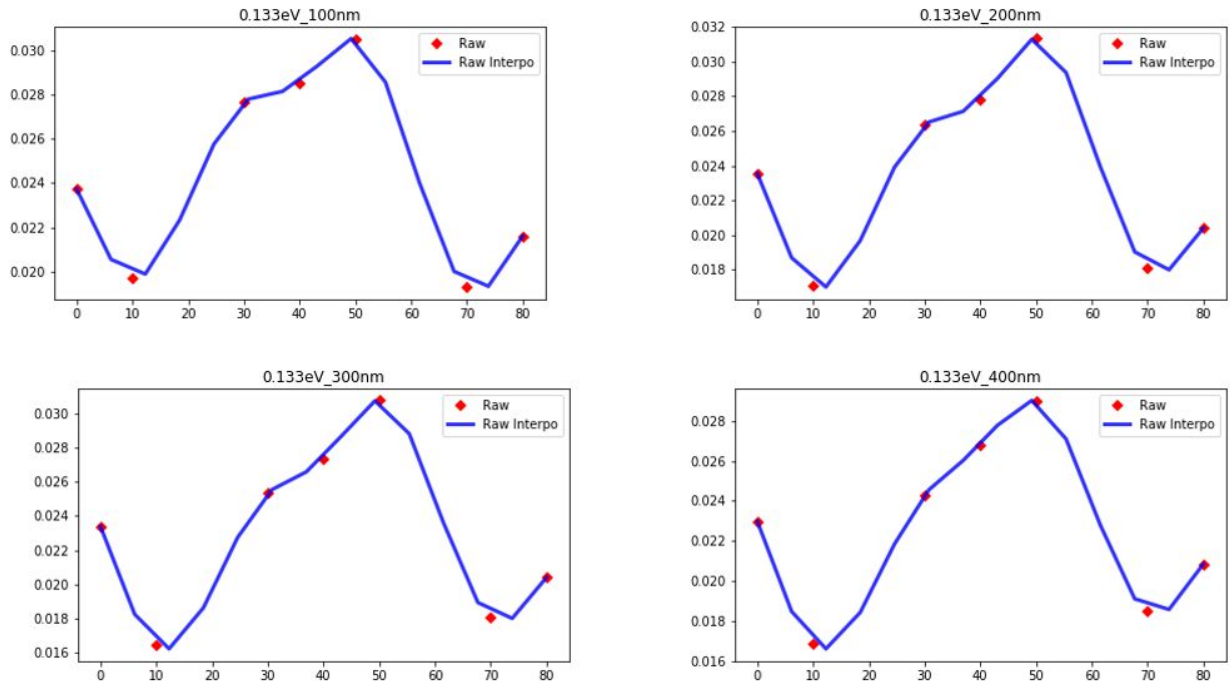

Fig24. Horizontal interpolation used for validation

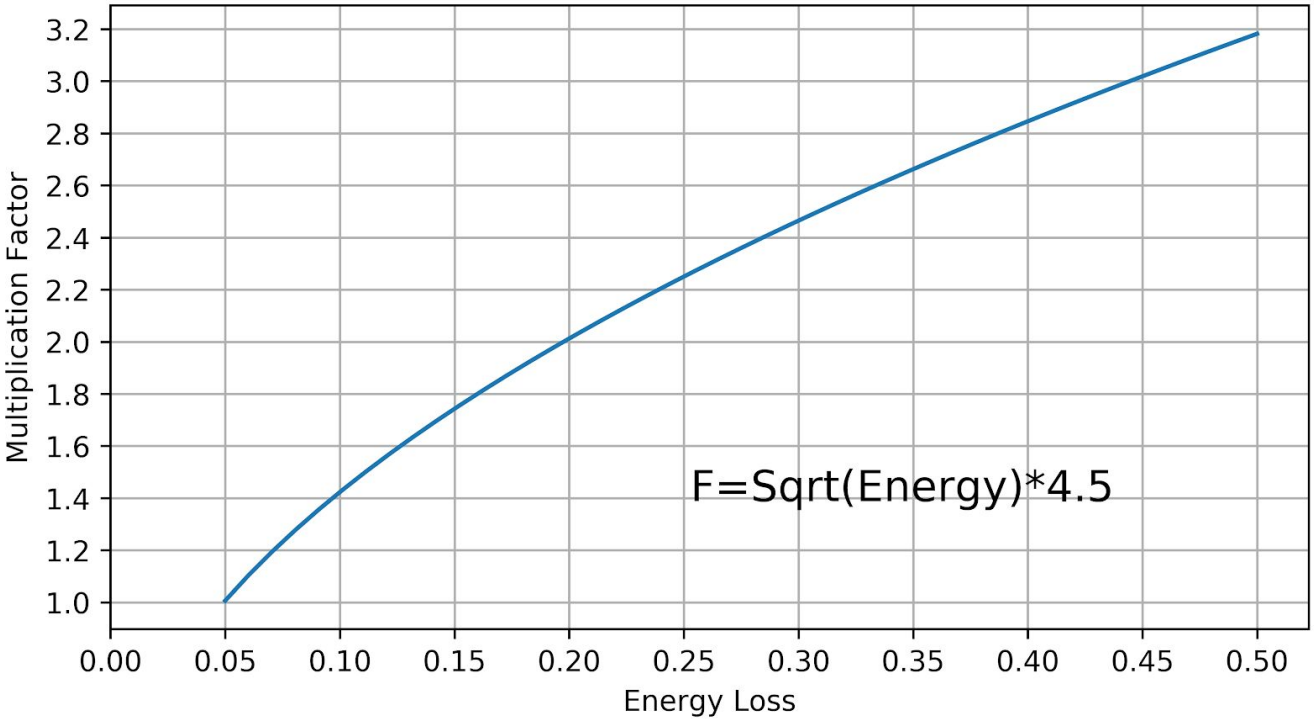

Fig 25 Multiplication Factor Formula for Data Adjustment

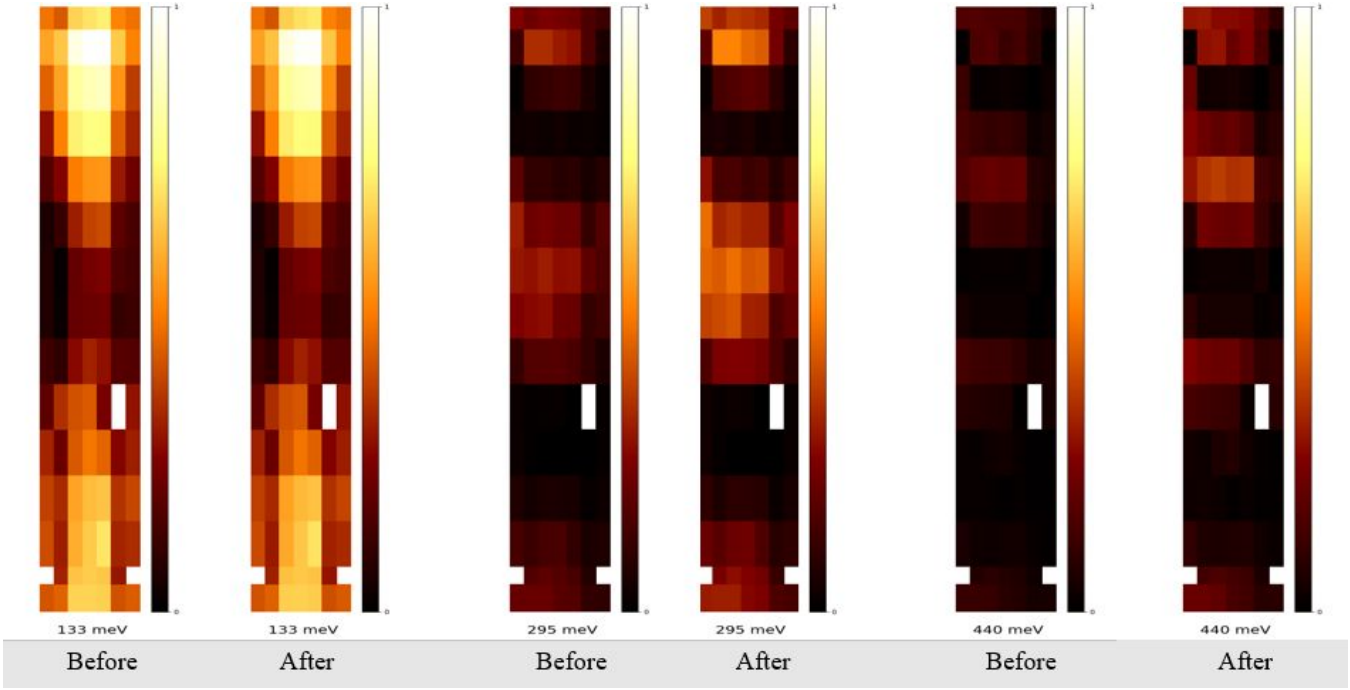

Fig26, The plot before and after multiplied by multiplication factor

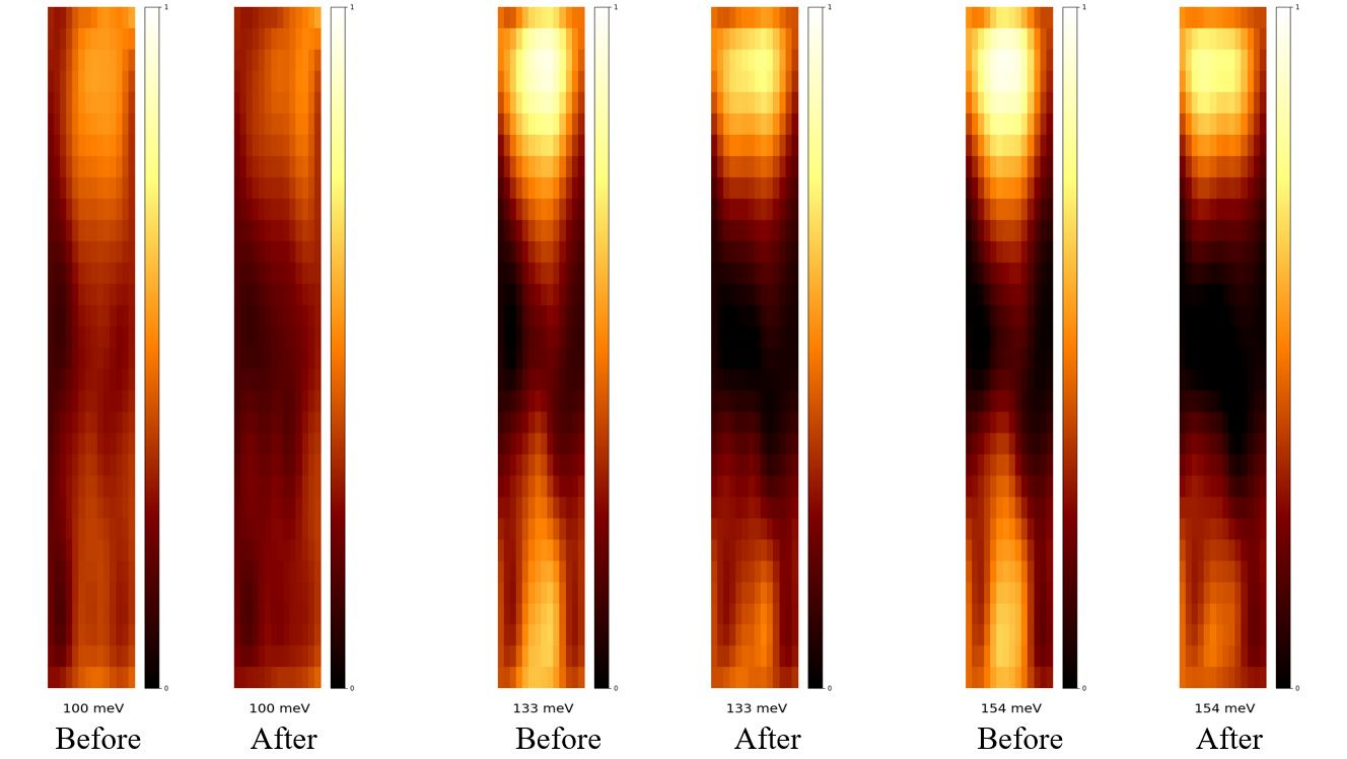

Fig27, The plot for 0.1/0.133/0.154eV before and after subtraction

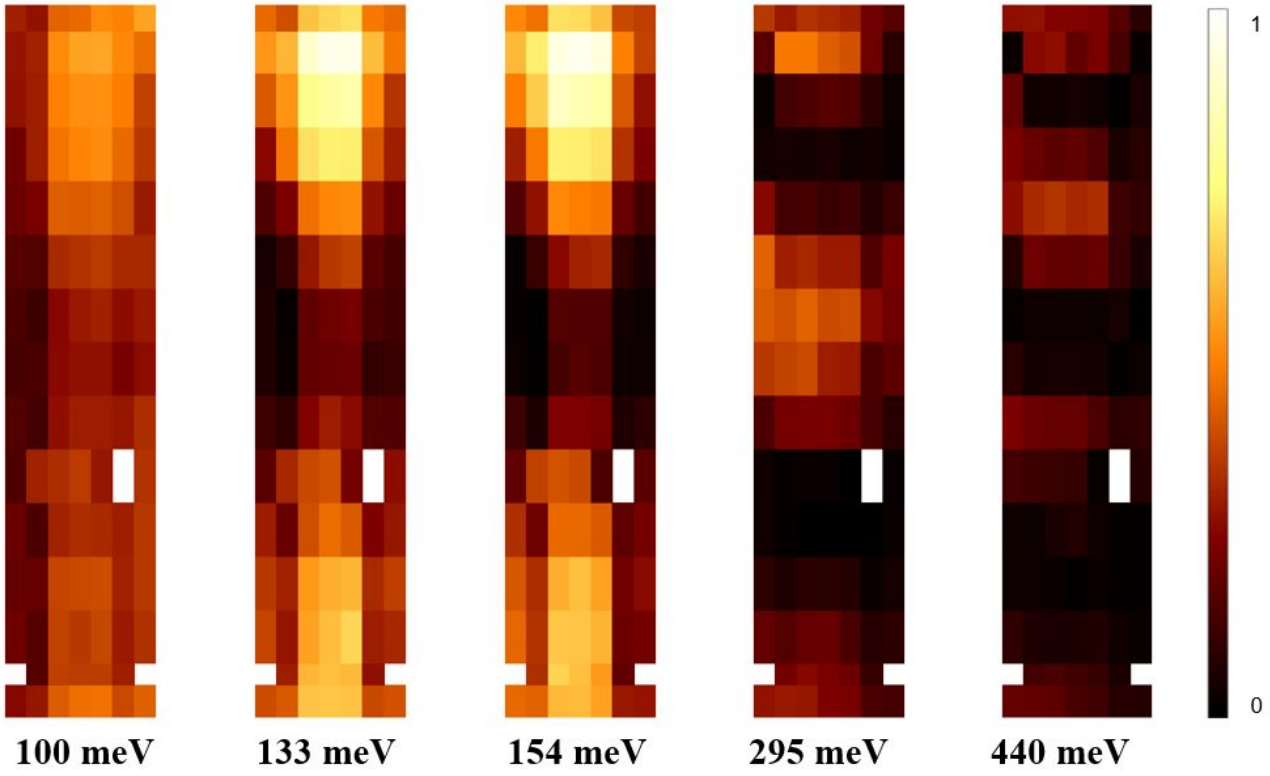

Fig28: Raw Data Final Result

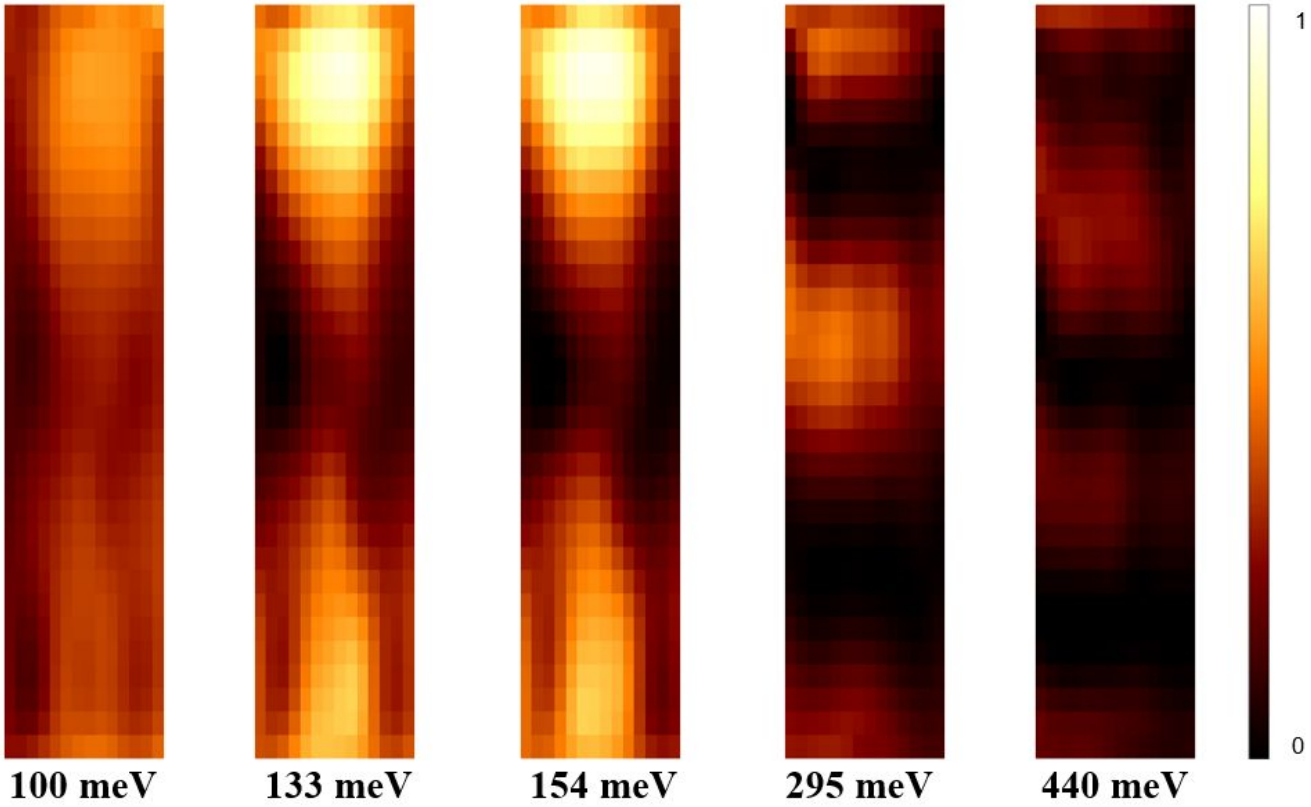

Fig 29: Interpolation Final Result

Appendix B

Interpolation method

B Spline

B Spline is made up of several basis functions which are framed by controlled points. Thus, the function can be seen has a linear combination of several segments, which can be represented as

S(x)= $\sum_j c_j B_j(x;t)$, c_j is the coefficient for each basis function, B_j is the basis function.

Basis function is defined on a knot vector: T=($t_0, t_1, ..., t_{k-1}, t_k, t_{k+1,...,} t_{n-1}, t_n, t_{n+1,...,} t_{n+k}$)

Total element in this vector is equal to the sum of total control point n+1, and the order of the curve k (n+k+1). The span of each knot is $t_i \le t \le t_{i+1}$ which is mapped onto a polynomial curve between two successive joints r(t_i) and r(t_{i+1})

Least Square B-spline

Based on the formula of the B-spline, this method will minimize the error of the interpolation line and the raw data by optimizing the

$$\sum_j (w_j * (S(x_j) - y_j))^2$$

RBF (Radial basis function)

Radial Basis Function is a method used for interpolation for multivariable linear combination of terms which is based on the univariate function. This method was first being used in the 20th century to approximate the function of data which has a finite number by (Powell 1981,Cheney 1966,Davis 1975).

The difference between the radial basis function and the B-spline method is the basis function of RBF only depend on the radial distance from its center. The function can be shown as,

$$\phi(x) = \psi(\|x - x_i\|).$$

The radial basis function can be represented as the linear combination of the basis function

s(x)= $\sum_{i=1}^{N} \lambda_i \phi(r)$.

There are several choices for the basis function. For example, Multiquadric: $\phi(r) = \sqrt{1 + (\varepsilon * r)^2}$, Gaussian: $\phi(r) = e^{-(\varepsilon * r)^2}$ ...