

A MULTIREOLUTION BINNING ALGORITHM, TO ANALYSE NEUTRON SPECTROSCOPY DATA

I. Bustinduy
ISIS Facility
Rutherford Appleton Laboratory
Chilton, Didcot OX11 0QX, United Kingdom
i.bustinduy@rl.ac.uk

F.J Bermejo and G. Bordel
Dept. of Electricity and Electronics
Univ. Basque Country
P.O. Box 644 Bilbao 48080, Spain
jbermejo@we.lc.ehu.es – german@we.lc.ehu.es

ABSTRACT

Volume modelling and visualization pose important challenges for scientific data handling. Neutron spectroscopy is an intensity-limited technique where efficient data capture is a must. Such requirements are usually met using wide angular coverage detectors. Such devices may generate several gigabytes of information produced for an individual experiment, which needs to be handled within short lapses of time.

This paper describes a technique for the construction of a volume model using information coming from different levels of 'binning' resolutions to be applied to the raw (as measured) data. The technique which is adaptive in nature provides an efficient representation of the information, allowing to explore in detail regions of the experimental 4D space where the sought data is concentrated.

KEY WORDS

Scientific and Mathematical Visualization, Data Treatment and Visualization, Volume Visualization, Information Visualization.

1 Introduction

Contrary to the case of other experimental techniques, where the construction of a volume model is straightforward (i.e. by simple interpolation), neutron scattering poses stringent limitations to such an endeavor because of three main reasons. First, the technique has been since its beginnings an intensity-limited tool. This results from the very nature of the process of neutron production (fission reactors or accelerator-based sources) which severely limits the achievable fluxes of particles impinging on the specimen of interest. Second, a neutron spectroscopy experiment on a crystal tries to explore a 4-dimensional space constituted by the three cartesian components of the momentum-transfer vector \mathbf{Q} as well as the scalar energy-transfer. For most applications, data may be sparse, mostly being concentrated in some regions of space. Finally, the achievable statistical accuracy of the collected data may vary easily from one to another region in the reciprocal space, because of instrumental limitations.

For reasons just delineated, an accurate representation

of the measured data, including some measure of their reliability constitutes the first step in data treatment. To do so, an accurate volume model needs to be developed from observations together with available a-priori knowledge on the specific problem to be addressed (i.e. crystal symmetry etc.).

Traditionally, scientific visualization processes comprise three, well-defined stages, that are:

1. *Model building* (interpolation, binning, resampling, subsetting, etc.).
2. *Mapping* (contouring, isosurface, etc.).
3. *Rendering*

Here we focus ourselves onto the construction of a volume model for data sets generated by a neutron chopper spectrometer. This leading method [1, 2], which visualization techniques are still in its infancy, requires to build a volume model. Because of the need of detecting and filtering out spurious data, as well as to avoid data correlation, a new way of building a hierarchical grid is here proposed. It is important to notice that practically all visualization tools require some type of volume model for their application, even in more innovative scenarios as *integrated visualization model* proposed by Robertson and de Ferrari [3] or the *Steering Model* proposed by Earnshaw and Jern [4], the construction of a volume model is required for the efficient analysis, visualization and transmission of the scientific data. Making the proposed model interesting in other areas where intense limited tools are used.

2 Data Characteristics

An idealized scattering experiment of the kind we are interested in (direct geometry), consists in the selection of a neutron beam with well defined energy and propagation direction (or wavelength or neutron velocity), that is made to exchange energy and momentum with the material of interest. Energy and momentum exchange are then measured in terms of the change in the velocity of the incident neutrons as well as in the direction of the outgoing (scattered) beams. For such a purpose a technique known as time-of-flight is

commonly used, where the change in velocity of the outgoing neutrons is measured by the time taken to reach a detector positioned at a given distance while the change in momentum is measured by the use of detectors positioned at different angles with respect to the incoming beam.

A given detector is thus placed at a given angle with respect to the incident neutron beam, and located at a distance L_2 from the sample, while this sits at L_1 from the neutron source. A beam of incident neutrons travelling with wave-vector \vec{K}_i ($= 2\pi/\lambda$) where λ stands for their De Broglie wavelength, will interact with the sample resulting in scattering at different angles, having final wave-vector \vec{K}_f .

The relevant quantities here are the momentum transferred to sample

$$\vec{Q} = \vec{K}_i - \vec{K}_f$$

As well as the energy exchange

$$\varepsilon = \frac{\hbar^2}{2m_N} \left(|\vec{K}_i|^2 - |\vec{K}_f|^2 \right). \quad (1)$$

The change in neutron energy is thus determined by the corresponding change in the time taken to reach the detector t . As we are dealing with neutrons within the thermal and epithermal ranges (below some 10 ev. or so) non-relativistic mechanics holds and thus the neutron velocity is calculated from $\vec{v} = L/T$, where $L = (L_1 + L_2)$ is the total flight-path. The neutron energy is then given by

$$E = \frac{1}{2} m_N \vec{v}^2 \quad (2)$$

As a result, by measuring the time-of-arrival of neutrons to the detectors one derives the final wave vector for each one of the detected neutrons.

A crystalline solid by virtue of its inherent anisotropy requires a three-dimensional variable to describe its structure. The latter is specified in full in terms of a decomposition into crystal planes that are indexed according to well established procedures (Miller indices). Because of the nature of the experimental technique the position, of an entity within the crystal is most conveniently described by setting ourselves within the reciprocal (Fourier) space. There, the crystal planes are specified in terms of three h, k, l indices and the change in momentum thus involves a three-dimensional variable \mathbf{Q}_{hkl} . In other words, the information provided by a scattering experiment involves searches within a four-dimensional space, energy-transfer being the additional dimension. This sets our basic problem as one dealing with searches over a four-dimensional space that contains the relevant information within rather restricted regions. That is, the scattered intensities corresponding to static (positional) correlations give rise to strong intensity 'spots' within \mathbf{Q}_{hkl} leaving most of it empty. Conversely, the signature of atomic or spin motions will be comprised within narrow regions of $[\mathbf{Q}_{hkl}, \Delta\mathbf{E}]$.

For a severely intensity-limited technique exploration of the adequate portions of $[\mathbf{Q}_{hkl}, \Delta\mathbf{E}]$ is thus a must. Information will come in form of histograms of spectrum number and time of flight bin. These quantities are then converted into momentum-transfer and frequency (or energy-transfer).

This study is focused onto the most demanding case, that are single-crystal experiments performed on time-of-flight neutron spectrometers. Last generation machines employ Position Sensitive Detectors (PSD) that provide close to continuous coverage over a large solid angle array in the forward direction (about 160.000 detector pixels and an additional narrower strip of detectors in the horizontal plane). On such instruments one can control of resolution in both momentum and energy-transfers by choosing different incident energies and instrument setups. Such a pixellated detector array gives the experimenter a possibility for high-speed visualization and data assessment. Development of analysis software thus plays a vital role in the exploitation of data. Time-of-flight and pixel position of the scattered neutrons are stored in an array of 10^8 pixels, from which the scattering function is constructed, in software, in a volume of reciprocal space of typically containing 10^7 volume elements.

The data is then represented as intensity (i.e. neutron counts) measured along some trajectories within wave vector-energy space. The origin of these trajectories define a 2D grid and so a set of trajectories defines a data volume. The data format of interest for physicists corresponds to projections in wave vector-energy space $(\vec{Q}_h, \vec{Q}_l, \vec{Q}_k, \varepsilon)$.

These projections are called *viewing axes* $\vec{U}1, \vec{U}2, \vec{U}3$ ¹, and for the purpose of plotting they should be chosen to be orthogonal. Notice that the fourth vector-component, that is the energy axis is, by definition orthogonal to all wave vector components.

Our aim here is to provide the necessary technology to be able to apply the different volume visualization techniques in this scientific area. The nature of the problem, requires to rebin raw data in a hierarchical grid over relevant viewing axes, retaining high flexibility and preserving high statistics areas. This will necessarily pass through the construction of an adaptive volume model that keeps track of local variations, choosing a large integration step on low statistics zones, and a small integration step otherwise.

¹The experimental technique (direct geometry) requires to fix the value of \vec{K}_i , this means that it is possible to express the wave vector-energy space in terms of just 3 vectors, because the $(\vec{Q}_h, \vec{Q}_l, \vec{Q}_k, \varepsilon)$ components are not independent. (In the case of indirect geometry the fixed parameter would be \vec{K}_f , not affecting to the projection, from 4D to 3D space).

3 Related Work

Work previously reported mostly deals with 2D images. Within those, the **Adaptive Intensity Binning** of Sanders and Fabian [5], is perhaps a close precursor of the methods described below. Such authors consider a bin as a collection of pixels, meaning by a pixel one of the individual picture elements. The algorithm attempts to adaptively bin a single image based on the number of photons in each region. The number of pixels plotted in a graph is always constant, as also is the number of pixels that form the image. Its main interest is to facilitate human perception, plotting with the same color, pixels, that are contained in the same bin. This may present problems in cases where, for example; we have two bins a and b , where a is made by just one pixel (good statistics), and the bin b , is made by 15 pixels, (bad statistics), depending on the user-selected color scaling, high statistics areas are practically undistinguishable by the user.

Also, a **Voronoi Tessellation** scheme has been discussed by Cappellari and Copin [6]. There, Voronoi Tessellation is employed to produce optimal 2D binnings. The procedure introduces some improvements, and in particular it satisfies the *Topological, Morphological and Uniformity Requirement* presented by the authors of the paper.

The algorithm presented in this paper shares a viewpoint close to that presented by Sanders and Fabian. Our choice was determined by factors related to the size of our input data (8 million counts per experiment), as well as the ease in implementing 3D versions.

4 Multiresolution Algorithm

In contrast to a common binning algorithm, where given a data space a grid is built just adding to the bin all the counts or data points enclosed into the boundaries of the bin $b = \{b_1, b_2, b_3, \dots, b_k\}$, the presented multiresolution binning algorithm creates an adaptive volumetric model from discrete volume data.

By convention we will here consider the term *bin* or *pixel* as a collection of *counts* or *data points*, where the data points are simply the raw data we want to analyse. the pixel term will be used when dealing with images produced by equal-width or *fixed* binning algorithms.

We will construct a grid where the bin size is not fixed, but rather the boundaries of each bin will be determined by the statistics of the data points that encloses each bin. As each data set has its own peculiarities; each instrument has a very particular accuracy limitations, each sample reacts in so many different ways depending on the incident energy beam, orientation, temperature, pressure; it is impossible to determine a priori the statistical function that will define the bin size. This makes almost impossible to implement a predefined automatic decision about the general parameters for the representations. Interaction is the way to determine these parameters letting the user deal with their values for each particular data set [7–9]. Each

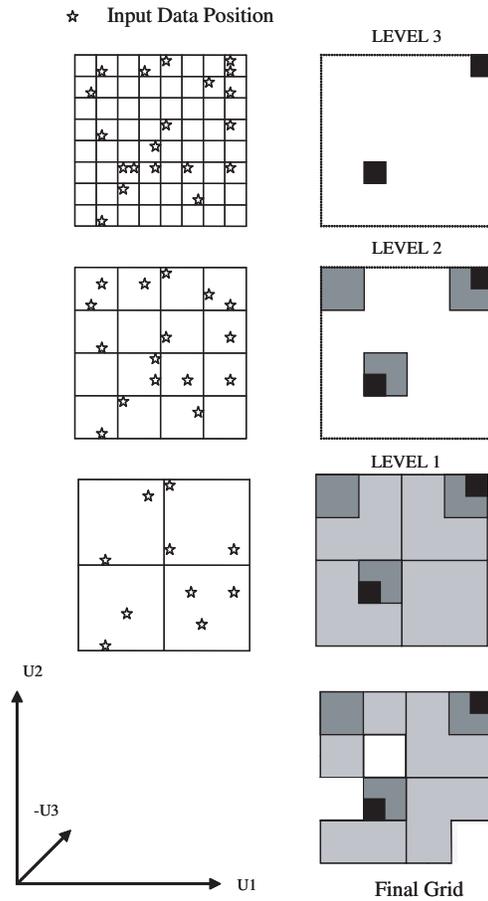


Figure 1. To illustrate the multiresolution algorithm consider the 2D case, where the number of levels is equal to 3, and the resolution pattern has chosen to be determined by the Level 2 binning size. Although the principles can be extended trivially to higher dimensions

scientist knows better than anybody the particular behavior of their sample. In particular, the user establishes the boundaries of the grid, the maximum binning size, and the minimum binning size applied to each dimension, as well as the ratio error/intensity, *RATIO* that will act as a threshold, validating the size of each bin.

In other words, the algorithm will start creating a grid with a minimum bin size (also determined by the user), then we will calculate the arithmetic mean of intensity and the associated error value in each bin as follow:

$$S = \frac{1}{n} \sum_{i=1}^n S(x_i)$$

$$ERR = \frac{1}{n} \sqrt{\sum_{i=1}^n ERR(x_i)^2}$$

Where n stands for the data points x_i , enclosed to the boundaries of this bin $b = \{b_1, b_2, b_3, \dots, b_k\}$. In our case the limits are taken as inclusive to the left and exclusive to the

right along each axes '[]' to maintain compatibility with already existing binning algorithms.

After that, a decision based on the error/signal ratio is made about accepting the bin as a valid one or not:

So **IF** $ERR/S \leq RATIO$

THEN this bin will be selected, and the data points that form the bin, will be masked from the original data set, this action is taken to avoid error correlation dispersion. In such a way we impose the restriction that each data point will only contribute to only one bin.

ELSE the bin size will be doubled, collecting only the remaining data points.

This process will continue up to the maximum binning size.

And finally, the algorithm will mask the resulting image with a characteristic pattern of the instrument resolution, to avoid painting areas where there is no real input data.

So, the method can be seen as four steps. Given a DataSet of points $\{x\}$, these three steps are:

1. Construct the GRID for $LEVEL = 1$, and insert all data point x in DataSet into it.
2. For each level;
 - Extract the bins which ERR/S ratio is below the threshold, to the FINAL GRID
 - Remove those contributing points from GRID
3. Mask the resulting image with the instrument resolution pattern.

4.1 Implementation issues

In order to avoid possible problems related with the boundaries that arise from the specification of maximum and minimum binning size, The user is left to specify the grid boundaries and the maximum binning size, for each dimension. The same applies to the threshold value ($RATIO$), where the way minimum binning size is specified, is by the number of levels ($NLEVEL$), the algorithm will go up, doubling the size of the bins. Another issue, is how to deal, with data points having no signal ($S = 0$). This means an instability in our condition and the way we have considered these points is to be above the threshold.

The relationship between the resolution in each axis is set from $\Delta U_{max} = 2^{NLEVEL-1} \cdot \Delta U_{min}$. This means that the number of levels this algorithm covers in each dimension, will be the same. This restriction allows us to use consecutive convolution products between matrices, improving the execution time of the algorithm.

One of the main limitations, is what to do with the remaining points, that the algorithm has not collected, that is those which remain below the threshold even if we use the largest binning size. Different approximations have been

taken such as avoid plotting the remaining points or collecting them within the maximum binning size scheme, and plotting them using a different transparency level. Our own experience shows that modification of the transparency attribute changes the perception of the color adopted by the bin. It thus seems better to show them with the same opacity level in the final image, since the contextual visual information [7, 9, 10] is more important than the imprecision we gain in the last level of integration, giving some level of uncertainty when the user visualizes the biggest bins.

The final implementation enables the user to fully interact with the algorithm by means of a GUI, enabling to choose the boundaries, binning maximum and minimum sizes for each axis, error-intensity threshold and color axis scaling.

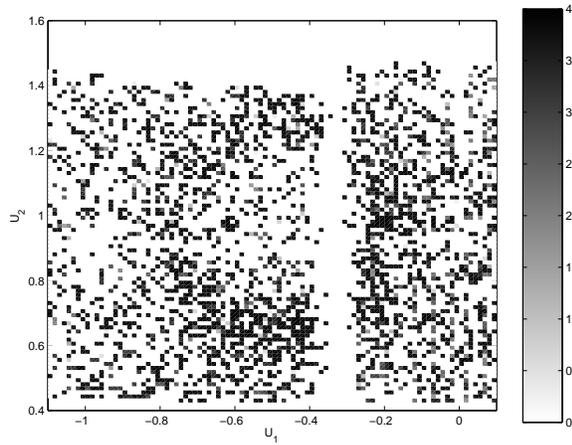
5 Results

5.1 2D Results

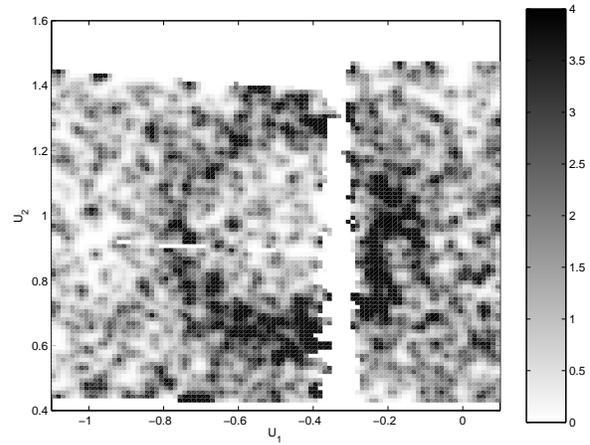
The most widely used method of volume visualization, consists in creating successively 2D plots, through cuts along the data volume. This provides a compromise between speed of volume viewing and accuracy of representation of details. It however presents some deficiencies, that will be explained from now on. At present, the available visualization software (named *Mslice* [11]), requires to take a *slice* of data as a function of any two variables from the four components of momentum and energy transfer (\vec{Q}, ε) .

As a result we obtain 2D color intensity plots see fig.2(c) and fig.2(a). Where the resolution will be constant over the entire graph, the user will then decide whether to take a small, or large integration step, which of course, will be determined by statistics of the overall experiment, not adapting to the local peculiarities. It is also possible to use a gaussian shape filter to smooth the obtained 2D color plot, helping the user to gain some insight of the predominant patterns see fig.2(d) and fig.2(b). This represents the traditional solution adopted to represent 'low statistics' data, or to stand out information signal from the background. The chosen smoothing filter, a gaussian shape kernel, acts as a low pass filter, providing a gentler smoothing and preserving edges better than similarly sized common mean filters. In this particular case, the kernel is a rotationally symmetric gaussian low-pass filter of size 3x3 pixels, with an standard deviation of approximately 0.5. The different n-levels of smoothing are obtained by repeatedly, convolving the same kernel n-times.

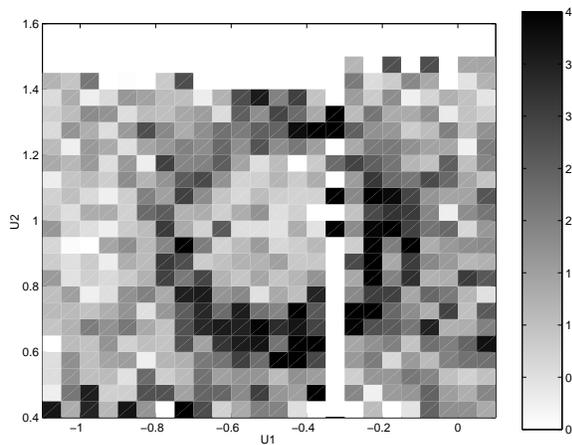
In both previously mentioned figures, a pattern can be perfectly recognized, nevertheless, using a convolution filter we are introducing some correlation, as the output will be a weighted average of each pixel's neighborhood, with the average weighted towards the value of the central pixels. The weighted average is just only in terms of intensity, the error tolerance is no considered here, as a result we



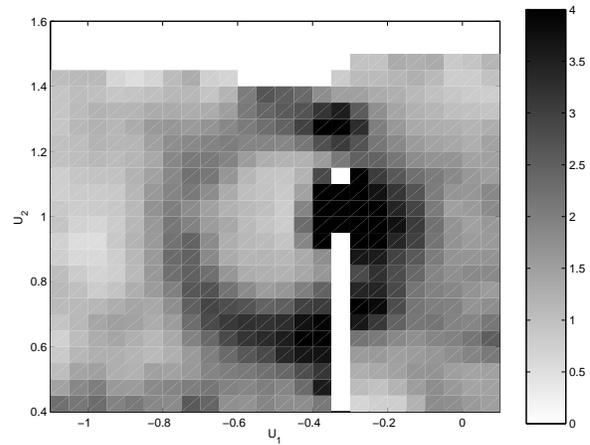
(a) Minimum binning size – 'fixed' binning algorithm



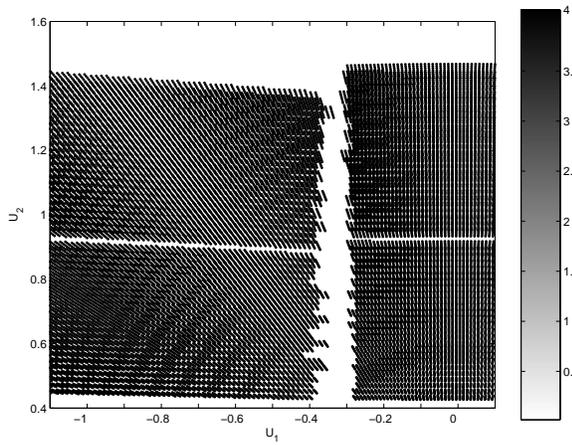
(b) Smoothed minimum binning size – 'fixed' binning algorithm



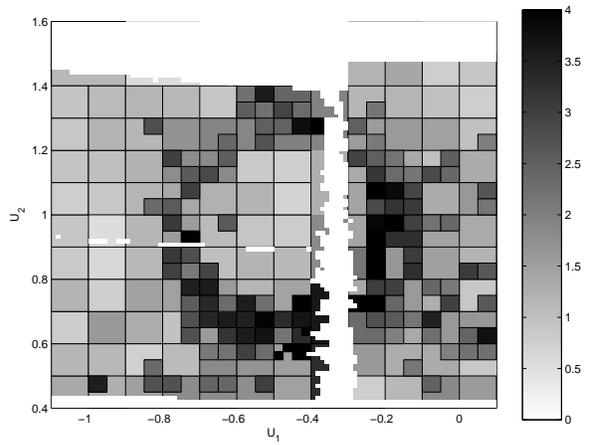
(c) Optimal binning size – 'fixed' binning algorithm



(d) Optimal binning size – 'fixed' binning algorithm after using a gaussian shape smoothing filter



(e) Input raw data



(f) 'Multiresolution binning' Algorithm

Figure 2. Represents different color maps, projected onto the perpendicular axes $U_3 = [0,0,Q_l,0]$, being $U_1 = [Q_h,0,0,0]$, and $U_2 = [0.5Q_h, -Q_l, 0,0]$. In all the cases the color bar has chosen to go from 0 to 4 (abs. units). From (a) to (b) the figures have been produced using a equal-width binning algorithm, which bin size is $[0.0125, 0.0125]$. From (c) to (d) bin size is $[0.05, 0.05]$. Fig.(e) represents a scatter plot of raw input data. Fig.(f) has been produced using the presented algorithm, with a threshold of 0.35, going from the maximum binning size: $[0.2, 0.2]$ to the minimum binning size: $[0.0125, 0.0125]$.

have that a pixel with a very low accuracy contributes in the same way it does a very high accuracy pixel, therefore we lost accuracy in the displayed information, this could mislead the researcher, in terms of confusing measured data with the obtained patterns.

Notice that especially in the fig.2(b) case, much of the noise still exists, although it has decreased in magnitude somewhat. It has been smeared out over a larger spatial region. Increasing the standard deviation of the gaussian kernel, continues to reduce/blur the intensity of the noise, but also attenuates the high frequency details (e.g. edges) significantly.

Finally consider fig.2(f) which is produced by the multiresolution algorithm. It not only displays information about the average intensity in each pixel but also yields the associated error-bar. The intensity is represented by the assigned color to the pixel, where the accuracy of such a measure will be given by the size of the pixel. In addition it represents the outline of the input data more accurately than the current implementation of the 'fixed' binning algorithm, where the boundaries of the detectors, are not handled properly, in particular, we could wrongly infer from fig.2(d) that there is a high intensity peak centered in the $[U_1 = -0.35, U_2 = 1, U_3 = 3]$ position, where it is clear, see fig.2(e), that there is no input data.

We can also integrate along the third axis in some restricted range, in addition to specifying the integration ranges in the other two orthogonal directions, the result being a 1D graph. This 1D graph is what traditionally gives accurate information, showing the intensity values, and their correspondent error-bars. Unfortunately, this plot only shows a very narrow information of the entire experiment, not letting users appreciate the nature of the experiment as a whole.

5.2 3D Results

The present algorithm, was also implemented to construct a 3D grid, see fig.3. Volume visualization enables to obtain a broader vision of the experiment, allowing to localize the area of interest, fast and precisely. It may also make possible to stick out properties of the material under study, that otherwise, using the traditional system; using 2D slices, wouldn't be noticed. As a practical example, a crystal misalignment was discovered in the Rb_2MnF_4 Experiment [12], thanks to the 3D representation.

One of the most popular volume visualization techniques, consist in displaying surfaces of constant intensity, better known as *isosurfaces*. Unfortunately, to get the most of this technique, it is required to have a minimum quality in the input data in terms of signal error, otherwise the obtained image is confusing. For very low statistics data, it has been proven that the best way of visualizing volume data sets consists in using interactive slices and cuts of the volume under study.

Once the volume data set has been constructed we can apply the different volume visualization techniques, we

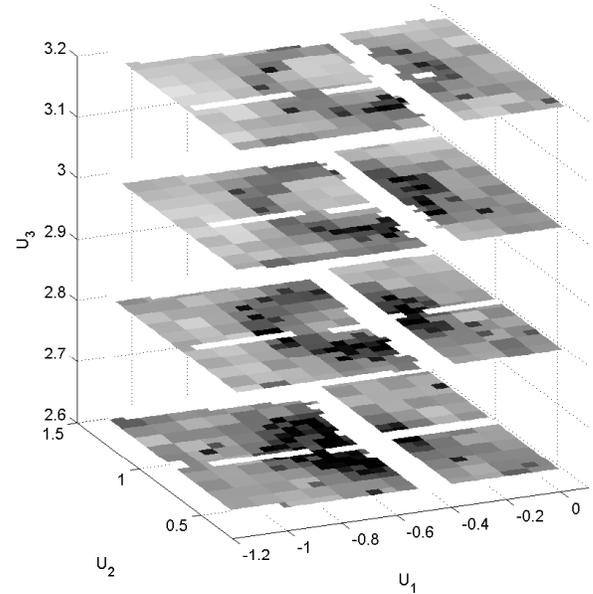


Figure 3. Slices made from a volume data generated from the *cobalt* experiment using the 3D implementation of the multiresolution algorithm. The used threshold value was again 0.35, and bin sizes go from maximum bin size of $[0.2, 0.2, 0.4]$ to minimum binning size of: $[0.025, 0.025, 0.05]$.

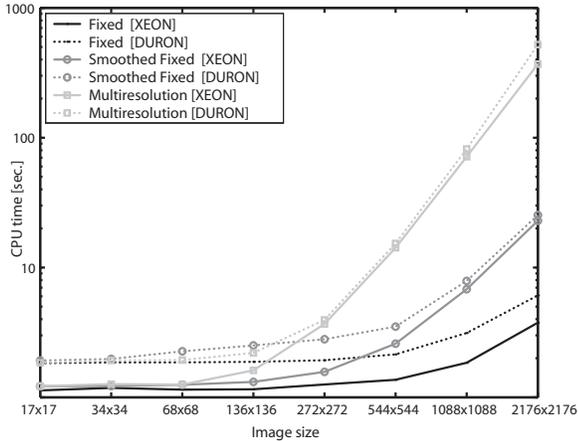
consider to extract better information. In this case fig.3 shows constant slices cutting the volume data set to bring this information.

In the 2D version we always could modify the thickness of the plane, adding more points to the contribution of each pixel. This will not affect the resolution in the visualization of the plane. Of course, getting more points the resulted intensity will be smoothed in each pixel. Whereas in the 3D version; the resolution in one axes and the information contained in the rectangular cubic lattice are linked, which forces us to think in terms of rectangular cubic pixels, where the pixel size will need a careful thinking, to choose the best compromise between the resolution and the information content. The choosing of a very narrow step in one axis forces the perpendicular plane's voxels to contain fewer points. Here again we rely on visual feedback from the end users to choose the correct dimensions.

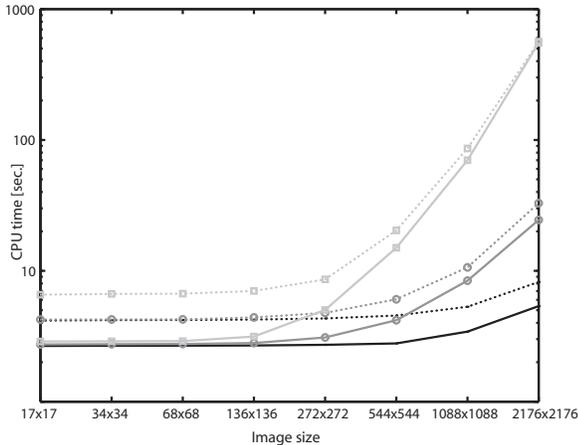
6 Interactivity - CPU Time

Time consumption is a fundamental issue to ensure a visual feedback. The formulation of queries by direct manipulation and the immediate display of the results has many advantages for both novices and experts. This enables users to gain a deeper understanding of the data set, providing information about the nature of the problem that can not be grasped with a static representation. [7, 13, 14]

Some few measurements have been carried out to test



(a) Experiment 1



(b) Experiment 2

Figure 4. Required times (in logarithmic scale) to construct the final image, the x axis show the image size, in the *Multiresolution* case, means the size of the grid constructed when LEVEL= 1 (using the minimum bin size), which has been compared with the time required to construct the same size grid in both *Fixed* and *Smoothed Fixed* cases.

the computational cost of our algorithm. The chosen data sets for this estimation were:

- **Experiment 1:** 244Mb of required RAM and a total of 3.792.208 data inputs to be analysed
- **Experiment 2:** 515Mb of required RAM and a total of 8.036.160 data inputs to be analysed

Both data sets were tested on two different machines,

- **AMD DURON** 1.3GHz Clock Speed, 1,024,000 KB of RAM
- **INTEL XEON** 2.4GHz Clock Speed, 2,096,092 KB of RAM

Fig. 4, compares the required CPU time to build a 2D grid. It represents three different executions: the traditional binning algorithm, the same algorithm applying a gaussian filter after, and the multiresolution algorithm. However, in practice we work within the first 5 levels, which means a resolution divergence of 2^4 times, where the proposed version keeps a reasonable time dependency behavior, below 9 seconds in the worst scenario.

7 Conclusion

Experiments with different data sets demonstrate clearly the effectiveness of multiresolution binning Algorithm when compared with the current 'fixed' version.

The current visualization technique constructs a lattice where each pixel's dimension is fixed for the entire slice or volume. This entails the decision between having good resolution (with the shortcoming of getting low statistics values) or trying to get high statistics, making a broader integration, to construct the lattice (and therefore losing resolution in high statistics areas of the reciprocal-energy transfer space).

The main problems that a fixed binning algorithm presents are, first, outliers may dominate the final presentation, also, skewed data is not well handled, and finally: correlation exist, when we try to smooth it, even if we use the most modern adaptive kernel smoothing algorithms.

Both versions, 2D and 3D of the multiresolution algorithm, show information about the intensity and its correspondent error-levels simultaneously without messing up the resulting image. The size of the pixels is used in a convenient way to stand out the high-low statistics areas. Presenting high statistics areas with higher resolution and poor error-intensity relation zones with lower resolution, we not only *filter* the image but also show the *accuracy* of the displayed intensity, in principle we can guarantee that the showed information has a minimum level of accuracy.

We built a necessary volume model, specifically thought for the neutron scattering data, which will enable the application of the different volume visualization techniques, opening a gate to the application of the latest techniques to this scientific visualization field. It is important to

notice that practically all visualization tools require some type of volume model for their application, and therefore a correct implementation of a volume modelling will allow the implementation of the different visualization techniques that are already available for the rest of the scientific areas, such as medical image processing, earth sciences and so on [15, 16].

Specifically the generated volume data sets were visualized in terms of separate slices, not having a true 3D representation of the data, limiting the possibilities of the user, the mind can glean information from animation that is virtually impossible to obtain by separately viewing still images [17]. The impact of iteration, which is another dimension of scientific visualization, allows the researcher, to cooperate with the computer by directing the computations, increasing the chances for inspiration, insight, and understanding.

References

- [1] S. M. Hayden, H. A. Mook, Pengcheng Dai, T. G. Perring and F. Dogan, The structure of the high-energy spin excitations in a high-transition-temperature superconductor, *NATURE*, 03 June 2004, Vol 429 No 6991 pp531-534.
- [2] J. M. Tranquada, H. Woo, T. G. Perring, H. Goka, G. D. Gu, G. Xu, M. Fujita and K. Yamada, Quantum magnetic excitations from stripes in copper oxide superconductors, *NATURE*, 03 June 2004, Vol 429 No 6991 pp534-538.
- [3] P.K. Robertson and L. de Ferrari, Systematic approaches to visualization: is a reference model needed? in L. Rosembaum et al, *Scientific Visualization-Advances and Challenges*, (London: Academic Press 1994) 287-305.
- [4] R.A. Earnshaw and M. Jern, Fundamental approaches to interactive real-time visualization systems in L. Rosembaum et al, *Scientific Visualization-Advances and Challenges*, (London: Academic Press 1994) 223-238.
- [5] Sanders JS, Fabian AC, Adaptive binning of X-ray galaxy cluster images, *MON NOT R ASTRON SOC* JUL 21 2001, 325 (1): 178-186.
- [6] Cappellari M, Copin Y, Adaptive spatial binning of integral-field spectroscopic data using Voronoi tessellations, *MON NOT R ASTRON SOC* JUN 21 2003, 342 (2): 345-354.
- [7] Wong PC, Visual data Mining, *IEEE Computer Graphics and Applications*, vol 19, no. 5, Sept/Oct 1999, 20-21.
- [8] W. Hibbard, H. Levkowitz, J. Haswell, P. Rheingans, and F. Schroeder, Interaction in Perceptually-Based Visualization, *Perceptual Issues in Visualization*, G.G. Grinstein and H. Levkowitz, eds., Springer-Verlag, 1995, pp. 23-32.
- [9] S.K. Card, J.D. Mackinlay, and B. Shneiderman, Using Vision to Think, *Readings in Information Visualization eds.*, San Francisco, USA, 1999.
- [10] L.Gregory J.Kittler, Using Contextual Information for Image Retrieval, *11th International Conference on Image Analysis and Processing*, Palermo, Italy, September 26 - 28, 2001.
- [11] Radu Coldea, ISIS / Rutherford Appleton Laboratory, *Mslice*, <http://www.isis.rl.ac.uk/excitations/mslice/>, May 2001.
- [12] R.A.Cowley, ISIS Experimental Report Rutherford Appleton Laboratory, *Spin fluctuations in Rb₂MnF₄*, June 2002.
- [13] J.M. Hellerstein, R. Avnur, A. Chou, C. Hidber, C. Olston, V. Raman, T. Roth, and P.J. Haas, Interactive Data Analysis: The Control Project, *Computer*, vol. 32, no. 8, Aug. 1999, pp. 51-58.
- [14] B. Shneiderman, Dynamic Queries for Visual Information Seeking, *IEEE Software*, vol. 11, no. 6, 1994, pp. 70-77.
- [15] Min Chen and John V. Tucker, Constructive Volume Geometry, *Computer Graphics Forum*, volume 19, number 4, 2000, pages 281-293.
- [16] G.M. Nielson, Volume Modelling, In: M. Chen et al. (eds.). *Volume Graphics*, Springer, 2000, 29-48.
- [17] G.M. Nielson: Guest Editors Introduction: Visualization in Scientific Computing, *IEEE Computer* 22(8), (1989), 10-11.