

## Interactive Visualization of Multidimensional Coincidence Spectra

---

**Miroslav Morháč<sup>1</sup>**

*Institute of Physics, Slovak Academy of Sciences  
Dúbravská cesta 9, Bratislava, 845 11, Slovak republic  
E-mail: Miroslav.Morhac@savba.sk*

**Vladislav Matoušek**

*Institute of Physics, Slovak Academy of Sciences  
Dúbravská cesta 9, Bratislava, 845 11, Slovak republic  
E-mail: Vladislav.Matousek@savba.sk*

### Abstract

The paper presents direct visualization techniques of multidimensional nuclear spectra as well as visualization techniques based on projections of embedded subspaces. While the first group of graphical models is limited to four dimensions, the second one can be theoretically extended to any dimension. The presented algorithms of visualization have been implemented in nuclear data acquisition, processing and visualization system developed at the Institute of Physics, Slovak Academy of Sciences. The focuses on presentation of nuclear spectra. However the majority of algorithms can be successfully applied for visualization of scalar arrays of other data types.

*XI International Workshop on Advanced Computing and Analysis Techniques in Physics Research  
Amsterdam, the Netherlands  
23-27 April, 2007*

---

<sup>1</sup> Speaker

## 1. Introduction

The power of computers to collect, store and manipulate experimental data has increased dramatically. In today's nuclear physics experiments the number of detectors being included in the measurements is going up to one hundred or more. The results of such measurements, however, generate such large data sets as to be nearly incomprehensible. Scanning these large sets of numbers to determine trends and relationships is a tedious and ineffective process. To address this problem the physicists have turned to visualization of experimental data. If the data are converted to a visual form, the trends are often immediately apparent. Without visualization much of the increased power of computers would be wasted because experiments are poor at gaining insight from data presented in numerical form.

The goal of visualization of experimental data is an improved understanding of the result of the information gathered during experiment. It is one of the most powerful and direct ways how the huge amount of information can be conveyed in a form comprehensible to a human eye. As a result, development of visualization algorithms takes on great significance, offering a promising technology for transforming an indigestible mass of numbers into a medium, which experimenters can understand, interpret and explore.

The visualization techniques presented in this work make it possible to display either raw experimental spectra, processed data or to make slices of the same or lower dimensionality in an interactive way. They allow obtaining an imagination about event distribution and correlations in coincidence spectra up to five-dimensional space.

The objective of the paper is to present visualization techniques and graphical models to display multidimensional nuclear spectra (histograms), which were implemented in the nuclear data acquisition, processing and visualization system [1, 2]. Though the software package is designed mainly for use with nuclear data, any kind of data can be processed as well. Other systems for nuclear spectra graphical representation were presented in [3, 4].

## 2. Direct visualization techniques of scalar fields

A scalar variable is a single quantity, in the case of nuclear spectra - counts, which can be represented as a function of independent variables - particle energies. Most scalar visualization techniques use a consistent approach across one-, two-, or three-dimensional fields. More recent techniques, e.g. of the visualization of three-dimensional fields, attempt to show the full three-dimensional variations of a scalar variable within a volume field. These techniques include isosurfaces, particle clouds, volume slicing and sampling planes [5-7].

The sophisticated visualization algorithms are presented in [8]. The paper presents conventional as well as newly developed visualization techniques and graphical models. The structure and complexity of the algorithms lend themselves for implementation in on-line live mode during the data acquisition or processing. The pictures can be simultaneously updated.

One can select various attributes of the display, e.g. color of the spectrum, the limits of the displayed part of the spectrum, window, marker, type of scale, and various display modes,

slices, to rotate two-, or more-dimensional data. In the above-mentioned paper, we have developed the direct visualization algorithms up to four-dimensional data.

## 2.1 Two-dimensional spectra

Two-parameter coincidence nuclear spectrum (histogram) is represented by a matrix of data with two independent variables (parameters) and one dependent variable, (counts), i.e.,  $c = f(x, y)$ . To project the three-dimensional scene onto a two-dimensional screen the perspective transformation is employed. To display three-dimensional data on screen we have employed the following model

$$\begin{aligned} x' &= t_{xx} \cdot i + t_{xy} \cdot j + v_x \\ y' &= t_{yx} \cdot i + t_{yy} \cdot j + t_{yz} \cdot c + v_y, \end{aligned} \quad (1)$$

where  $t_{xx}, t_{xy}, t_{yx}, t_{yy}, t_{yz}, v_x, v_y$  are transform coefficients reflecting translation in both original two-dimensional scalar field (in  $x, y$  dimensions as well as in counts) and in the position on screen, scaling, rotation around  $z$ -axis and elevation of the view. The position of a point on the screen is  $x', y'$  and

$$y = y_{\min} + k_y \cdot j \quad x = x_{\min} + k_x \cdot i; \quad i \in \langle 0, n_x \rangle; \quad j \in \langle 0, n_y \rangle$$

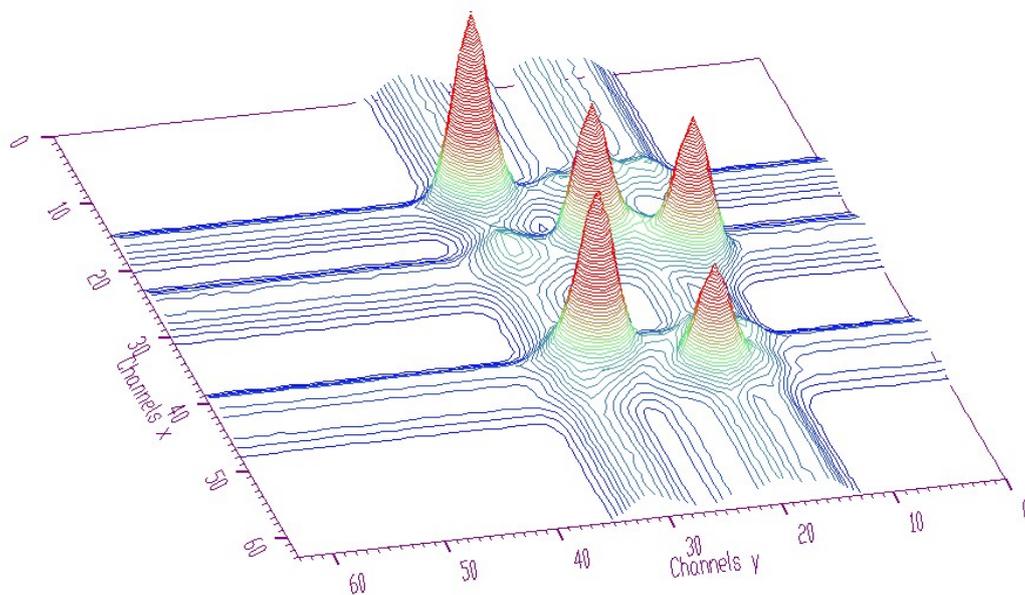
$$k_x = \frac{x_{\max} - x_{\min}}{n_x}; \quad k_y = \frac{y_{\max} - y_{\min}}{n_y}$$

$n_x, n_y$  are numbers of nodes of a regular grid. The model proposed in such a way allows:

- to choose and display any part of the spectrum by setting  $x_{\min}, x_{\max}, y_{\min}, y_{\max}$  to appropriate values
- to set any range of displayed counts –  $c_{\min}, c_{\max}$
- to place the display of spectrum anywhere on the screen
- to rotate and elevate the view of the spectrum
- to change the density of display nodes. This is important when displaying accumulated spectra in on-line mode, i.e., during the acquisition of spectra.

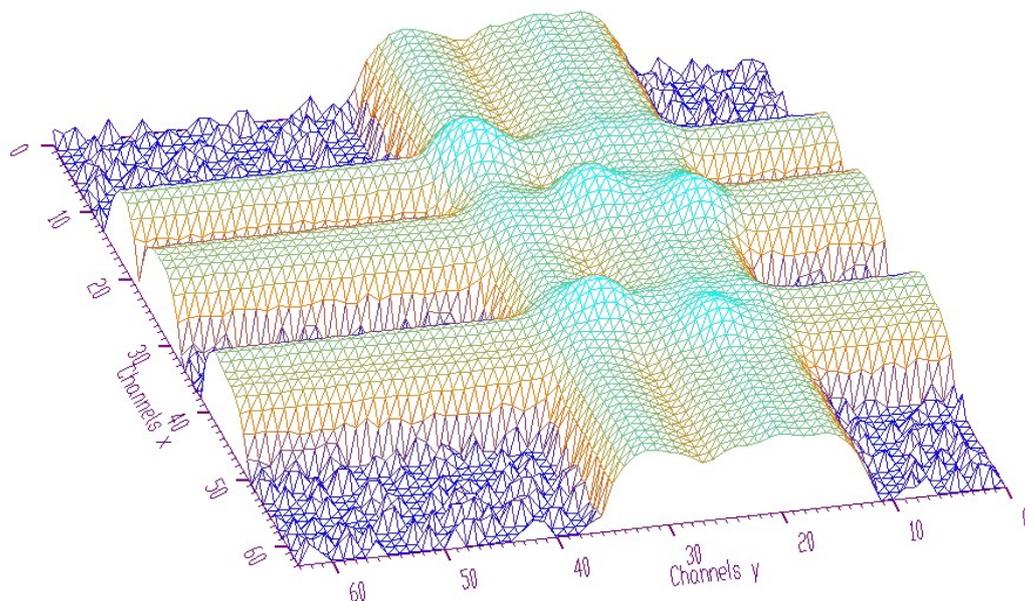
To illustrate the capabilities of the proposed visualization algorithms we introduce several examples. In Fig. 1 we present two-dimensional spectrum shown in contours display mode. The same spectrum shown in triangle display mode in log scale can be seen in Fig. 2. To identify interesting locations in spectra together with displayed spectrum one can display one-dimensional slices and to move with them in both directions (see Fig. 3).

Sophisticated surface display mode with shading according to heights of peaks is shown in Fig. 4. Shading according to the position of fictive light source is shown in Fig. 5. One can change the position of the light source thus giving the possibility to achieve special effects.

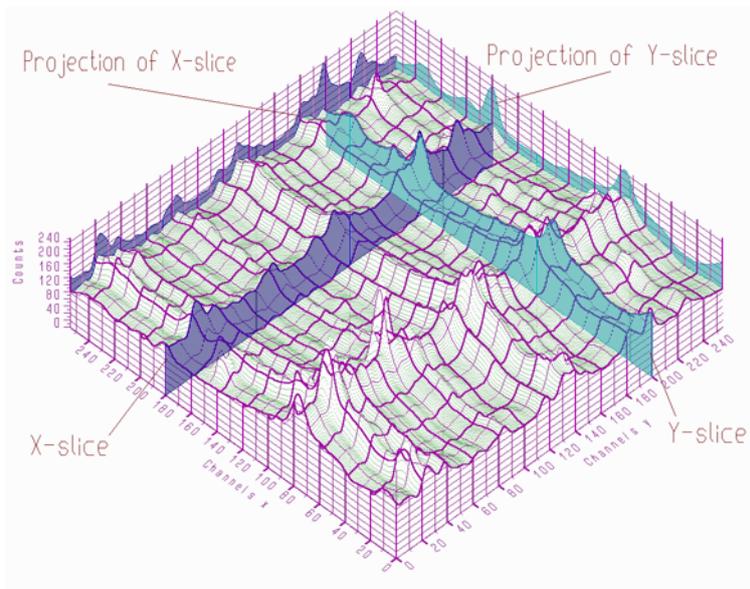


**Fig. 1 Two-dimensional spectrum shown in contours display mode**

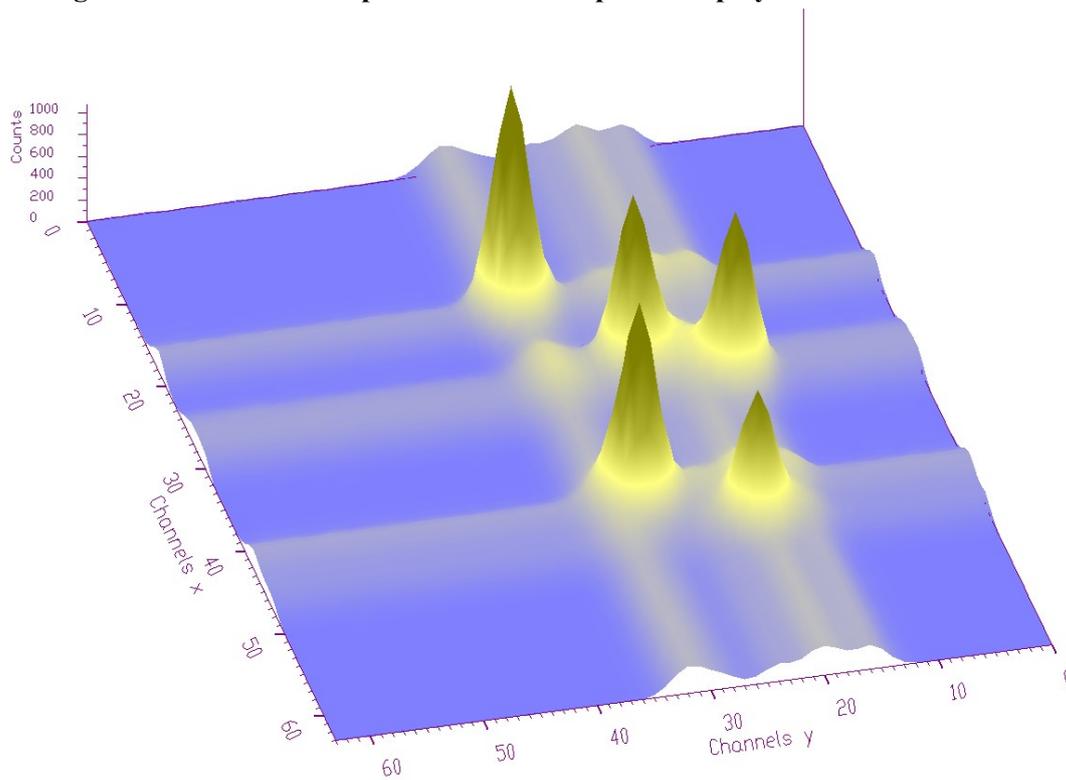
Furthermore, there exists possibility to combine both shading methods. In Fig. 6 we present the display mode with mixed surface shading (according to height and light position) with ratio 50:50. One can include also the display of shadows according to the light source (Fig. 7).



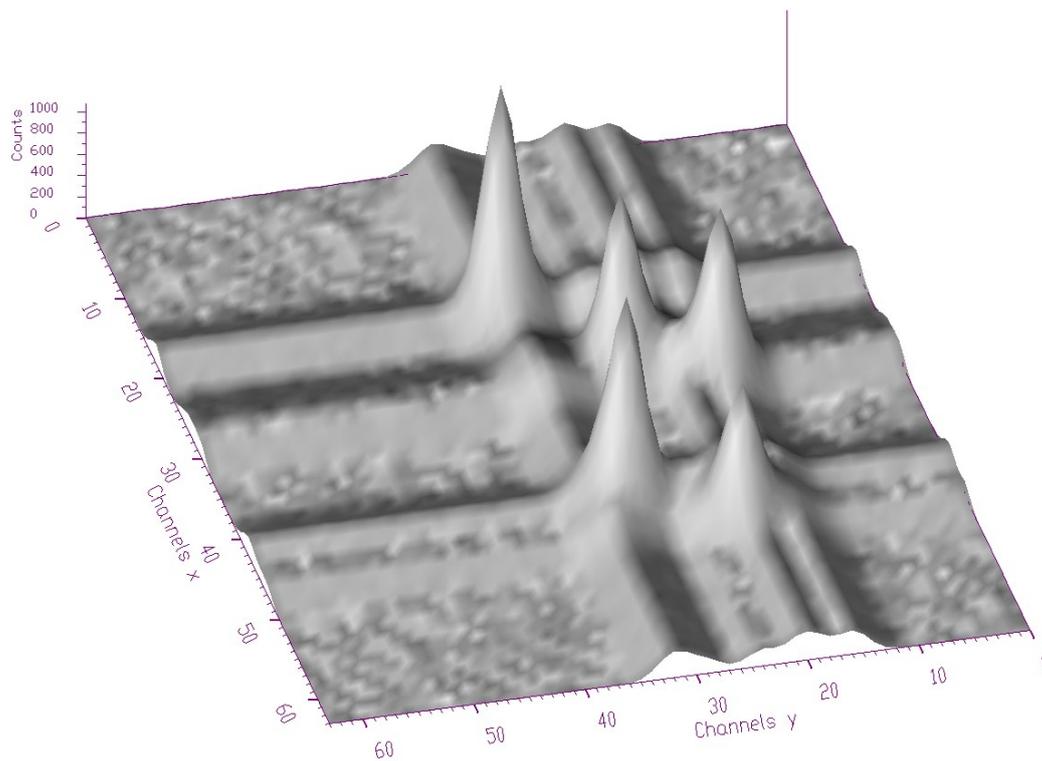
**Fig. 2 Two-dimensional spectrum shown in triangle display mode (log scale)**



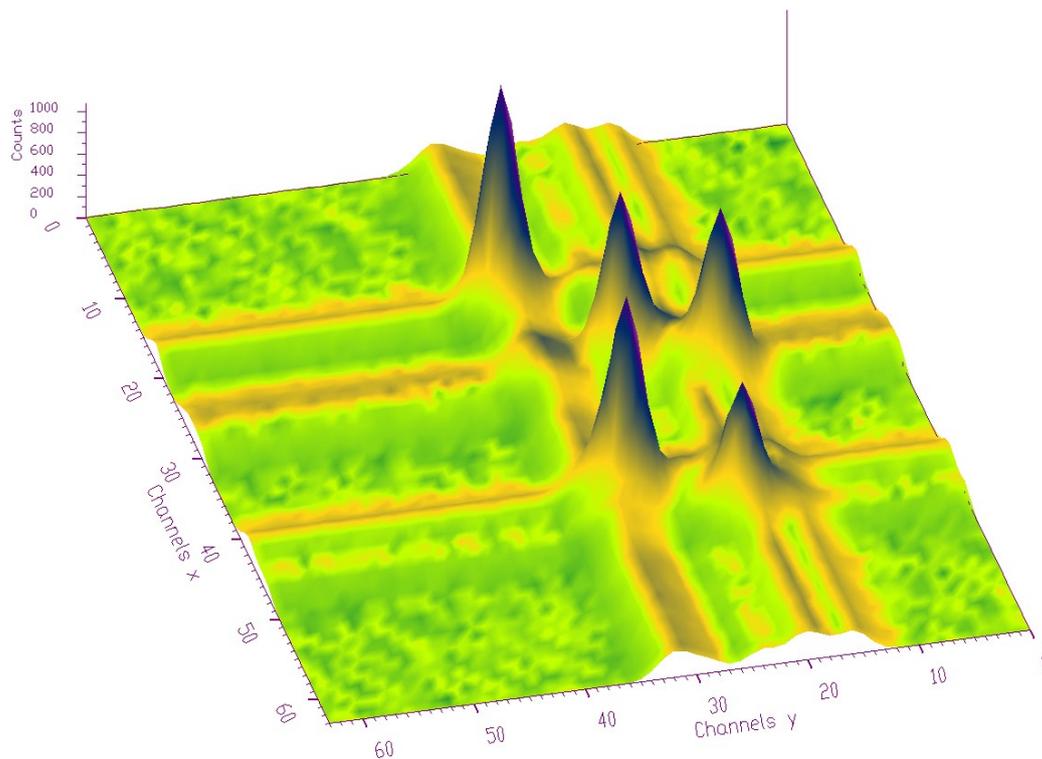
**Fig. 3** Two-dimensional spectrum shown in points display mode with raster and slices



**Fig. 4** An example of surface display mode with shading according to heights

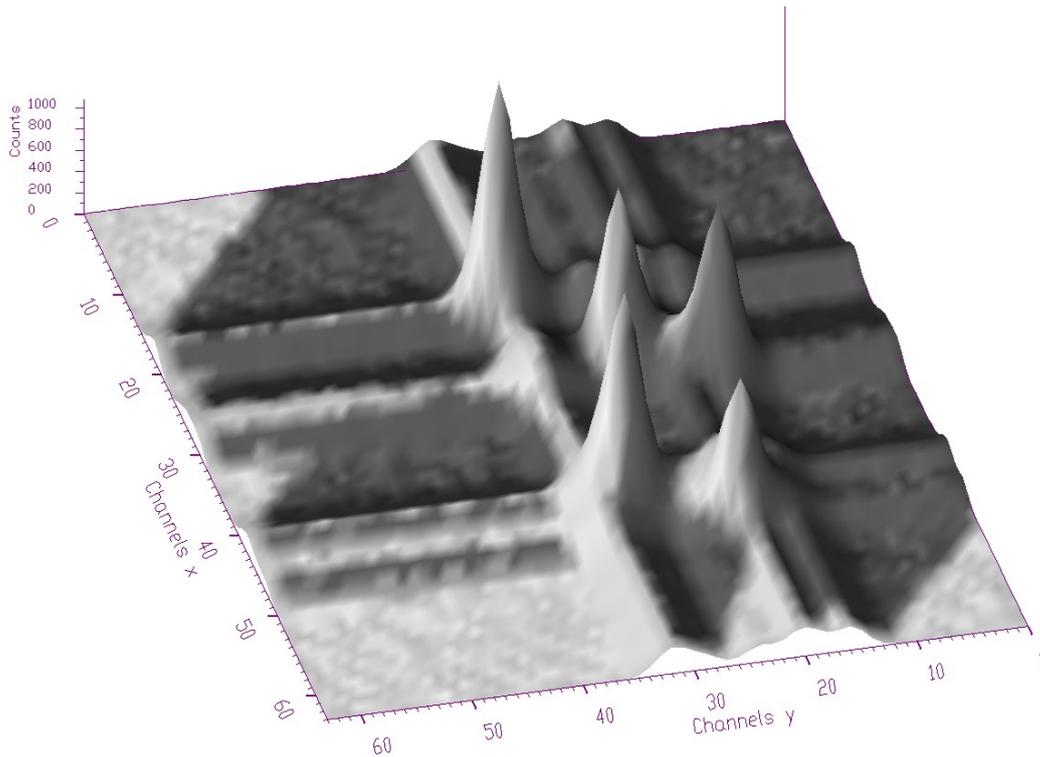


**Fig. 5** An example of surface display mode with shading according to fictive light source



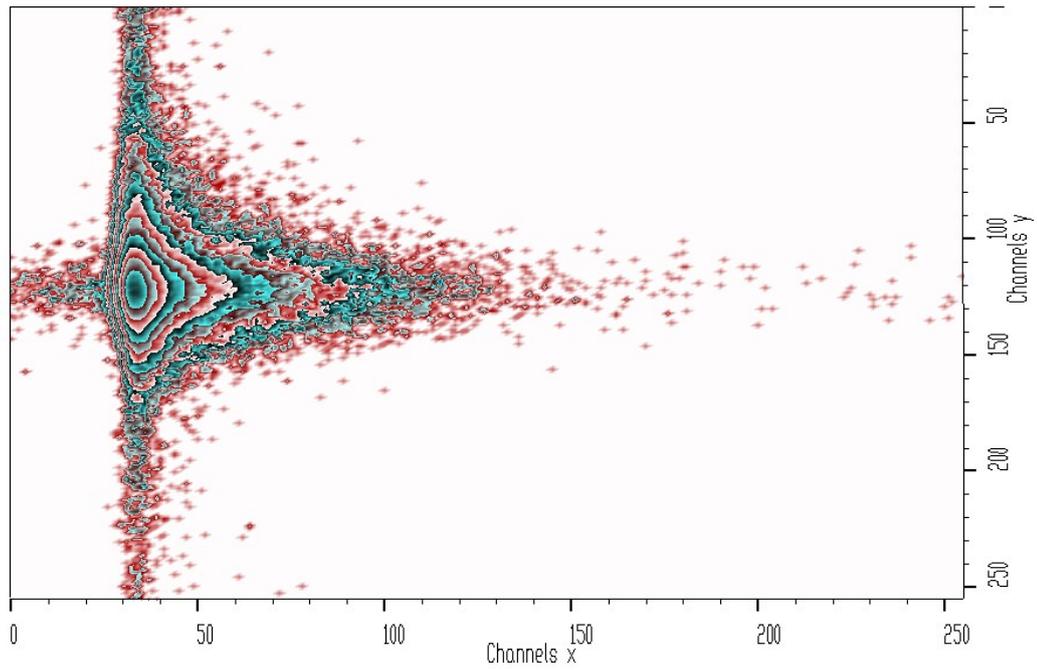
**Fig. 6** An example of surface display mode with combined shading algorithms (given in Fig. 4 and Fig. 5)

POS(AACAT)064

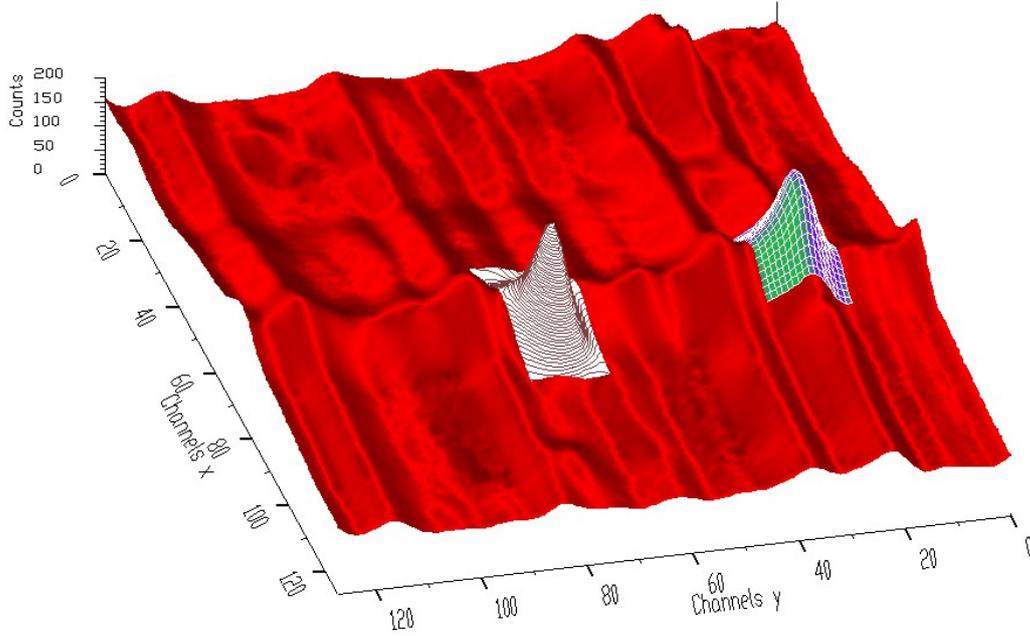


**Fig. 7** Surface display mode (like in Fig. 5) with shadows

All display parameters can be changed according to the needs of the experimenters. Informative way of the display is rectangular view with contour shading (positron annihilation spectrum) shown in Fig. 8. One can define Regions Of Interests (ROIs) in the spectrum. Every ROI has its own display parameters independent of the main spectrum and other ROIs. In Fig. 9, one can see two ROIs in the two-dimensional  $\gamma$ -ray spectrum displayed in different display modes and color shadings. Similarly, one can display also the peaks in the spectrum found in the process of peak identification (Fig. 10). There are many other display combinations possible. Their presentation however, goes beyond the scope of this work.



**Fig. 8** Rectangular view of positron annihilation spectrum with contour shading



**Fig. 9** Two-dimensional  $\gamma$ -X-ray spectrum with two ROIs

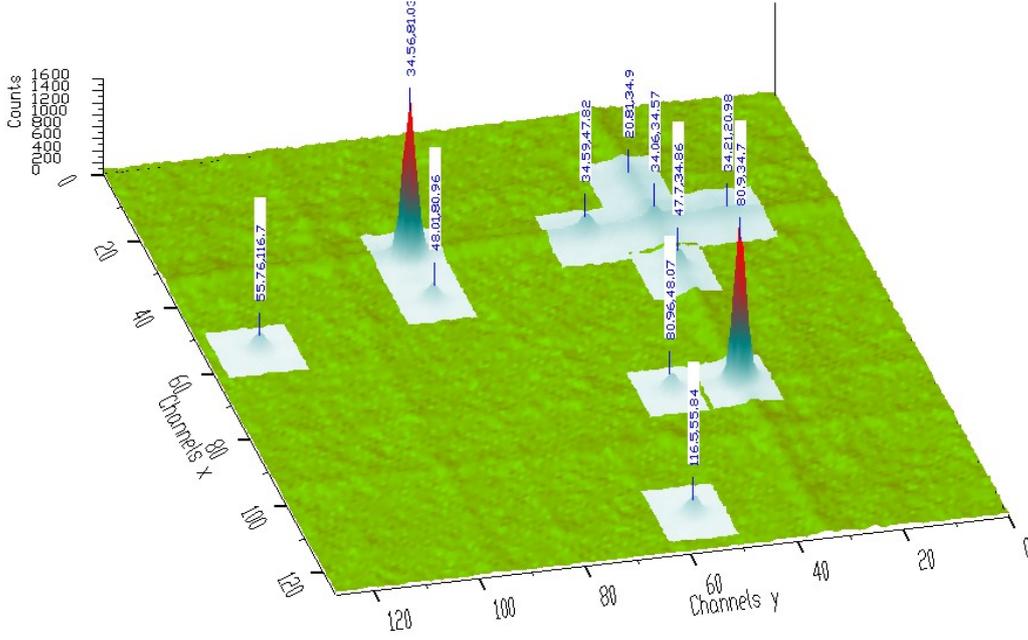


Fig. 10 Two-dimensional  $\gamma$ - $\gamma$ -coincidence spectrum with displayed peaks

## 2.2 Three-dimensional spectra

Analogously to the previous section three-parameter  $\gamma$ -ray coincidence nuclear spectrum is three-dimensional scalar field with three independent parameters  $x, y, z$  (particle energies) and one dependent variable – counts  $c = f(x, y, z)$ . As with two-parameter scalar fields one can idealize the display of three-parameter scalar field using discrete symbols at specific locations in space, or use techniques that show the variations in the three-parameter space. Hence each channel is defined by three parameters – coordinates  $x, y, z$  in original space, which determine the position of a channel. First let us consider a model where the channel is shown as a sphere (other marks as square, triangle, star etc. also can be used) with the size proportional to the event counts it contains. Then the position of the channel  $x, y, z$  on the screen is

$$\begin{aligned} x' &= t_{xx} \cdot i + t_{xy} \cdot j + t_{xz} \cdot k + v_x; \\ y' &= t_{yx} \cdot i + t_{yy} \cdot j + t_{yz} \cdot k + v_y, \end{aligned} \quad (2)$$

where  $t_{xx}, t_{xy}, t_{xz}, t_{yx}, t_{yy}, t_{yz}, v_x, v_y, v_z$  are display transform coefficients reflecting translations in both original three-dimensional scalar field (in  $x, y, z$  dimensions as well as in counts) and in the position on screen, scaling, rotation around axes  $x, y, z$  and

$$x = x_{\min} + k_x \cdot i; y = y_{\min} + k_y \cdot j; z = z_{\min} + k_z \cdot k,$$

where

$$k_x = \frac{x_{\max} - x_{\min}}{n_x}; k_y = \frac{y_{\max} - y_{\min}}{n_y}; k_z = \frac{z_{\max} - z_{\min}}{n_z};$$

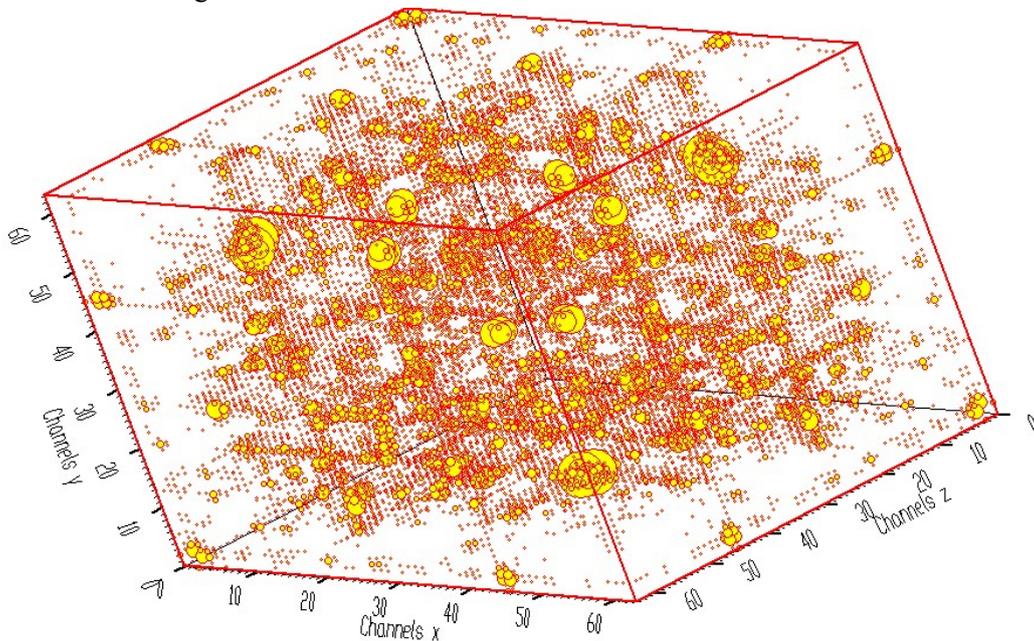
$n_x, n_y, n_z$  are numbers of nodes of regular grid. The model proposed in such a way allows:

- to choose and display any part of three-dimensional array -  $x_{\min}, x_{\max}, y_{\min}, y_{\max}, z_{\min}, z_{\max}$
- to choose any range of displayed counts –  $c_{\min}, c_{\max}$
- to place the picture anywhere on the screen
- to rotate the spectrum around the axes  $x, y, z$
- to change the density of display nodes.

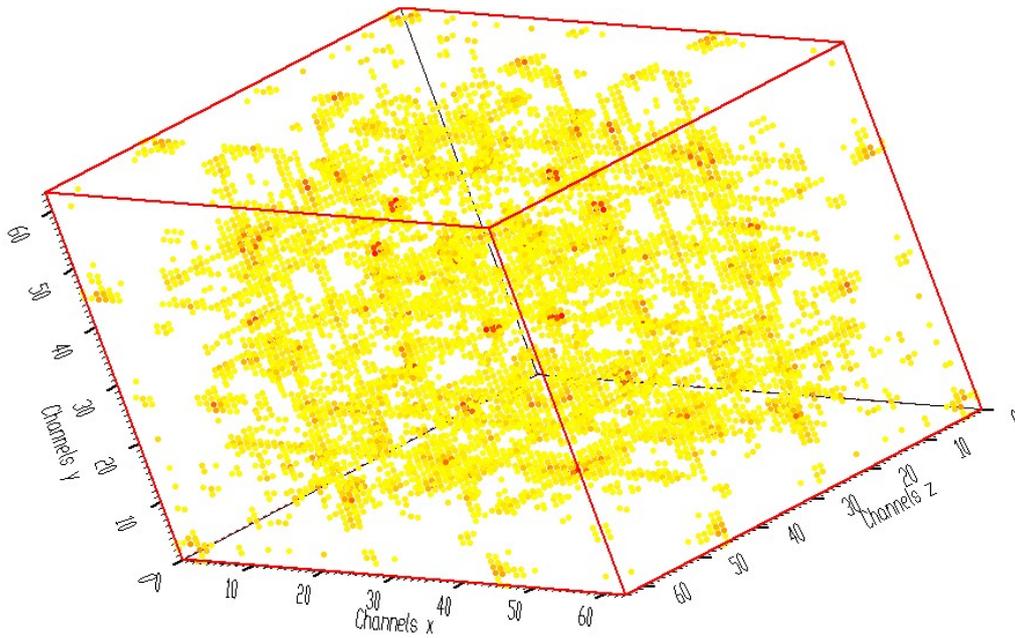
The particle gradient display modes where the channels are shown as spheres with either diameter or color proportional to their contents are shown in Fig. 11 and Fig. 12, respectively. From the figures, one can localize interesting parts (peaks) in the spectrum. Sometimes however, to identify these parts, it is preferable to display only a slice in the spectrum, to move with it and interactively find appropriate channels. One- and two-dimensional slice in three-dimensional spectrum is shown in Fig. 13 and 14, respectively.

One can use even more sophisticated surface display mode. The defined surface separates the events with higher counts (statistics) from those with lower counts. Moreover, to achieve smooth surface one can interpolate the three-dimensional space using B-spline technique. In Fig. 15 and Fig. 16, we see the three-dimensional  $\gamma$ -ray coincidence spectrum and positron annihilation spectrum, respectively.

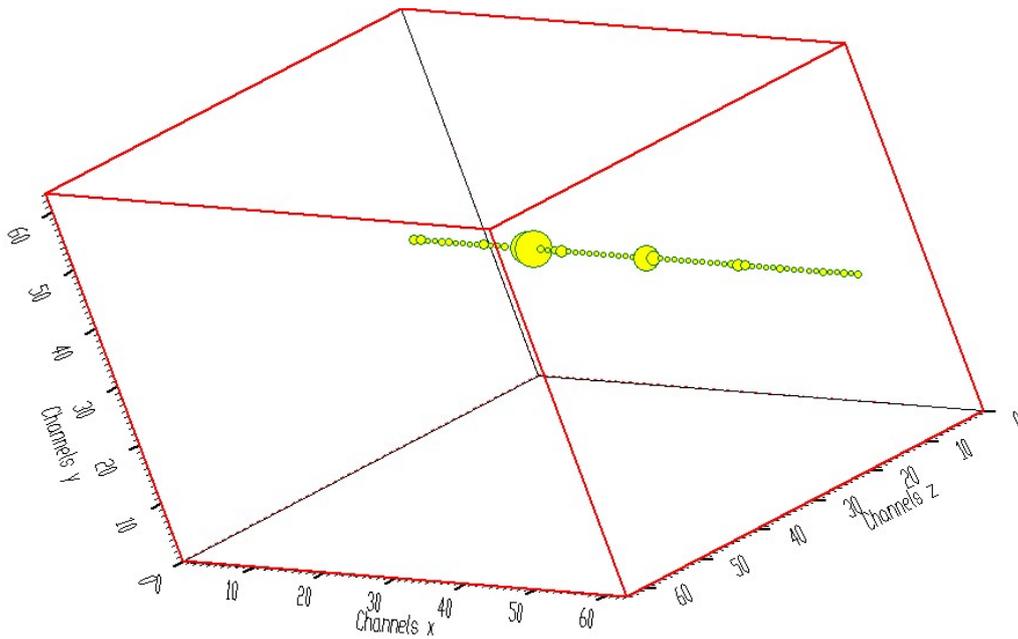
Finally, there exists a possibility to display three-dimensional spectrum in volume rendering mode. From the color contours on the sides of cube, one can get an imagination about positions of interesting peaks in three-dimensional space. An example of such a display mode is given in Fig. 17. Again, one can employ interpolation of the three-dimensional space using B-splines of various degrees.



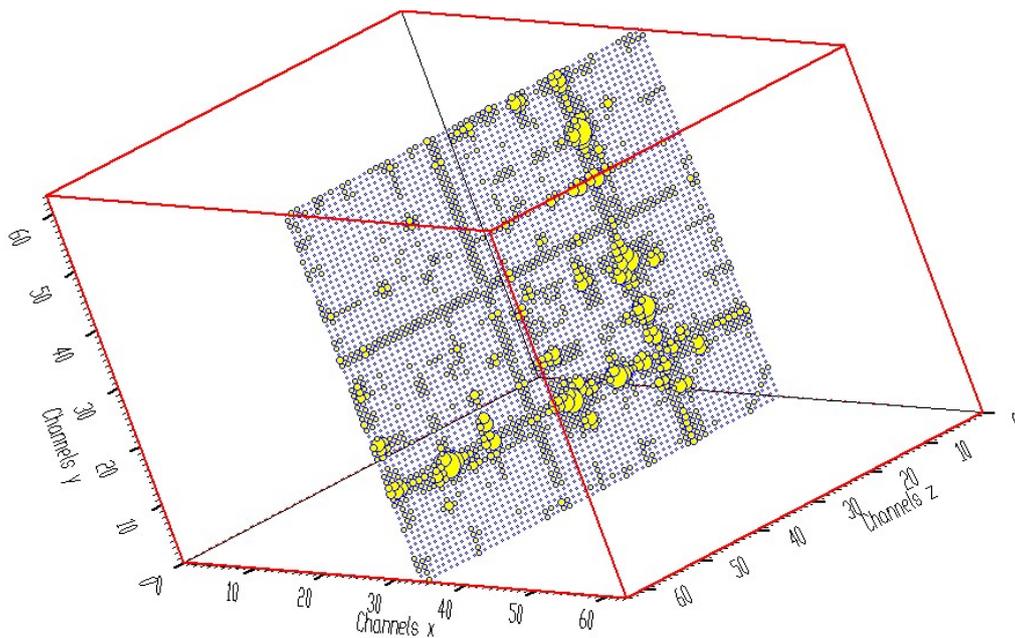
**Fig. 11 Three-dimensional  $\gamma - \gamma - \gamma$ -coincidence spectrum with channels shown as spheres with diameters proportional to counts**



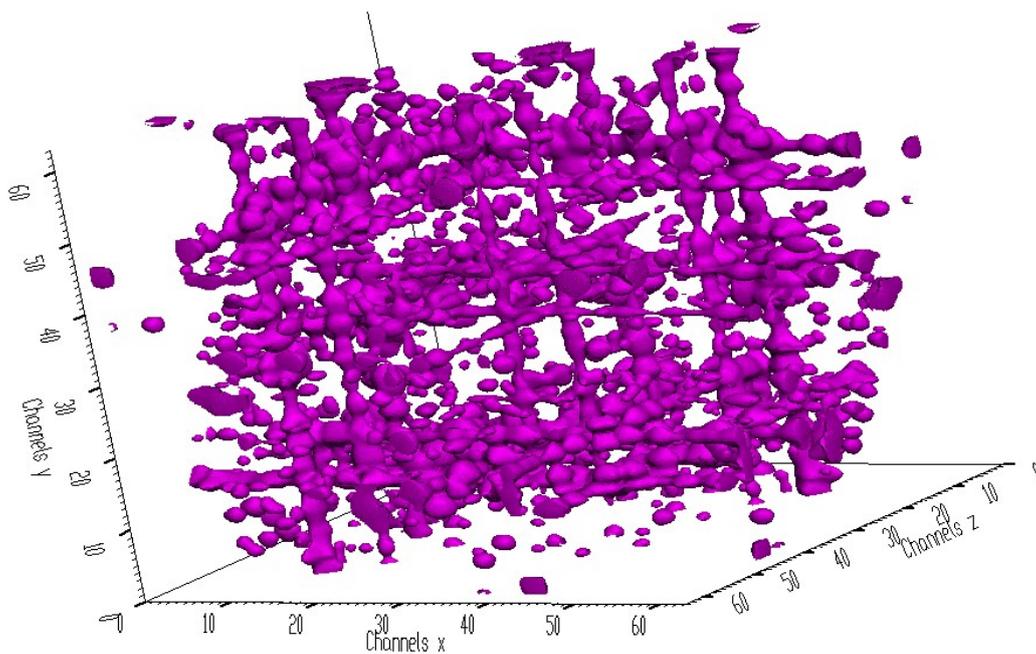
**Fig. 12** Three-dimensional  $\gamma-\gamma-\gamma$ -coincidence spectrum with channels shown as spheres with colors proportional to counts



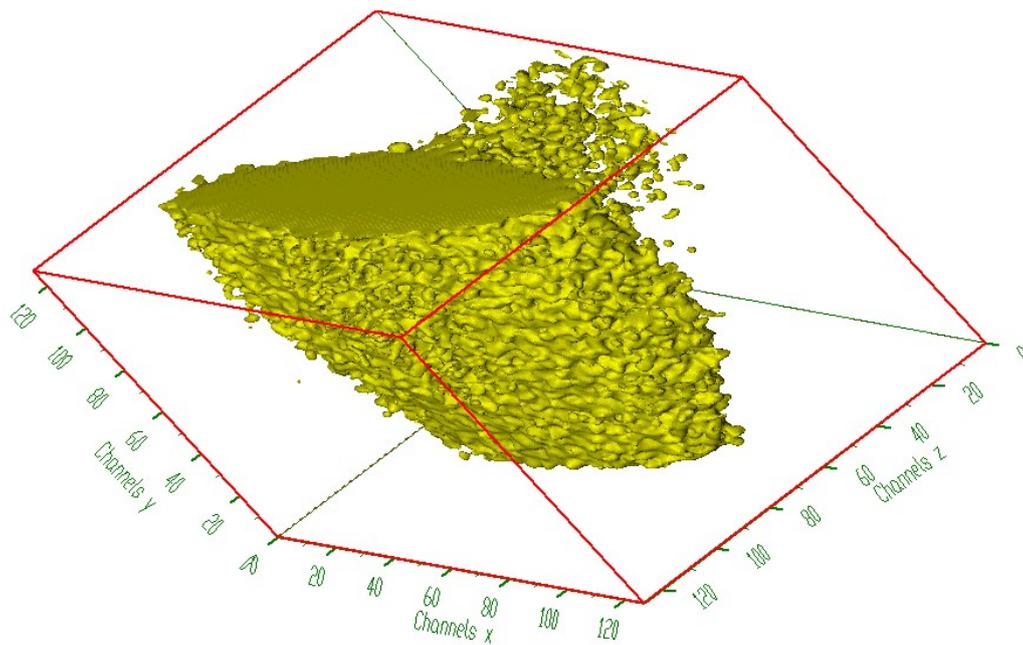
**Fig. 13** One-dimensional slice in three-dimensional spectrum



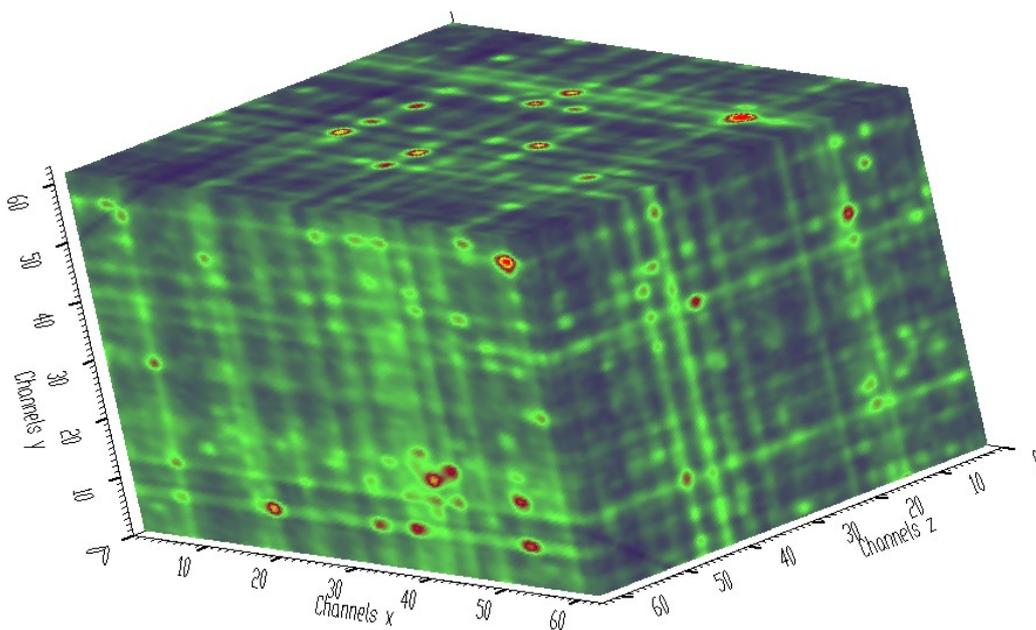
**Fig. 14** Two-dimensional slice in three-dimensional spectrum



**Fig. 15** Three-dimensional  $\gamma-\gamma-\gamma$ -ray coincidence spectrum shown in smoothed surface display mode



**Fig. 16** Three-dimensional positron annihilation spectrum shown in smoothed surface display mode



**Fig. 17** Three-dimensional  $\gamma$ - $\gamma$ - $\gamma$ -ray coincidence spectrum shown in volume rendering mode

### 2.3 Four-dimensional spectra

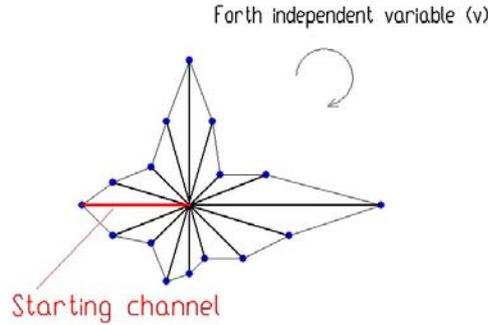
Now the counts is a function of four parameters (particle energies), i.e.,  $c = f(x, y, z, v)$ . Let us imagine that instead of one channel belonging to one point of 3-D space in three-parameter nuclear spectrum visualization now this point represents a slice in the fourth parameter, i.e.,  $c_{i,j,k}(v) = f(x_i, y_j, z_k, v)$ . We depict each slice as a closed polygon with the center positioned in analogy with three-dimensional data at the location

$$\begin{aligned} x'_{i,j,k} &= t_{xx} \cdot i + t_{xy} \cdot j + t_{xz} \cdot k + v_x \\ y'_{i,j,k} &= t_{yx} \cdot i + t_{yy} \cdot j + t_{yz} \cdot k + v_y. \end{aligned} \quad (3)$$

For the positions of the vertices of the polygon  $i, j, k$  on screen we have

$$\begin{aligned} x''_{i,j,k}(v) &= x'_{i,j,k} - r_{\max} \cdot \frac{c_{i,j,k}(v) - c_{\min}}{c_{\max} - c_{\min}} \cdot \cos\left(\frac{2\pi(v - v_{\min})}{v_{\max} - v_{\min} + 1} + \varphi_0\right), \\ y''_{i,j,k}(v) &= y'_{i,j,k} + r_{\max} \cdot \frac{c_{i,j,k}(v) - c_{\min}}{c_{\max} - c_{\min}} \cdot \sin\left(\frac{2\pi(v - v_{\min})}{v_{\max} - v_{\min} + 1} + \varphi_0\right), \end{aligned} \quad (4)$$

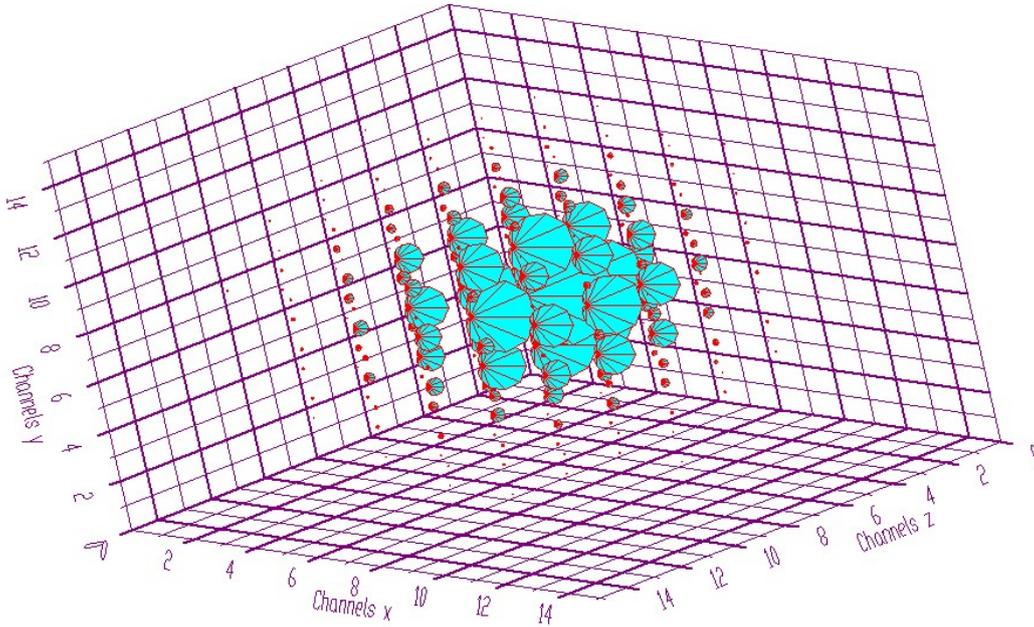
where the fourth parameter  $v \in \langle v_{\min}, v_{\max} \rangle$ ,  $r_{\max}$  (constant value) is maximum distance of a polygon vertex from its center,  $\varphi_0$  is starting angle of the display of the first vertex of the polygon and  $c_{\min}, c_{\max}$  determine the range of displayed counts. The principle of 4D display is illustrated in Fig. 18.



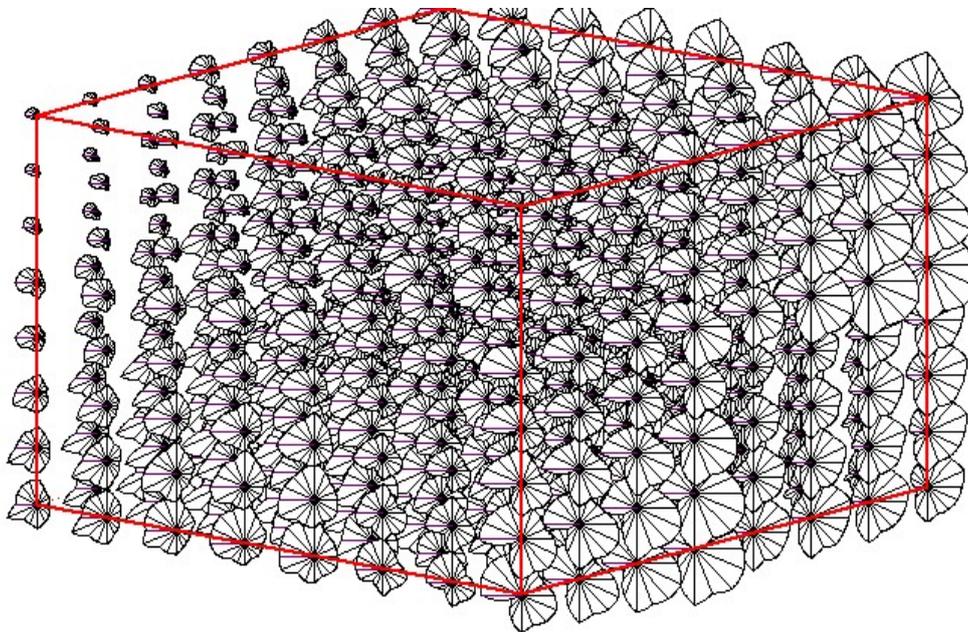
**Fig. 18 Principle of 4D display**

In Fig. 19 we show four-dimensional display of the synthetic Gaussian with the center at  $x = y = z = v = 8$ . Three parameters determine the position of the center of the slice. The channels of the slice are shown as bars (drawn in red color) starting in the center of the slice with lengths proportional to their contents. The channels are displayed starting from 9 o'clock position in clockwise direction. The example of a chunk (16x16x16x16 channels) of smoothed

experimental  $\gamma-\gamma-\gamma-\gamma$ -ray spectrum through the use of this display algorithm is shown in Fig. 20.

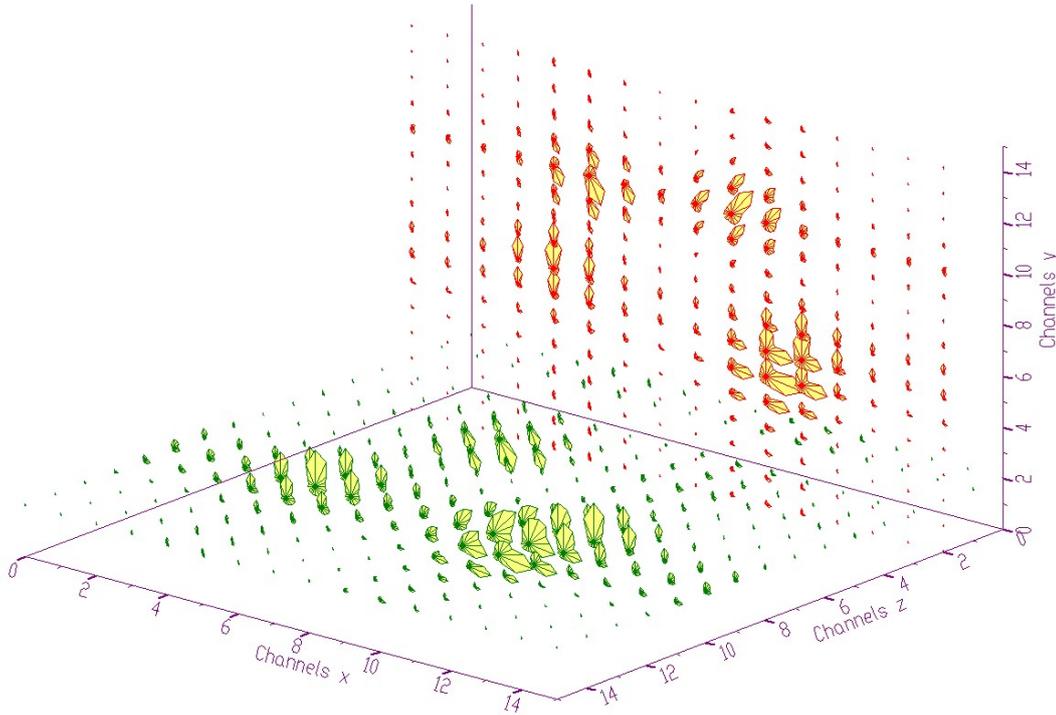


**Fig. 19 Display of four-dimensional synthetic Gaussian**

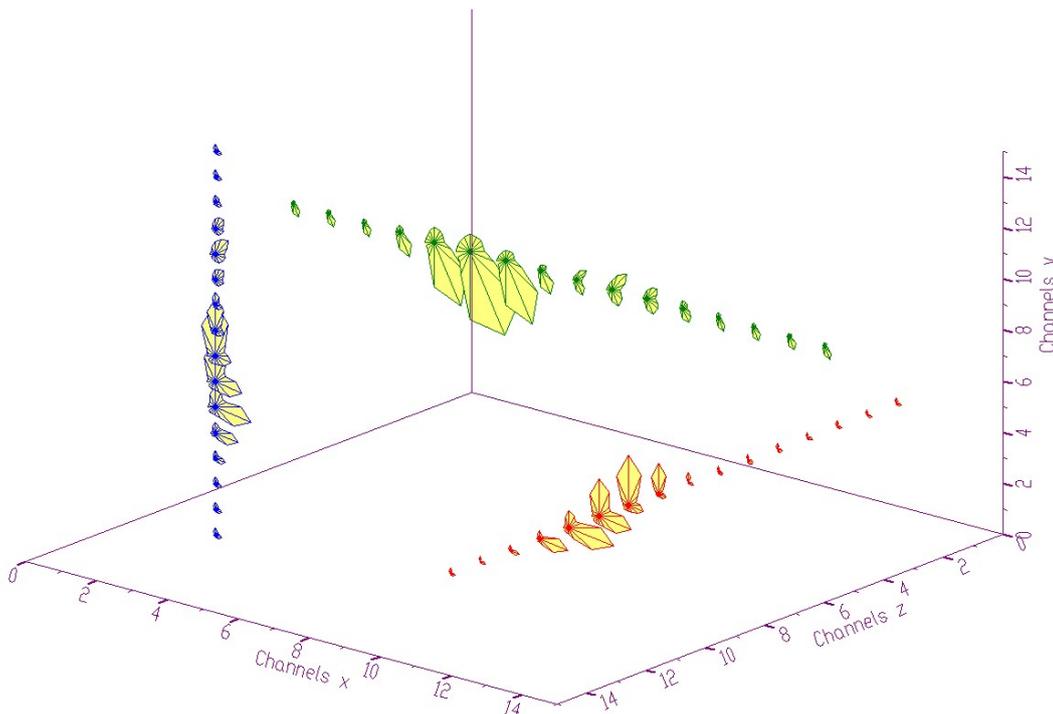


**Fig. 20 Smoothed experimental four-parameter spectrum**

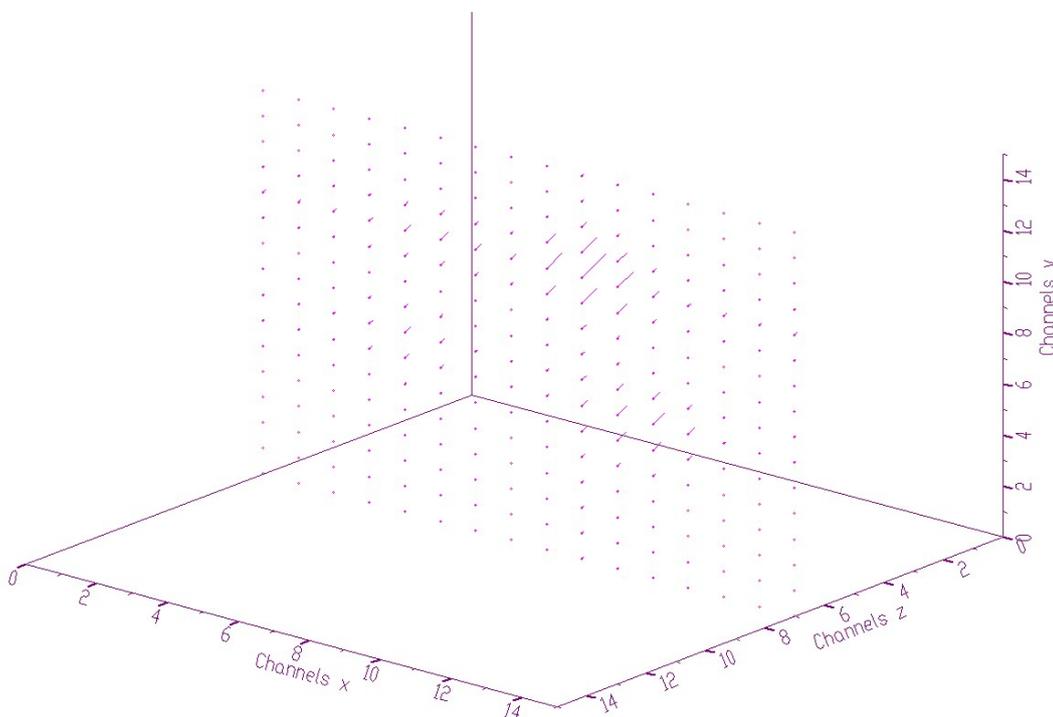
Analogously to 2 and 3D data to find interesting parts of the spectra one can display slices of various dimensionality. In Fig. 21 we present 3D slices with fixed  $y$  and  $z$  variables, respectively. Changing the values of  $y$  and  $z$  one can move with the slices. Subsequently in Fig. 22 we give an example of three two-dimensional slices in four-dimensional spectrum with fixed  $xy$ ,  $xz$  and  $yz$  variables. In Fig. 23 we illustrate similar situation when we fix the fourth variable. We show the two-dimensional slice with fixed  $zv$  variables. Finally in Fig. 24 we introduce the display of three one-dimensional slices with fixed  $xyv$ ,  $xzv$  and  $yzv$  variables, respectively.



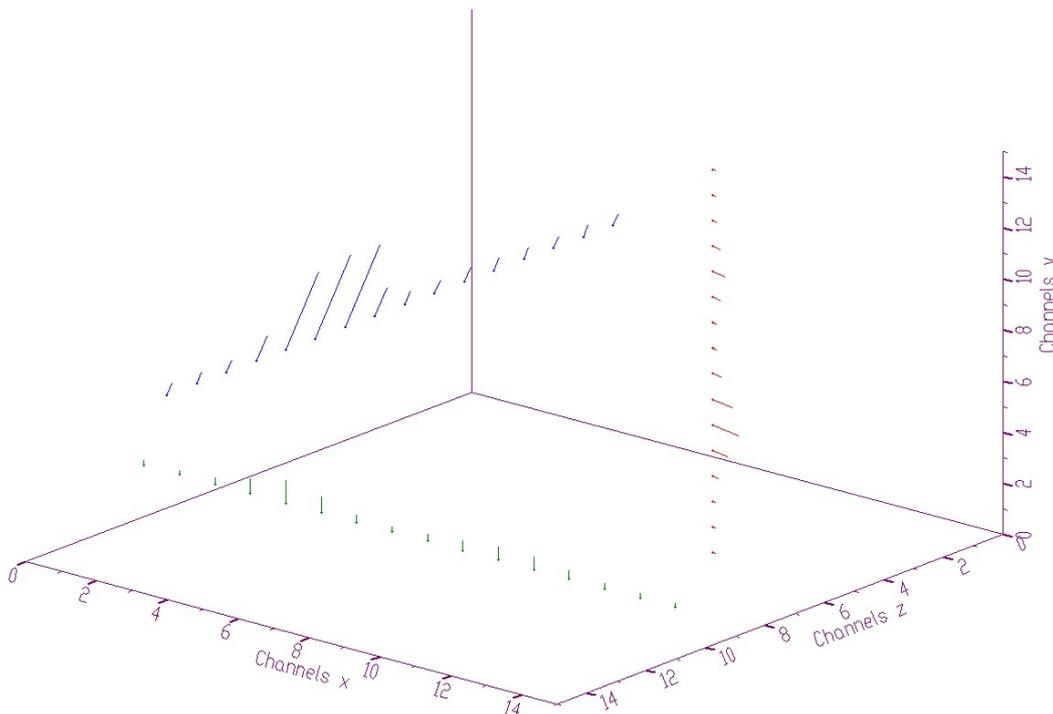
**Fig. 21** Two three-dimensional slices in four-dimensional spectrum



**Fig. 22 Three two-dimensional slices in four-dimensional spectrum with fixed xy, xz, yz variables**

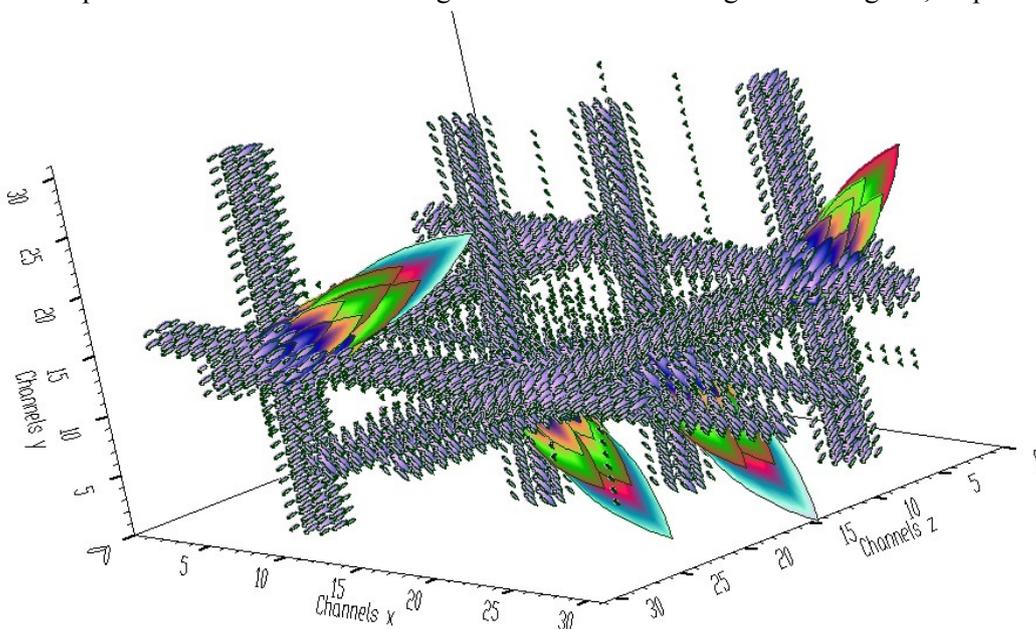


**Fig. 23 Two-dimensional slice in four-dimensional spectrum with fixed zv variables**

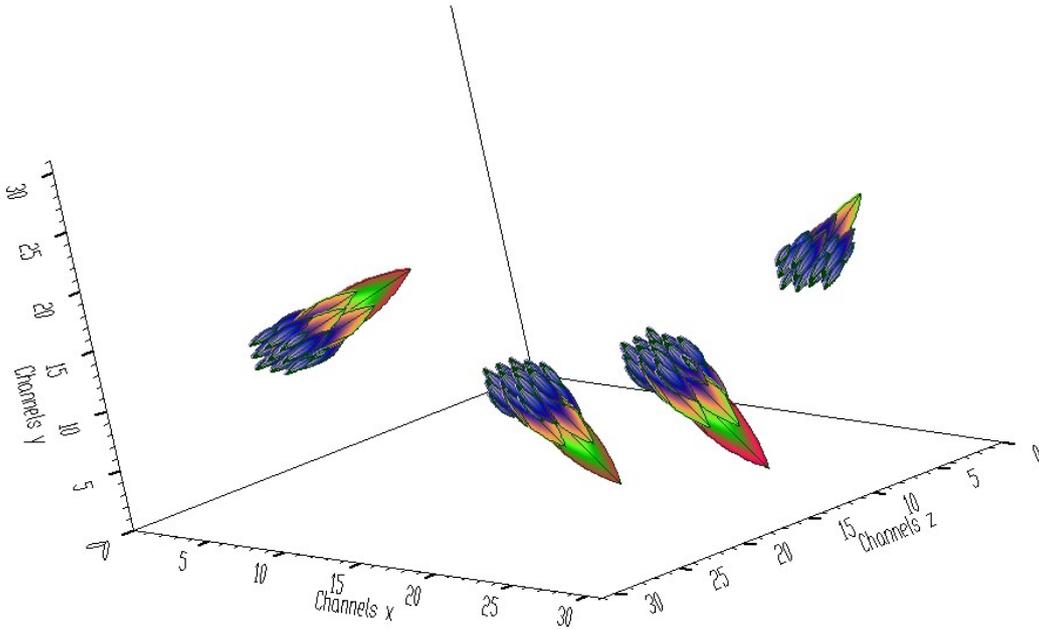


**Fig. 24 Three one-dimensional slices in four-dimensional spectrum with fixed  $xyv$ ,  $xzv$ ,  $yzv$  variables**

To illustrate the display of ridges in four-dimensional space we show the examples of synthetic spectrum before and after background elimination in Fig. 25 and Fig. 26, respectively.

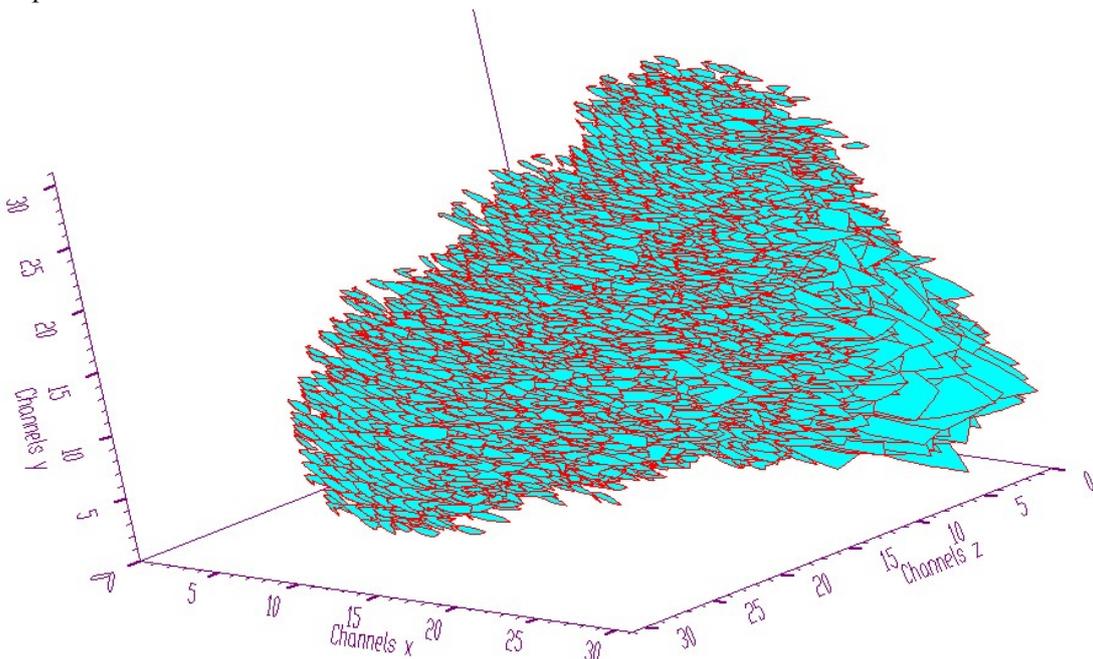


**Fig. 25 Synthetic four-dimensional spectrum before background elimination**



**Fig. 26 Four-dimensional peaks after background elimination from the data from Fig. 25**

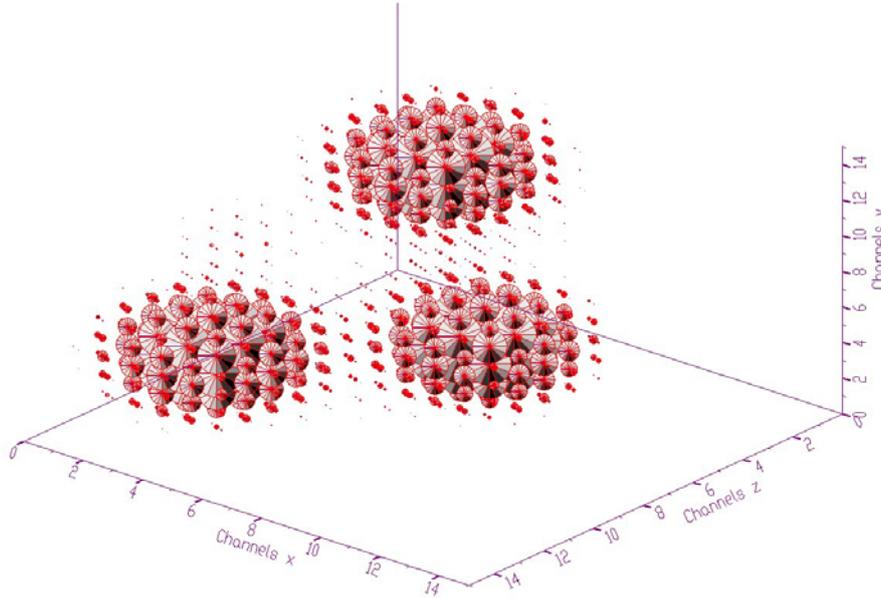
In Fig. 27, we present four-fold coincidence positron annihilation spectrum with interpolated channels.



**Fig. 27 Four-fold coincidence positron annihilation spectrum**

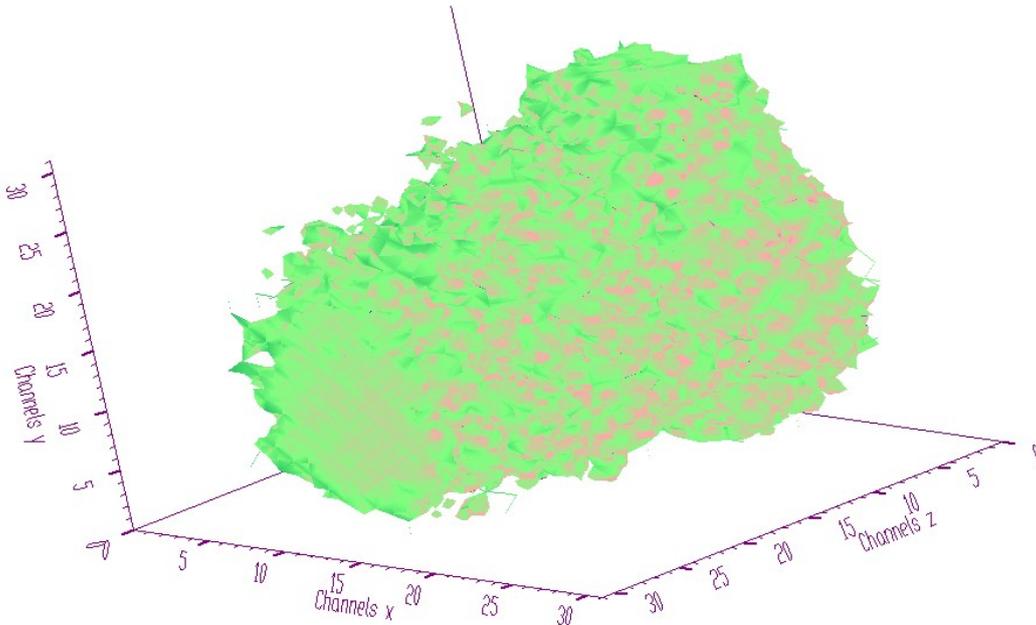
In pies display mode one can change the color (level of shading) while keeping the radius of circle constant. According to the resolution in the fourth independent variable the circle is divided to channels (pies) with colors proportional to the contents of channels. The size of the

circle is proportional to the sum of counts in the fourth dimension  $\sum_{v=y_{\min}}^{y_{\max}} f(x_i, y_j, z_k, v)$ . An example of synthetic 4D spectrum in pies display mode is shown in Fig. 28.



**Fig. 28 Four-dimensional synthetic spectrum displayed in pies display mode**

Finally, in Fig. 29 we present four-dimensional spectrum in isosurface mode. Analogously to three-dimensional data, the surface separates the channels with higher counts from those with lower counts. In this case, however, the color of the surface is defined by the position of the first occurrence of the channel with the same or higher value than the defined boundary value.



**Fig. 29 Four-dimensional spectrum shown in isosurface display mode**

### 3. Technique of successive projections of embedded subspaces

The dimensionality of above-presented visualization techniques is limited to four. However, with increasing dimensionality of nuclear spectra the requirements in developing of multidimensional scalar visualization techniques becomes striking. In principle, the above-mentioned algorithms can be used even for higher dimensions by employing technique of embedded subspaces. The goal is to propose a technique that allows one to localize and scan interesting parts (peaks) in multidimensional spectra. Moreover it should permit to find correlations in the data, mainly among neighboring points, and thus to discover prevailing trends around multidimensional peaks.

The proposed technique makes benefit of specific character and features of nuclear spectra. It utilizes the fact that the interesting objects (peaks) have shape of quasi Gaussians. Further, in enormous multidimensional space the events are distributed very sparsely, which allows to preserve main features of data even after reducing the dimensionality by employing projection functional. Successive decreasing the dimensionality makes it possible to determine the positions of appropriate multidimensional peaks.

Without loss of generality, we shall assume the reduction of the space up to two-dimensional one. Other alternatives are also possible, but the display of two-dimensional array using perpendicular view allows utilizing screen area the most efficiently. Let us start with three-dimensional spectrum  $f(x, y, z)$ . Let us apply a projection functional reducing dimensionality by one to two-dimensional array, e.g.  $f^{(1)}(x, y) = F[f(x, y, z)]$ . In place of the functional one can use, e.g. sum of channels contents in a slice

$$f^{(1)}(x, y) = \sum_{z=z_{\min}}^{z_{\max}} f(x, y, z),$$

or maximum in a slice

$$f^{(1)}(x, y) = \max \{f(x, y, z)\},$$

where  $z \in \langle z_{\min}, z_{\max} \rangle$  or any other suitable operation. Let us display each channel  $i, j$  in the form of a mark with size proportional to  $f^{(1)}(i, j)$ . Again, because of the most efficient way of utilizing the screen, in place of the mark we choose a rectangle. The rectangle represents a “window” into the subspace. Inside of the rectangle, we can display the slice  $f(i, j, z)$ ,  $z \in \langle z_{\min}, z_{\max} \rangle$ . From the distribution of rectangles, one can find out the positions of interesting peaks, then focus the view to an appropriate region or to zoom a slice to full screen size, respectively.

Let us proceed to four-dimensional data  $f(x_1, x_2, x_3, x_4)$ . In place of the functional, we shall use the sums of channels in appropriate two-dimensional regions

$$f^{(1)}(x_1, x_2) = \sum_{x_3=x_{3\min}}^{x_{3\max}} \sum_{x_4=x_{4\min}}^{x_{4\max}} f(x_1, x_2, x_3, x_4). \quad (5)$$

Then inside of each rectangle belonging to the channel  $i_1, i_2$  we display two-dimensional slice  $f(i_1, i_2, x_3, x_4)$ , using any of the two-dimensional above presented graphical models.

In the case of five-dimensional spectrum, we can apply one-, or two-step reduction of dimensionality, i.e., either

$$f^{(1)}(x_1, x_2) = \sum_{x_3=x_3 \min}^{x_3 \max} \sum_{x_4=x_4 \min}^{x_4 \max} \sum_{x_5=x_5 \min}^{x_5 \max} f(x_1, x_2, x_3, x_4, x_5) \quad (6)$$

or

$$f^{(1)}(x_1, x_2, x_3, x_4) = \sum_{x_5=x_5 \min}^{x_5 \max} f(x_1, x_2, x_3, x_4, x_5) \quad (7)$$

$$f^{(2)}(x_1, x_2) = \sum_{x_3=x_3 \min}^{x_3 \max} \sum_{x_4=x_4 \min}^{x_4 \max} f^{(1)}(x_1, x_2, x_3, x_4). \quad (8)$$

In the first case in each rectangle window belonging to the channel  $i_1, i_2$  one can display three-dimensional slice  $f(i_1, i_2, x_3, x_4, x_5)$  using any of the three-dimensional graphical models. In the second one, in each rectangle belonging to the channel  $i_1, i_2$  one can display two-dimensional distribution of  $f^{(1)}(i_1, i_2, x_3, x_4)$  again in the form of rectangles. Then in each rectangle belonging to the channel  $i_1, i_2, i_3, i_4$ , one can display the one-dimensional slice  $f(i_1, i_2, i_3, i_4, x_5)$ . Employing this algorithm and using successive zooming one can localize the positions of five-dimensional peaks.

Though realizing the technical limitations, apparently the technique of embedded subspaces lends itself to generalization for  $p$  – dimensional nuclear spectra employing several level merging and projections. Without loss of generality, we shall assume that  $p$  is even. Analogously to the above-given relations one can write

$$f^{(1)}(x_1, x_2, \dots, x_{p-2}) = \sum_{x_{p-1}=x_{(p-1) \min}}^{x_{(p-1) \max}} \sum_{x_p=x_p \min}^{x_p \max} f(x_1, x_2, \dots, x_p) \quad (9)$$

$$f^{(2)}(x_1, x_2, \dots, x_{p-4}) = \sum_{x_{p-3}=x_{(p-3) \min}}^{x_{(p-3) \max}} \sum_{x_{p-2}=x_{(p-2) \min}}^{x_{(p-2) \max}} f^{(1)}(x_1, x_2, \dots, x_{p-2}) \quad (10)$$

.

.

$$f^{(j)}(x_1, x_2, \dots, x_{p-2j}) = \sum_{x_{p-j-1}=x_{(p-j-1) \min}}^{x_{(p-j-1) \max}} \sum_{x_{p-j}=x_{(p-j) \min}}^{x_{(p-j) \max}} f^{(j-1)}(x_1, x_2, \dots, x_{p-2j-2}) \quad (11)$$

.

.

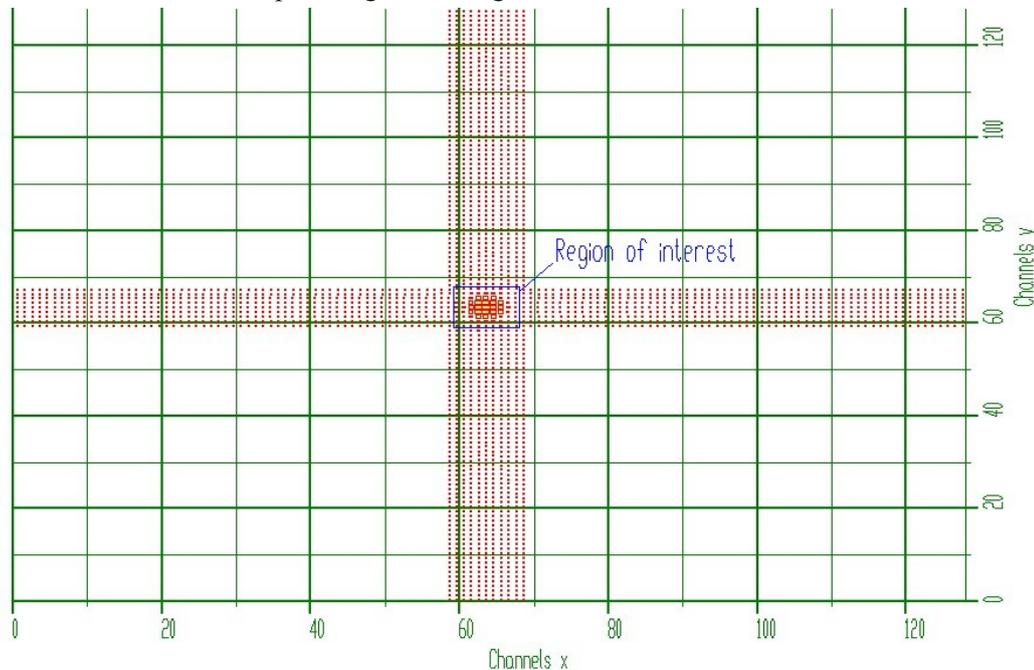
$$f^{(p/2-1)}(x_1, x_2) = \sum_{x_3=x_3 \min}^{x_3 \max} \sum_{x_4=x_4 \min}^{x_4 \max} f^{(p/2-2)}(x_1, x_2, x_3, x_4), \quad (12)$$

where  $j$  is the level of merging. Apparently, if  $p$  is odd 0-level subspaces are one-dimensional..

Obviously, from theoretical point of view this algorithm has no limitation. However, due to technical limitations of today's computers (sizes of memories, resolution of displays) the practical meaning of these formulas for higher values of  $p$  is rather restricted. Finally, we would like to emphasize that the given algorithm of embedded subspaces presents one of the possible approaches to cope with the problem of visualization of multidimensional nuclear spectra. In principle one can change dimensionality of subspaces at every level of merging, rotate subspaces, define other projection functional etc.

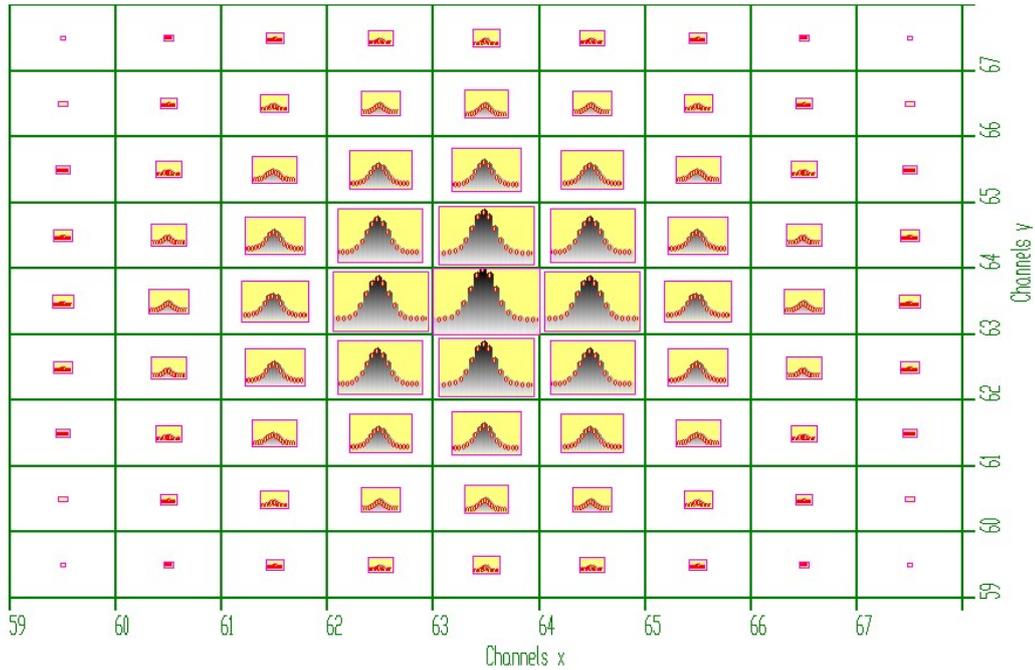
### 3.1 Three-dimensional spectra

Using the technique of embedded subspaces we can divide three-dimensional space to outer two-dimensional subspace and inner one-dimensional subspaces (slices in the third variable). An example of the projection of synthetic three-dimensional Gaussian and two ridges to two-dimensional outer subspace is given in Fig. 30.



**Fig. 30 Projection of three-dimensional synthetic spectrum to two-dimensional outer subspace**

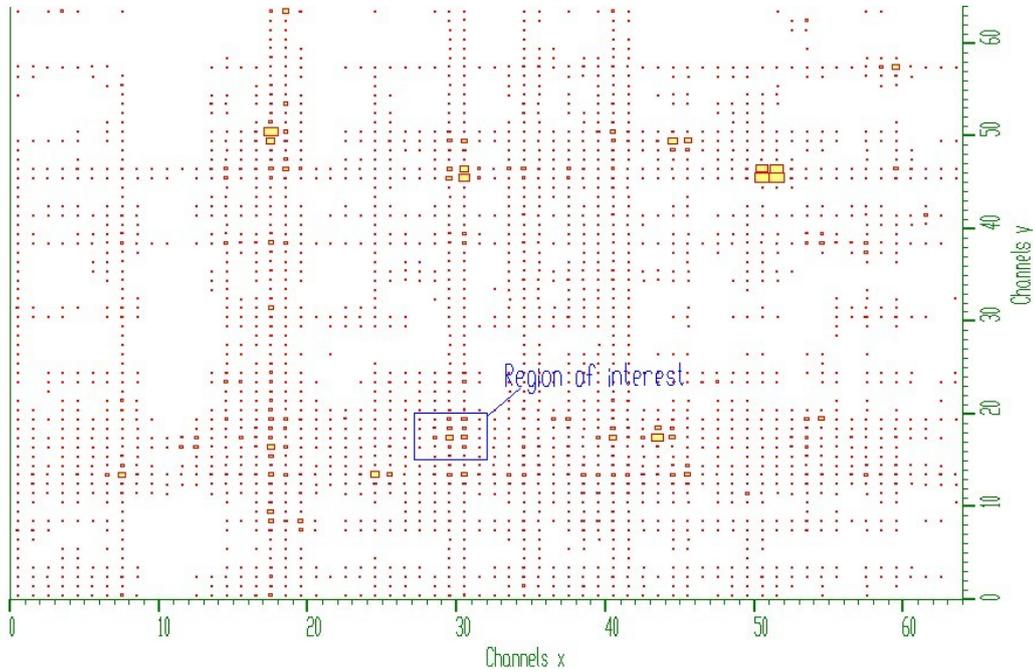
The sizes of small rectangles are proportional to the contents inside of the slice in the appropriate channel of outer space. The display inside of rectangles is senseless because of poor resolution. Let us imagine that we are interested in the peak region denoted in Fig. 30 as “Region of interest”. Let us focus the display to the ROI and enable the display inside rectangles (Fig. 31).



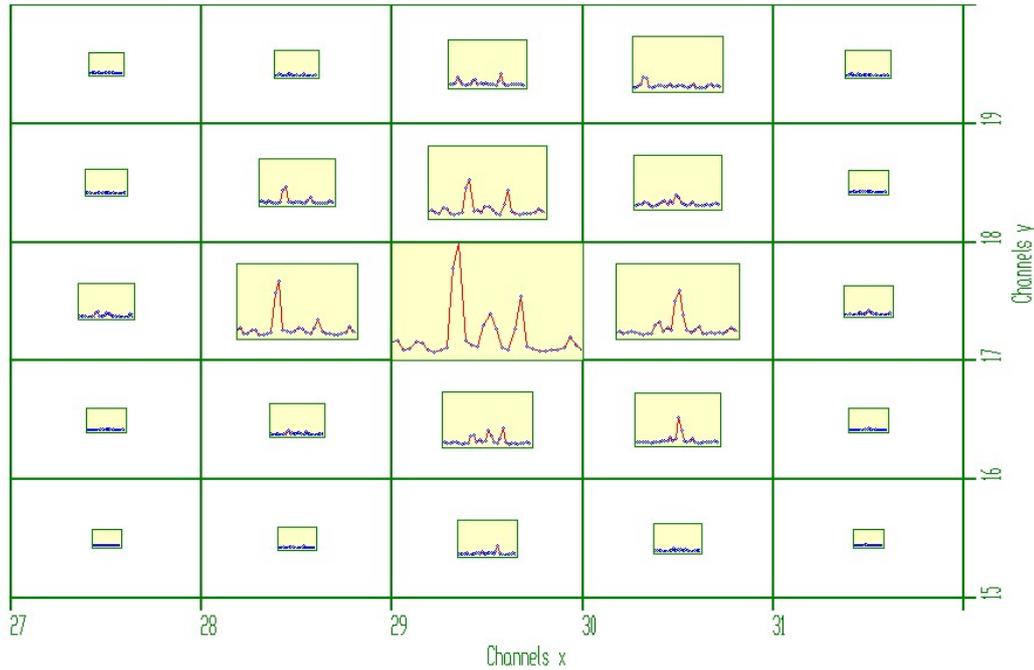
**Fig. 31 Displayed one-dimensional inner subspaces of the ROI**

One can see simultaneously the distribution of the two-dimensional projection (yellow squares) together with one-dimensional slices. One can observe correlations among neighboring points inside of rectangles as well as correlation of corresponding points in the rectangles in both x and y directions.

Let us illustrate an example of experimental three-dimensional spectrum. Again, outer subspace and zoomed ROI are given in Fig. 32 and Fig. 33, respectively.

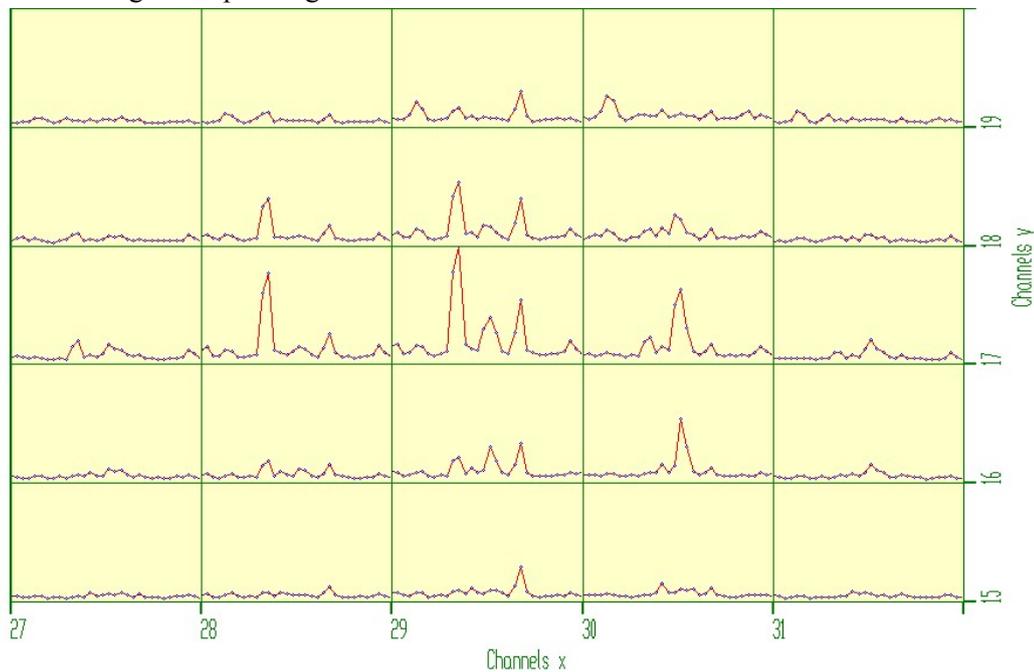


**Fig. 32 Experimental three-dimensional spectrum - outer subspace**



**Fig. 33** Zoomed ROI of the data from Fig. 32

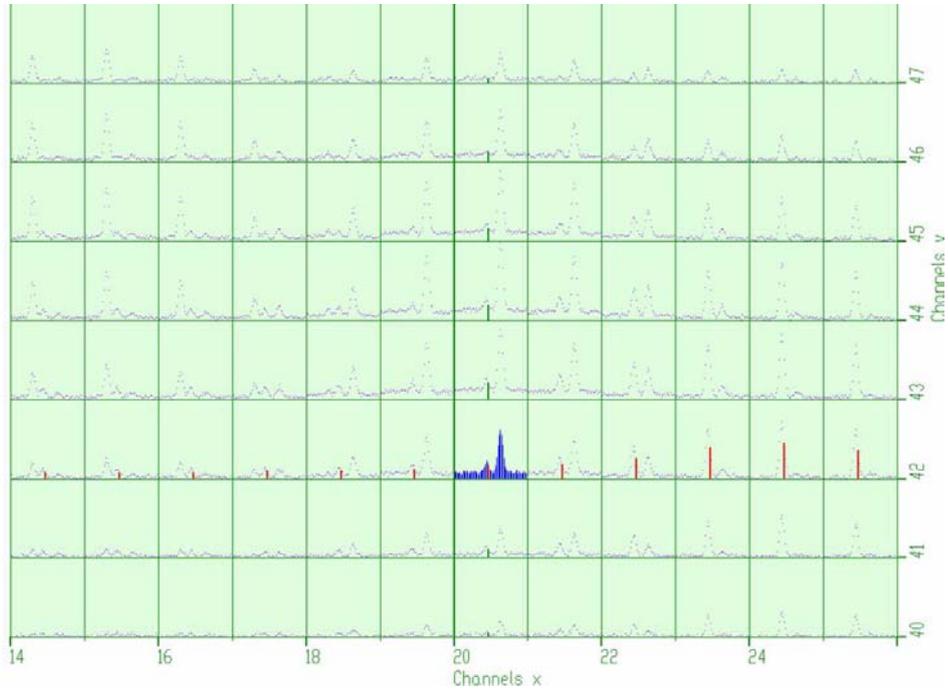
If the experimenter is not interested in the distribution in outer subspace and wishes to see details in the rectangles the size can be set equal (Fig. 34). Then one can observe better the correlations among corresponding channels.



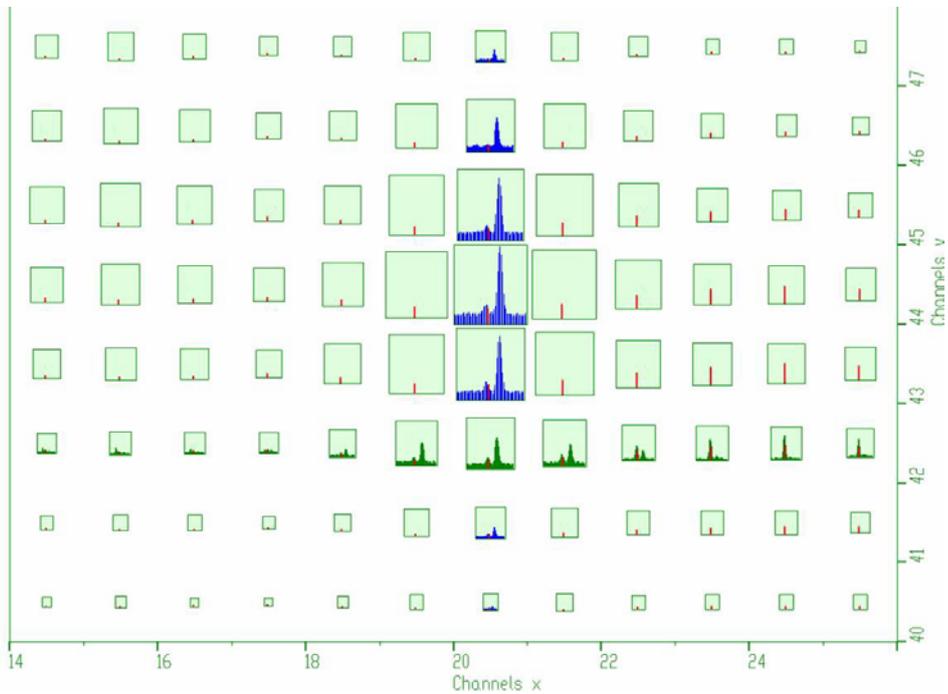
**Fig. 34** Zoomed ROI of the data from Fig. 32 with equal sizes of rectangles

Analogously with the direct visualization technique now we can also display one- and two-dimensional slices and thus to determine correlations among neighboring channels in different directions. In Fig. 35 we present spectrum in point display mode with equal sizes of rectangles

together with one-dimensional slices for fixed  $xy$  (blue bars),  $xz$  (green bars) and  $yz$  (red bars) variables, respectively. In Fig. 36 we present two dimensional slices for fixed  $x$  (blue bars),  $y$  (green bars) and  $z$  (red bars) variables, respectively.



**Fig. 35 One-dimensional slices for fixed  $xy$ ,  $xz$  and  $yz$  variables**

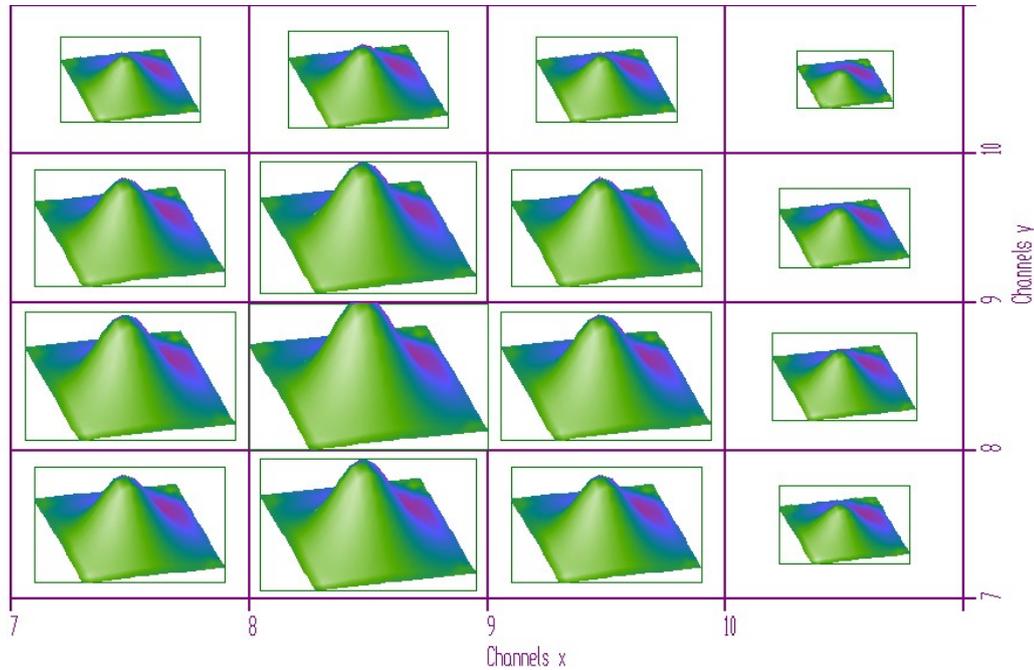


**Fig. 36 Two-dimensional slices for fixed  $x$ ,  $y$  and  $z$  variables**

POS(ACT)064

### 3.2 Four-dimensional spectra

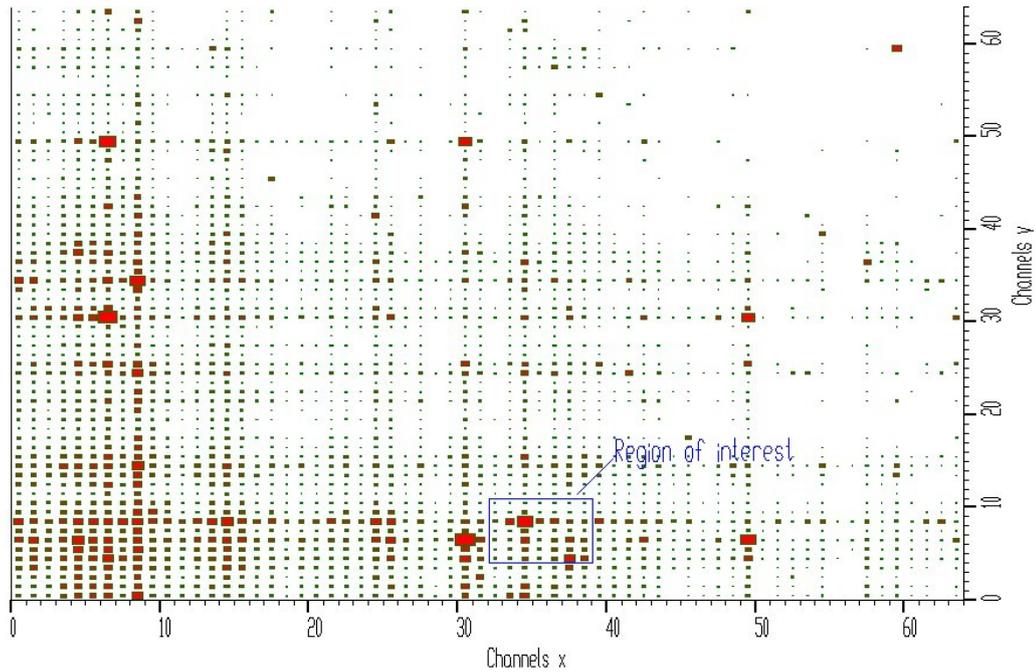
Now the dimensionality of both outer and inner subspaces will be two. An example of a view focused on four-dimensional synthetic Gaussian is illustrated in Fig. 37.



**Fig. 37 Four-dimensional synthetic Gaussian (shown both, outer and inner subspaces)**

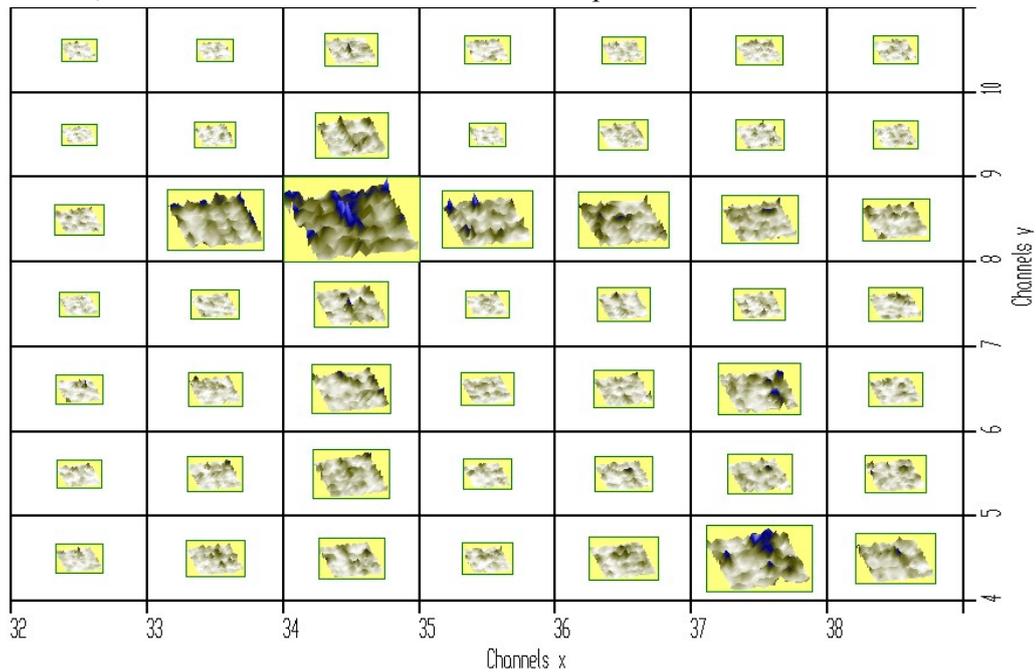
One can watch correlations in four dimensions, i.e., in two-dimensional subspace in each rectangle and among corresponding points in neighboring rectangles in both directions in outer subspace.

Let us proceed to experimental four-dimensional spectrum. In Fig. 38, one can see a projection of the four-dimensional rather noisy spectrum to outer subspace and in Fig. 39 zoomed ROI of it.



**Fig. 38 Projection (outer subspace) of four-dimensional experimental  $\gamma-\gamma-\gamma-\gamma$ -ray coincidence spectrum**

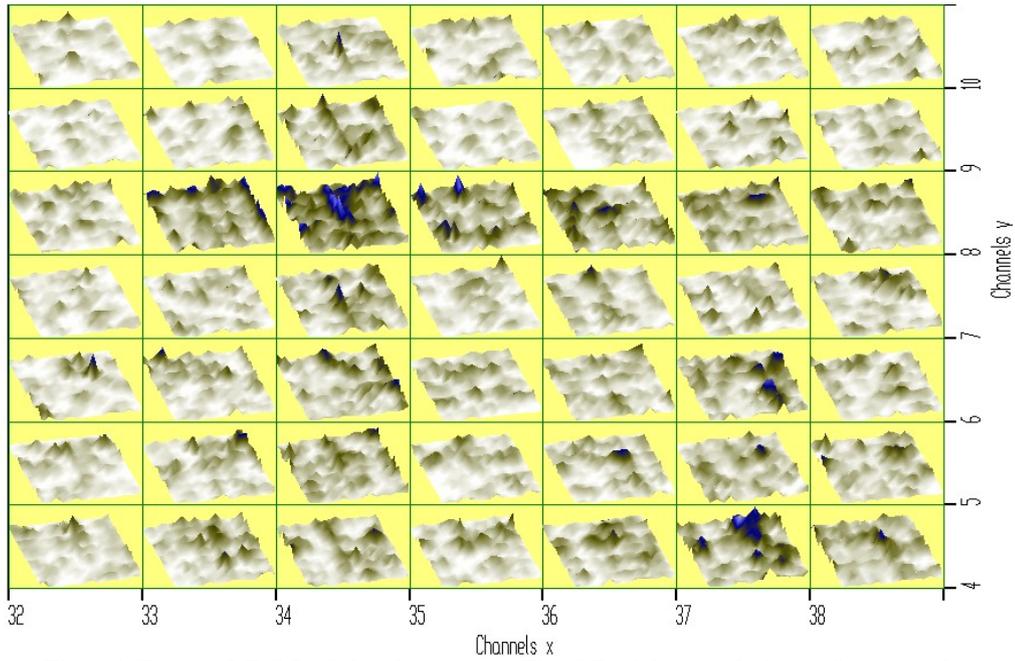
From this view, one can observe basic tendencies in the spectrum.



**Fig. 39 Zoomed ROI of the data from Fig. 38**

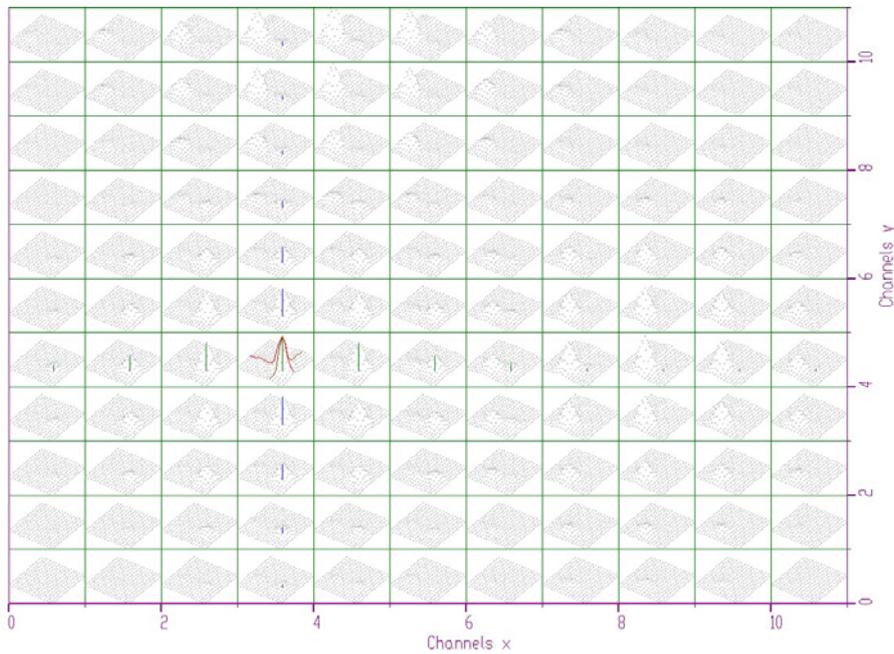
If desired, to see better the details, one can expand rectangles to equal size (Fig. 40).

POS (ACAT) 064



**Fig. 40** Zoomed ROI of the data from Fig. 37 with equal sizes of rectangles

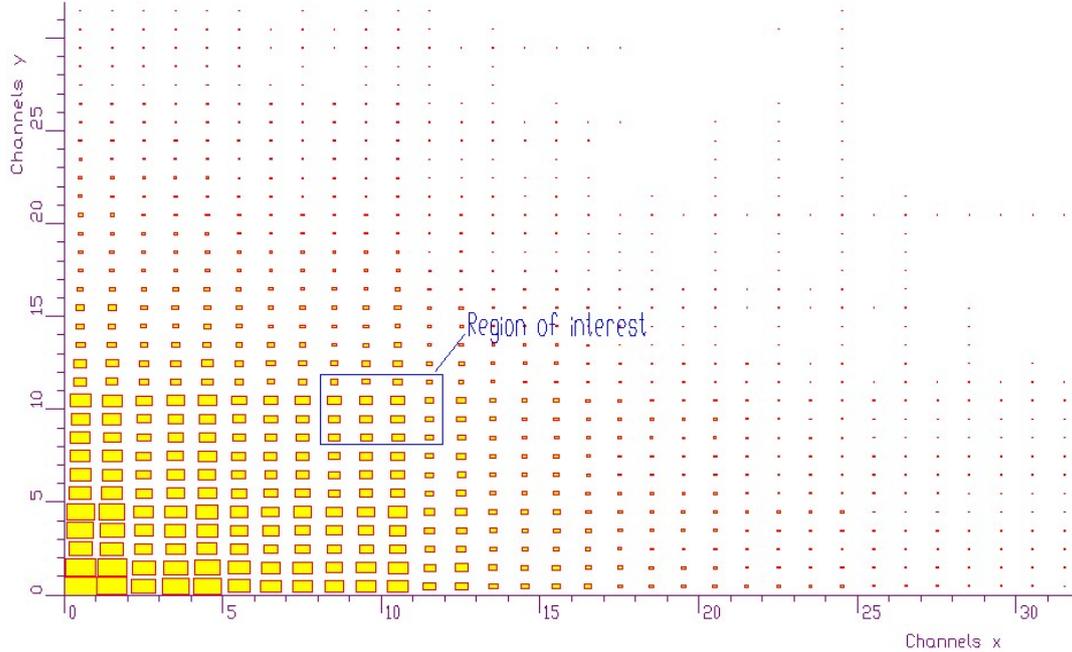
Also for four-dimensional spectra displayed using the technique of embedded subspaces one can show one-, two- and three-dimensional slices. In Fig. 41 one can see four-dimensional spectrum shown in points display mode with equal sizes of rectangles together with four one-dimensional slices for fixed  $xyz$ ,  $xyv$ ,  $xzv$  and  $yzv$  variables, respectively. One can observe the correlations among neighboring channels in four directions.



**Fig. 41** Four-dimensional spectrum shown in points display mode together with four one-dimensional slices for fixed  $xyz$ ,  $xyv$ ,  $xzv$  and  $yzv$  variables

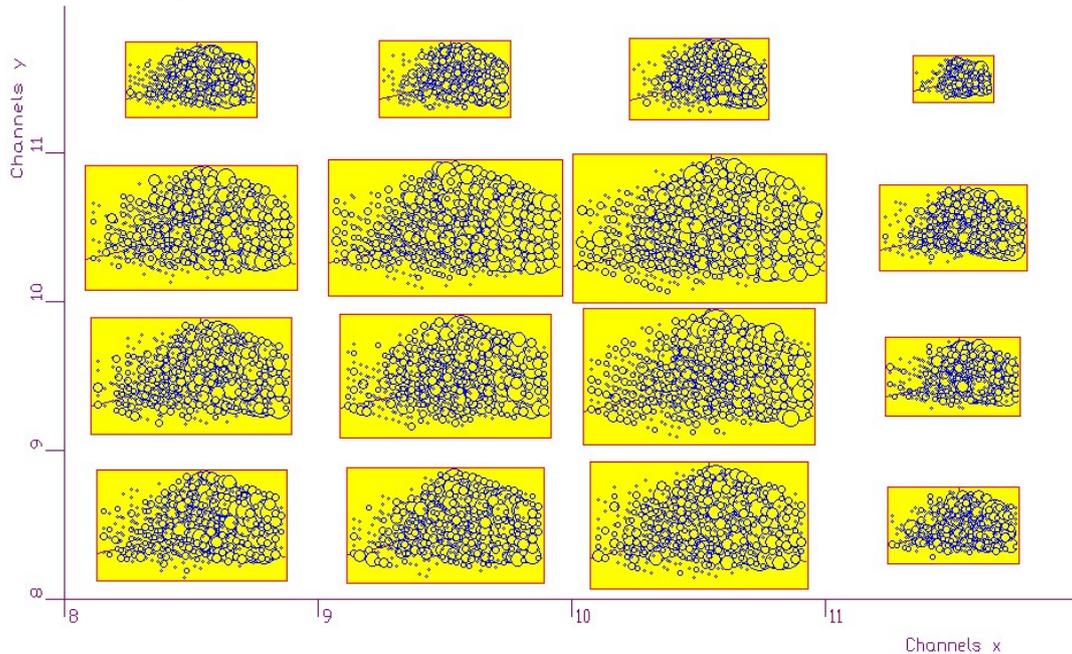
### 3.3 Five-dimensional spectra

Five-dimensional space can be divided in two ways, i.e., either two-, plus three-dimensional subspaces or two-, plus two-, plus one-dimensional subspaces. Let us start with the first case. In Fig. 42 one can see outer subspace of five-fold  $\gamma$ -ray spectrum.



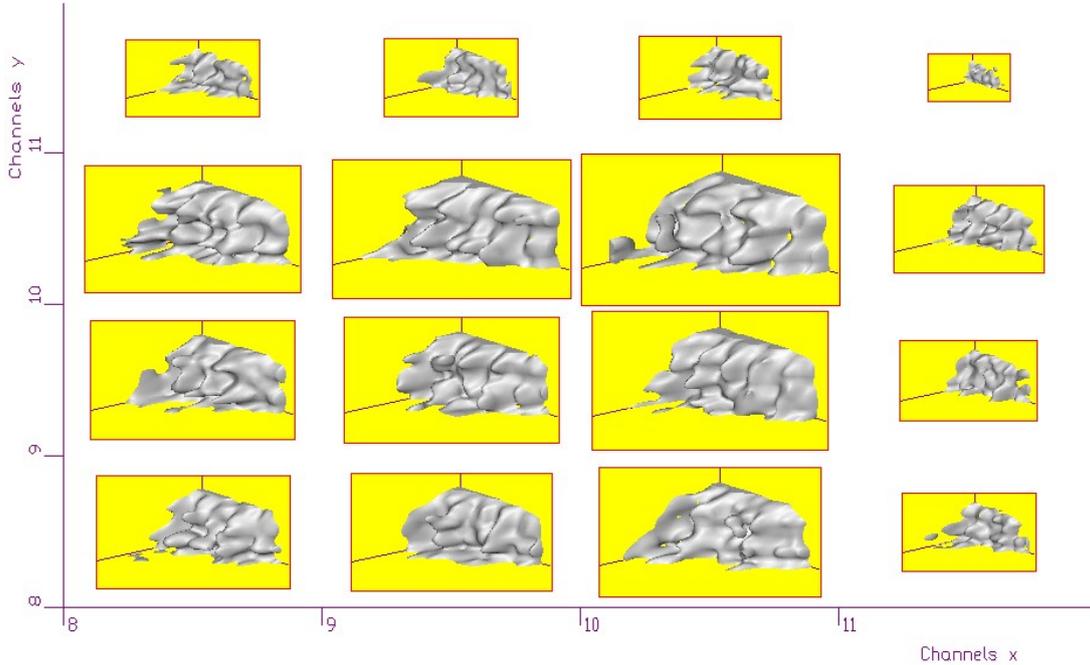
**Fig. 42** Outer subspace of five-fold  $\gamma$  – ray spectrum

One can observe high level of background due to relatively low statistics in the spectrum. Let us assume we are interested in the outlined ROI. Zoomed ROI with enabled display of inner three-dimensional subspaces is illustrated in Fig. 43.



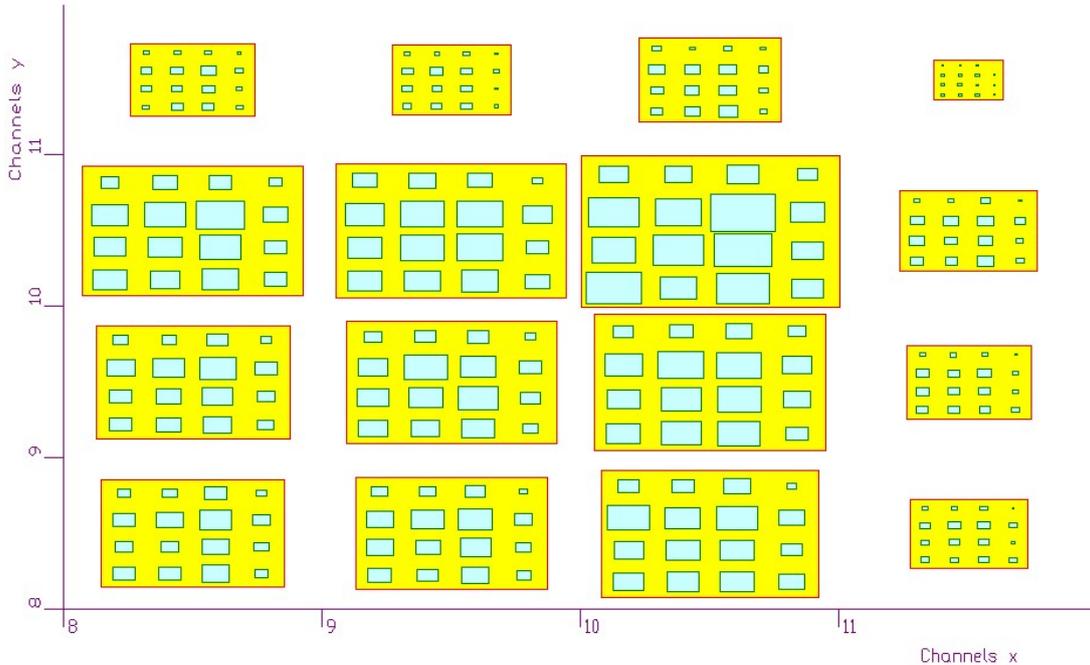
**Fig. 43** Zoomed ROI of the data from Fig. 42 with displayed three-dimensional inner subspaces

From the presented chunk of the five-dimensional space, mainly from the shown inner three-dimensional subspaces it is difficult to discover tendencies in the spectrum. It can be improved, to some extent, by smoothing the data using B-splines and employing isosurface display technique shown in Fig. 44.



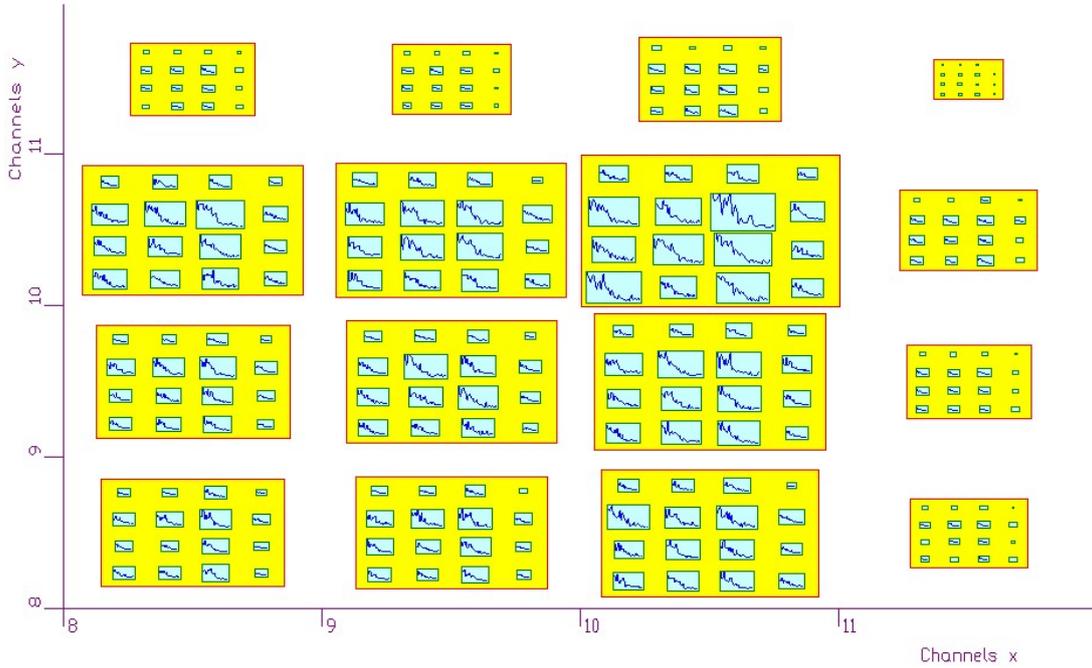
**Fig. 44 Zoomed ROI of the data from Fig. 42 with inner subspaces displayed in isosurface display mode with B-spline smoothing**

The alternative way is to proceed in the projections and to divide inner three-dimensional subspaces to two- plus one-dimensional ones. In Fig. 45 we show outer and the first level inner subspaces (display of the second level inner subspaces was disabled).



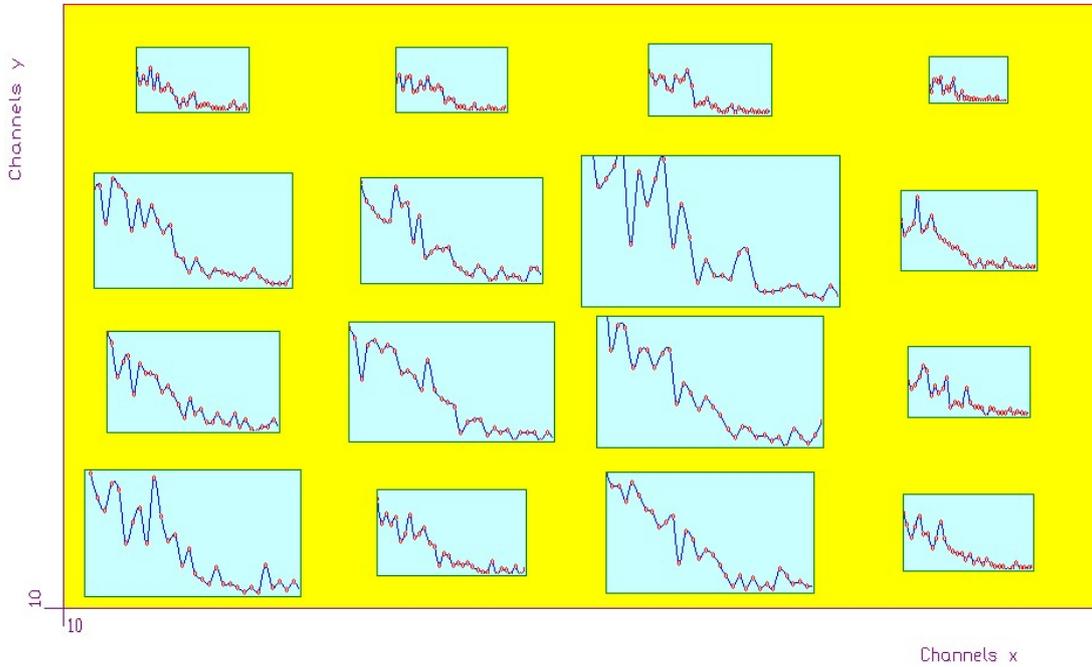
**Fig. 45 Zoomed ROI with shown outer and the first level inner subspaces**

While yellow rectangles represent outer subspace, the light blue ones represent the first level inner subspaces. If we enable the display of the second level subspaces, we can see all three levels simultaneously (Fig. 46).



**Fig. 46 Zoomed ROI with shown outer, the first level and the second level inner subspaces**

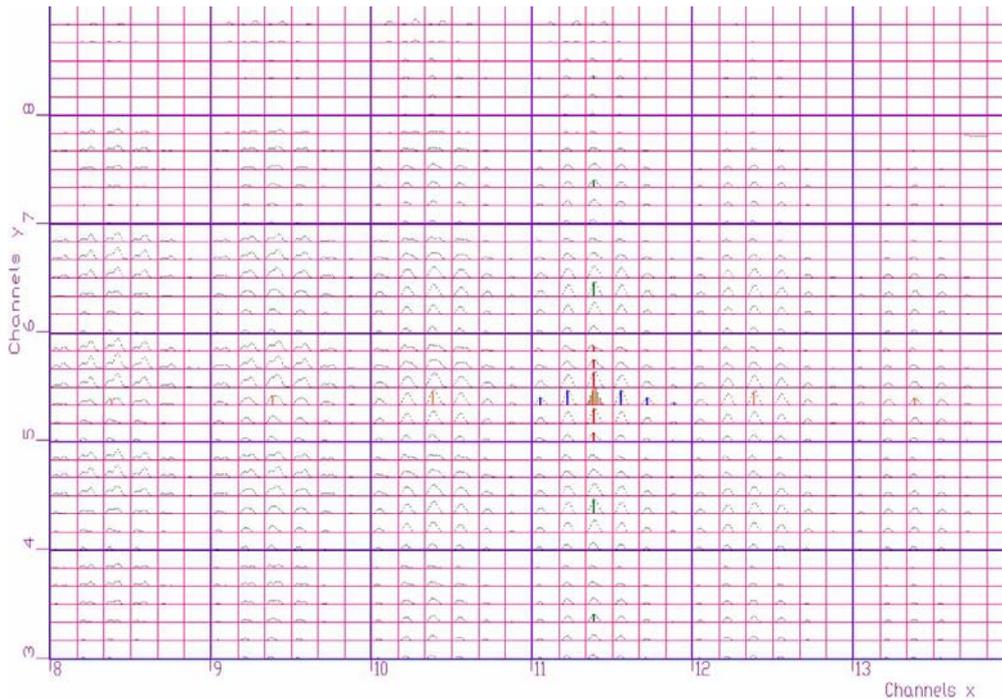
One may focus attention to the channel  $x = 10$ ,  $y = 10$ , where the volume of data is the biggest (yellow rectangle here is the largest, Fig. 47).



**Fig. 47 One channel of the outer subspace zoomed**

Now one can see in detail the slices in the second level inner subspaces. Employing this technique one can scan multidimensional nuclear spectra. Moving from the outer subspace to inner ones and back gives possibility to discover interesting parts in such a spectrum.

Let us denote the fifth independent variable as  $w$ . Analogously to three- and four-dimensional data in Fig. 48 we show five-dimensional spectrum shown in points display mode with equal sizes of rectangles together with five one-dimensional slices for fixed  $xyzv$ ,  $xyzw$ ,  $xyvw$ ,  $xzvw$  and  $yzvw$  variables, respectively. One can observe the correlations among neighboring channels in five directions.



**Fig. 48** Five-dimensional spectrum shown in points display mode together with five one-dimensional slices for fixed  $xyzv$ ,  $xyzw$ ,  $xyvw$ ,  $xzvw$  and  $yzvw$  variables

#### 4. Conclusions

In the paper we have presented conventional as well as new developed visualization algorithms of nuclear spectra. For 3D spectra we have proposed particle gradient display technique and isosurface display technique. Raw data can be interpolated using B-spline algorithms up to 4-th degree. For 4D spectra we have designed the algorithms based on slicing in fourth dimension, pies display mode as well as isovolume display mode.

Furthermore, we have derived new technique of visualization of multidimensional spectra based on projections of embedded subspaces. This allows one, in interactive way, to localize interesting parts in the data of this kind, to find correlations among neighboring points and to discover trends in multidimensional data.

The visualization algorithms presented have been implemented in the data acquisition, processing and visualization system DaqProVis which is being developed at Institute of Physics, Slovak Academy of Sciences [9]. The algorithms for the display of 2D spectra have also been implemented in ROOT system in TSpectrum2Painter class (SPECTRUMPAINTER directory) [10].

### Acknowledgements

The work is supported by the Grant Agency of Slovak Republic through contract GAV 2/7117/27.

### References

- [1] M. Morháč, I. Turzo, J. Krištiak, *PC-CAMAC based data acquisition system for multiparameter measurements*, *IEEE Trans. on Nuclear Science* **42** (1995) 1.
- [2] M. Morháč, J. Kliman, V. Matoušek, I. Turzo, *Integrated multiparameter nuclear data analysis package*, *Nuclear Instruments & Methods in Physics Research* **A389** (1997) 89.
- [3] D. C. Radford, *ESCL8R and LEVIT8R: Software for interactive graphical analysis of HPGe coincidence data sets*, *Nuclear Instruments & Methods in Physics Research* **A361** (1995) 297.
- [4] J. Baczynski, *PC display system for real-time data graphical representation of spectroscopy measurement results*, *Nuclear Instruments & Methods in Physics Research* **A325** (1993) 523.
- [5] D. Hearn, M. P. Baker, *Computer Graphics*, Prentice Hall International, , Inc., New Jersey 1994.
- [6] W. M. Newman, R. F. Sproull, *Principles of Interactive Computer Graphics*, McGraw-Hill, Inc., London 1979.
- [7] R. S. Gallagher, *Computer Visualization Graphics Techniques for Scientific and Engineering Analysis*, CRC Press, Boca Raton, Ann Arbor, London, Tokyo 1995.
- [8] M. Morháč, J. Kliman, V. Matoušek, I. Turzo, *Sophisticated visualization algorithms for analysis of multidimensional experimental nuclear data*, *Acta Physica Slovaca* **54** (2004) 385.
- [9] M. Morháč, V. Matoušek, I. Turzo, J. Kliman, *DaqProVis, a toolkit for acquisition, interactive analysis, processing and visualization of multidimensional data*, *Nuclear Instruments & Methods in Physics Research* **A559** (2006) 76.
- [10] R. Brun, F. Rademakers, S. Panacek, D. Buskulic, J. Adamczewski, M. Hemberger, *ROOT, An Object-Oriented Data Analysis Framework*, Users Guide 3.02c, CERN, 2002.