Animator: A Tool for the Animation of Parallel Coordinates

N. Barlow, L. J. Stuart
The Visualization Lab, Centre for Interactive Intelligent Systems,
University of Plymouth, Plymouth, UK
nigel.barlow@plymouth.ac.uk , liz.stuart@plymouth.ac.uk

Abstract

In this paper, an uncommon use of parallel coordinates is illustrated using the Animator software. Animator is used to plot the parallel coordinates of objects in multi-dimensional space. Subsequently, the Animator software is used to animate the movement of individual objects, in this multi-dimensional space over time. Initial empirical studies of this technique for the visualization of data from Neurophysiological research, multi-dimensional spike train datasets, have shown that the technique is useful. Thus, Animator was developed for public access and is now freely available (including source code) from the Visualization Lab at the University of Plymouth, www.plymouth.ac.uk/infovis.

1 Introduction

Information Visualization is a relatively new domain of computer science which began as a result of the landmark report presented to the US National Science Foundation by McCormick, DeFanti and Brown [1] in 1987. This report was devoted to the issues that were currently arising from "Visualization in Scientific Computing". The report highlighted the fact that computationally the computer science community was advancing as it was now able to process greater quantities of data in less time. Furthermore, due to increases in computational capability, it was now possible to create massive quantities of data in previously inconceivable time periods. However, as a result of overcoming the "bottleneck" of computational capability, a new bottleneck was identified, namely the interpretation of these vast quantities of data. The report referred to the "fire hoses of data" that were being generated and highlighted the requirement for researchers to redirect their effort towards solving this new problem.

Since this time, researchers have started to develop strategies and mechanisms to support the process of comprehending these massive datasets. Indeed, the subject of Information Visualization is now recognized and many internationally prestigious researchers are now actively involved in this area.

2 Parallel Coordinates

One of the methods which have resulted from this new discipline of Information Visualization is that of Parallel coordinates. Parallel coordinates are a technique used to represent diverse sets of multi-dimensional data. These were originally pioneered in the 1980's, however in 1990, Inselberg and Dimsdale [2] and Wegman [3] regenerated their use within the Information Visualization community. It was their introduction of new features and their use of parallel coordinates for the comprehension of massive datasets of multi-dimensional data that led to the renewed interest in their use.

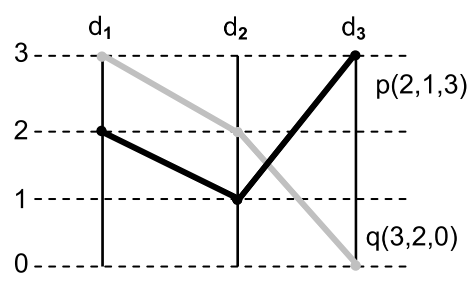

Figure 1. Representation of the points p(2,1,3) and q(3,2,0) in three dimensions using parallel coordinates.

Proceedings of the Eighth International Conference on Information Visualisation (IV'04)
1093-9547/04 $ 20.00 IEEE

This presentation of parallel coordinates denotes data points as vertical axis coordinate values distributed along a horizontal axis.

Thus, a specific point in n-dimensional Euclidean space is represented by n vertical axes values distributed along the horizontal axis. To illustrate this, consider the two 3-dimensional points p(2,1,3) and q(3,2,0). Refer to Figure 1 for an illustration of these two points in 3-dimensional space represented in parallel coordinates.

Parallel coordinates are traditionally used to identify correlations between variables and to convey aggregation information. An excellent example of such a parallel coordinate system is the PARallel COordinate Visualization, PARCOVI [4] system. Within PARCOVI, parallel coordinates are traditionally used to analyse the relationship of many variables, denoted in parallel axes, with one another. PARCOVI also provides considerable additional functionality including the capability to interactively modify data based different types of operations [5].

3 The Animator software

In this paper, a different concept for the use of parallel coordinates is presented. This concept was first proposed used by Stuart et al. [6] in 2001. The idea is to use parallel coordinates as a simple means of representing n-dimensional coordinates in a 2-dimensional plane. This would represent a snapshot of the position of multiple objects in time. It is proposed that these parallel coordinates are animated over time to represent the changing position of objects within the multi-dimensional space. Note that this change in position is viewed simply in 2-dimensional space.

3.1 Input Window

The Animator tool is used to visualize a number of parallel coordinates, n, in a number of dimensions d, for the duration of t time slices. Thus, n is the number of parallel coordinates that will be displayed in each snapshot. Each of the parallel coordinates in this snapshot will be displayed in d dimensions; this represents the number of vertical axes that are used in the display.

The initial screen of the Animator tool is shown in Figure 2. At this stage, the user is required to select the input filename and to input the values of n, d and t. The user is then able to proceed into the main part of the tool. Once the user is in the main part of the tool, the two core windows are shown. These are the controls area window and the display window.

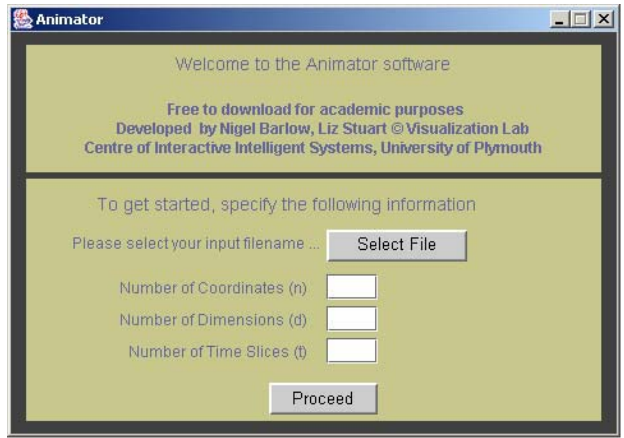

Figure 2. The initial screen of the Animator tool

3.2 Controls Area

The controls area of the tool provides the user with the flexibility to tailor the display in a number of ways. Based on much of the current research in interface design, it is essential that users have the flexibility to steer the exploration of their datasets. Furthermore, Shneiderman et al. [7] raised awareness that users need to be able to overview, to filter and zoom into data and finally, to drill down to get the lower level details of the values of individual data items. The main aim of the controls area is to provide this functionality for the user.

A snapshot of the controls area is shown in Figure 3. On the left, a list box is used to enable the user to specify any combination of parallel coordinates for display. Additionally, a command button is used below this list box to enable the user to quickly select all coordinates for display.

Currently, the user is not able to select non-consecutive group of time slices due to the perceptual complexity of the interpretation of this data.

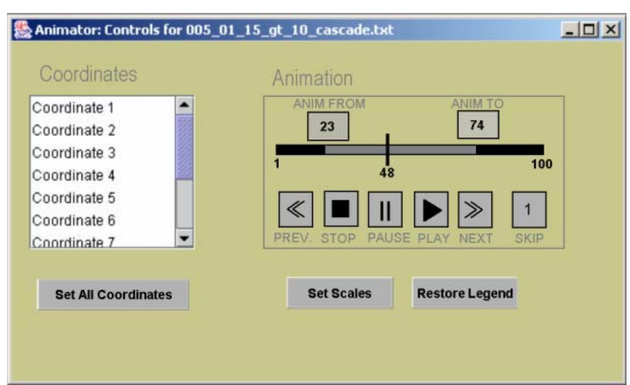

Figure 3. The Controls Area of the Animator tool

On the right of the controls area, the Animation facilities of the tool are shown. Overall, there are two modes of traversing through the dataset: animating or stepping. Regardless of the mode, the current time slice displayed is indicted by the vertical bar on the slider and the text box underneath this bar.

In the controls area, it is possible to animate all of the currently selected coordinates for all of currently selected time slices by clicking on the “Play” button. The user can end the animation by clicking on the “Stop” button and a “Pause” facility is also available. Using the slider bar, it is possible to fast forward or reverse through the animation.

The user is also able to step through the dataset, one snapshot at a time. This provides the user with the facility to study snapshots in greater detail. To advance the display to the next snapshot the user clicks the “Next” button and “Previous” to go back one snapshot.

Due to the fact that the user has not been provided with the facility to select non-consecutive time slice, a facility of progressing through the animation quickly has been specified. In either mode of traversing of the dataset, the user is able to skip snapshots thus only displaying every i^{th} snapshot of the currently selected dataset. Initially, the default value for i is set to 1 by the system. However, the user can simply change the value of i by editing the “Skip” text box on the form shown in Figure 3.

Note that the controls area also provides access to the data and legend windows.

3.3 Data Window

The data displayed by Animator must be normalized in order to provide a meaningful display. This normalization may be any one of the three distinct methods available. Thus normalization can be based on the current selection of the maximum and minimum values of

- the complete dataset (DATASET)
- the current snapshot (SNAPSHOT)
- each dimension within the current snapshot (DIMENSION).

Initially, DIMENSION normalization may be confusing to the user as the scales of the axes are constantly changing; however, it is useful to a more experienced user for finer analysis of the dataset.

In addition to normalization, the data window also provides the user with the facility to specify the resolution of the data displayed.

3.4 Display Area

The display area is the main window in which the parallel coordinates are displayed. All interaction with the controls area directly changes the display in this window.

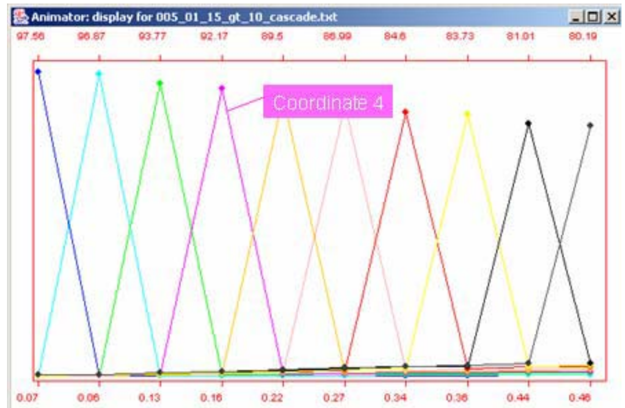

Figure 4. A snapshot of the display area of the Animator tool at time slice 5

A snapshot of the display window is shown in Figure 4, where n=10, d=10, t=10000 and the normalization was SNAPSHOT and snapshot 5 of this dataset is displayed.

The display area provides the user with some additional functionality. If the user clicks on any parallel coordinate line, the number of that parallel coordinate is display as shown in Figure 4. Alternatively, if the user clicks on the intersection point of any parallel coordinate lines, then the value is displayed as shown in Figure 5, in which time slice 12 of the same snapshot is displayed. All of this data is colour coded for continuity based on the legend of the current dataset.

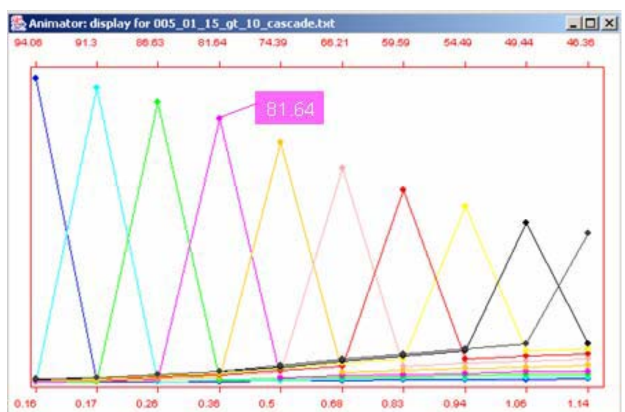

Figure 5. Display Area of the Animator tool at time slice 12

Proceedings of the Eighth International Conference on Information Visualisation (IV’04)
1093-9547/04 $ 20.00 IEEE

The dataset used to capture snapshots of the display area represents the position of 10 objects in 10-dimensional space. Note the change in position of these objects from time slice 5 to 12 and then onto 18, shown in Figure 6. From these three figures, the following trend can be deduced: As the coordinate number increases so the position of the corresponding objects in n-dimensional space get closer. This analysis corresponds accurately to the anticipated trends of this dataset. An in-depth study of this data is beyond the scope of this paper, however it is currently available [10].

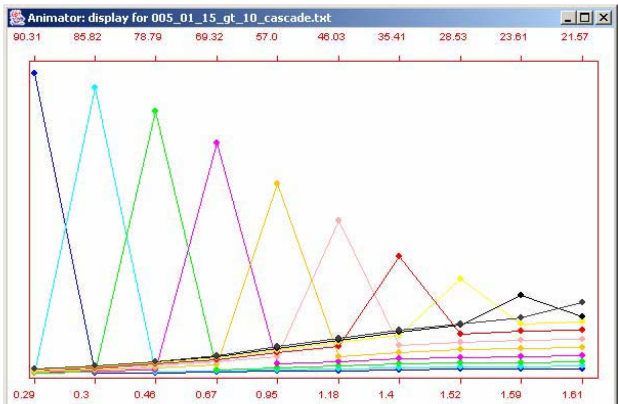

Figure 6. Display Area of the Animator tool at time slice 18

3.5 Coincident parallel coordinates

One of the main issues highlighted when using the initial prototype for this software was the procedure for dealing with coincident parallel coordinates. Due to the fact that the tool is fundamentally using a flat 2-d plane to display all of the data, this was an issue. Thus, the initial version of Animator addresses this to a lesser extent by enabling the user to detect coincidence. Hence, if the user clicks on a point at which there is more than one parallel coordinate, the value of the coordinate at that point is displayed. An example of this is shown in Figure 7.

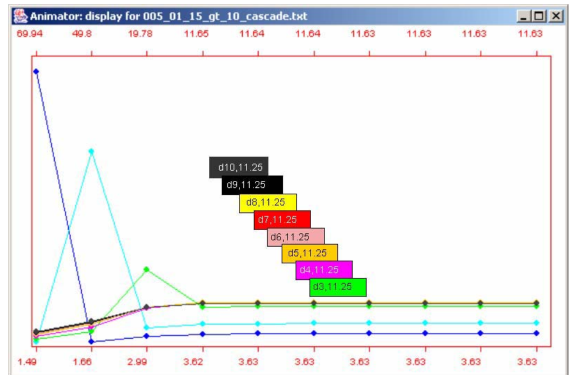

Figure 7. Display Area of the Animator tool at time slice 60. Note the seemingly coincident parallel coordinates.

Note that numerically, the points are not exactly the same, however, visually they appear to be the same. This is an example of when the DIMENSION normalisation is helpful for analysis.

This solution is reasonable at this stage whilst other methods of representing this data are considered. One such method is the use of 3D space to capture this data. In Figure 8, a snapshot of this concept is shown. In this figure, the coincidence revealed in Figure 7 is shown more distinctly.

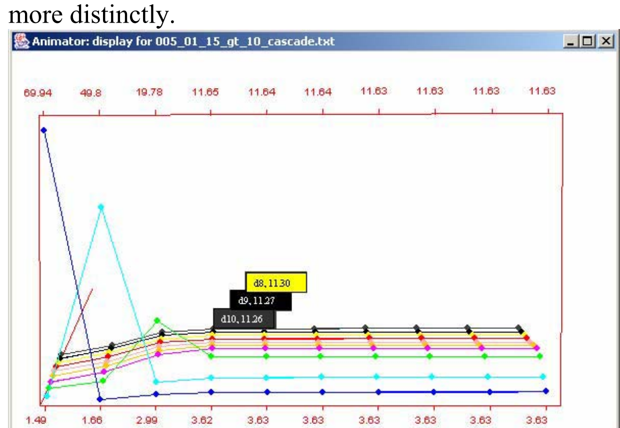

Figure 8. Display Area of the Animator tool at time slice 60 using 3D to alleviate the problem of coincidence.

3.6 Design issues

One of the fundamental design goals of this tool is flexibility for the user to tailor their environment so that it is as “natural” and intuitive to the user as possible. In particular the flexibility, terminology and clarity of the system were developed to satisfy this design goal.

Flexibility

In order to facilitate this, the majority of the initial system specifications can be superseded by the user. For example, in the display area, every parallel coordinate is allocated a colour by the system initially. The user is able to re-define the colour of any/all parallel coordinates and furthermore, they can also define the texture of the line type. This texture facility was included to enable the user to easily capture displays for monochrome publications when necessary due to the characteristics of the dataset. Other examples of this flexibility include the user's ability to change the background colour of the display area as well as the font, its style, size and colour.

Terminology

Currently the system refers to parallel coordinates and time slices. However, this terminology may not be natural to the user. For example, in the initial trials of the software, the term "parallel coordinate" was swapped for the term "particle" and the term "time slice" was swapped for the term "iteration". Therefore, the user could analyse "the position of their particles at different iterations of the algorithm" which was a more natural expression of the scenario for the user. This ability to tailor the interface should help the user to forget about how they must interact with software and enable them to get on with the business of actually analyzing the data.

Clarity

Due to the fact that Animator uses from two to four windows for each instance of the application, it is difficult for users to track windows when viewing two different datasets at the same time. Therefore, it was necessary to label each of the windows in the application with the filename of the original dataset. This way the user can easily track the datasets being analysed.

4 Initial results of Neurophysiological empirical studies

In brief, it is known that neuronal signals consist of short electrical pulses, known as "spikes", which have amplitudes of approximately 100mV and a typical duration of approximately 1-2ms. A sequence of spikes emitted by a specific neuron is known as the "spike train" of that neuron. Empirical studies were based on datasets made up of spike trains from a number of neurons, which could be connected in different architectures, thus multi-dimensional spike train datasets were analysed.

In 1985, Gerstein et al. [8], [9] defined the "gravity transform" algorithm, a novel and highly significant algorithm that modeled *n* spiking neurons as *n* particles in *n*-dimensional space. The position of the particles, which could change over time, was based upon their charge and the subsequent attraction of particles together. The charge of the particles was proportional to the spiking activity of the relative neuron.

Thus, Gerstein defined a method of interpreting multi-dimensional spike train datasets as particles undergoing attraction within a multi-dimensional space. This is an overly simplified description of the algorithm, however, the complete mathematical definition of the algorithm is available. The sole drawback to the gravity transform algorithm was the method of outputting data called a "distance graph".

In 2001, Stuart et al. [6] used an earlier version of the parallel coordinates animation software (not publicly released) to enhance the presentation of the data output from the gravity transform algorithm. This method of using parallel coordinates to animate particles in a multi-dimensional space underwent initial empirical testing for small assemblies of neurons. Furthermore, this technique of representing multi-dimensional spike train datasets has been used in further Neurophysiological studies [10].

5 Conclusions

This paper has presented an uncommon use for parallel coordinates in which animation is used to view multi-dimensional datasets over time. As described in section 4, this work has already undergone testing using scientific data and the initial results have been encouraging. The aim of the research was to provide a general, flexible and intuitive tool that could be used by non-specialised users with minimal effort.

Currently, the focus of our work is to encourage a range of different user groups to use this tool and provide feedback on its usability. This work is ongoing within the Neurophysiological community and it is also likely to develop within the robotic areas.

6 Future Work

The software presented in this paper is a robust software tool that can be used for many different purposes. Furthermore, the tool is flexible and enables the user to tailor their environment in order to provide a 'familiar' and intuitive interface into their subject specific data.

Based on experimental use of the software with Robotic data, further extensions to the tool are also planned. The main extension is providing the user with the functionality to view multiple data sets simultaneously, and the capability to stack parallel coordinates on top of each other in order to view the evolution of data in different data sets over time. This concept was initially introduced and tested by Marocco, Stuart and Cangelosi [11] where data from adaptive algorithms used to perform basic robot tasks was analysed using stacked parallel coordinates. As this area of work has already shown that it has potential, this functionality will be developed and provided to users.

Currently, when the user initializes the system they are required to enter the values: *n*, *d* and *t* representing the number of parallel coordinates, the number of dimensions and the duration of time slices, respectively. However, experience with the prototype tool has shown that it is more efficient if these values can be read directly from the input file. Thus, the tool will be developed to provide the user with the flexibility to choose between these input methods.

The Animator software was produced as part of the ongoing work at the Visualization Lab in the University of Plymouth. The Animator software and source code are available from the project web site www.plymouth.ac.uk/infovis with the standard copyright conditions for copyright material.

7 References

[1] McCormick, B.H., DeFanti, T.A. and Brown, M.D. (1987), Visualization in scientific computing, *Special issue ACM SIGGRAPH Computer Graphics*, 21, 6.

[2] Inselberg, A. and Dimsdale, B. (1990), Parallel Coordinates: A tool for visualising multi-dimensional geometry, *In Proceedings of Visualization'90*, 361-378.

[3] Wegman, E.J. (1990), Hyperdimensional Data Analysis Using Parallel Coordinates, *Journal of the American Statistical Association*, 411(85), 664-675.

[4] http://europa.eu.int/en/comm/eurostat/research/conferences/etk-99/papers/parcovi.pdf

[5] http://europa.eu.int/en/comm/eurostat/research/conferences/ntts-98/papers/cp/002c.pdf

[6] Stuart, L., Walter, M. and R. Borisyuk, Visualization of multi-dimensional Spike Trains, *In Proceedings of 4th International workshop Neural Coding'2001*, 47-48, (2001).

[7] Shneiderman, B., The eyes have it: A task by data type taxonomy of information visualizations, *Proc. IEEE Symposium on Visual Languages '96*, IEEE, Los Alamitos, CA (September 1996), 336-343

[8] Gerstein, G.L., Perkel, D.H. and Dayhoff, J.E. (1985), Cooperative Firing Activity in Simultaneously Recorded Populations of Neurons: Detection and Measurement, *Journal of Neuroscience* 5(4), 881-889.

[9] Baker, S.N. and Gerstein, G.L. (2000), Improvements to the Sensitivity of the Gravitational Clustering for Multiple Neuron Recordings, *Neural Computation* 12, 2597–2620.

[10] Stuart L., Walter M. and Borisyuk R. (2002), Visualization of synchronous firing in multi-dimensional spike trains, *BioSystems* 67:265-279

[11] Marocco, D., Stuart E.J. and A. Cangelosi, Visualization techniques in artificial life: The method of parallel coordinates, *1st Italian Workshop on Artificial Life*, The University of Calabria, September (2003).