Visual Analytics Interface Design for Parameter Optimization

A. Johannes Pretorius
University of Leeds
Leeds, LS2 9JT, UK
a.j.pretorius@leeds.ac.uk

Roy A. Ruddle
University of Leeds
Leeds, LS2 9JT, UK
r.a.ruddle@leeds.ac.uk

ABSTRACT
Visual analytics systems aim to transform data into knowledge by integrating visualization and automation. We present a classification of the elements of the parameter optimization problem from a perceptual perspective. To illustrate its utility, we apply this classification to the design of a visual analytics interface for parameter optimization of biomedical image analysis algorithms.

Author Keywords
Visual analytics, information visualization, interface design, model refinement, parameter optimization.

ACM Classification Keywords
H.1.2 Models and principles: User/machine systems; H.5.2 Information interfaces and presentation: User interfaces; I.3.6 Computer graphics: Methodology and techniques.

General Terms
Design, human factors.

INTRODUCTION

Visual analytics research places emphasis on how interactive visual interfaces facilitate sense-making [7]. Keim et al characterize the visual analytics process as a transformation of data to knowledge by integrating visualization and automation [2]. Automatic data analysis methods process input data and users visually analyze the generated output to discover context-sensitive knowledge.

In visual analytics systems, iteration is essential: users investigate alternative outputs and refine processing algorithms by adjusting parameters. The aim is to achieve a symbiosis between users' domain knowledge and visual capabilities, and the processing power and interactive graphics capabilities offered by technology. For parameter optimization there are often many possible parameter values, leading to a large parameter space to investigate.

In this paper we formalize our experience of designing a visual analytics interface for parameter optimization. We first classify the problem space from a perceptual perspective. Next, we apply the classification to the design of a visual analytics interface for parameter optimization of image analysis algorithms for biomedical photomicrographs [4]. We argue that our analysis and conclusions generalize to other spatial datasets in biomedicine (for example, MRI and histopathology) and beyond (GIS and astronomy).

VAW 2011, September 6–8, 2011, University College London, UK.

CLASSIFICATION

A visualization system displays data on a 2D array of pixels. The number of pixels varies from a few hundred thousand (smart phones, tablets) to a few million (desktop workstations) to tens or even hundreds of millions (wall-sized gigapixel displays). Display resolution dictates the amount of information that a user can see at any time. In turn, this affects the rate at which users can comprehend the effect of different parameter settings on input data.

The driving challenge for parameter optimization is to understand processed output in the context of the input data and the parameter settings used to produce it. For instance, in an image analysis context, users have input images and seek to arrive at appropriate parameter settings by inspecting the output of the analysis algorithms.

Input

The amount of the input data that can be considered at any time on a display of given resolution is affected by three factors: data dimensionality, data size and number of inputs.

Data dimensionality. 2D data can be shown in its entirety (with limitations, as noted below). With 3D data a given view shows just one projection, but generally users can adjust that projection to see any other part of the data. With high-dimensional data it is not practical, and often not possible, for users to view all projections of the data, even over an extended period of time.

Data size. The greater the size of each input data set (in pixels) the less can be seen in a given projection at a given display resolution. For example, some 2D imaging data can be shown at native resolution even on a mobile display. At the other extreme, even a gigapixel display can only show a fraction of an image with the resolution common in histopathology, remote sensing, and astronomical sky surveys [8].

Number of inputs. The greater the number of input datasets the lower the resolution that each can be shown at if all are to be seen together (for direct comparison, for example).

Parameters

It is often a challenge to represent an overview of parameter space due to its combinatorial nature. This is influenced by individual parameters' data type, sample distribution, and the number of parameters being considered.

Data type. There are three data types to consider: nominal, ordinal, and numerical [6]. Ordinal and numerical data have an implicit ordering that users will also expect in a visual representation of it [9]. Nominal data depict categories with an arbitrary ordering. This introduces another aspect to recognize: assistive algorithms, such as sorting, are likely to result in more fragmented output. Perceptual patterns in those outputs will be harder for users to discern.

Sample distribution. Parameters are often uniformly sampled. This has the advantage that representations of intervals between succeeding values can be compact. If sampling is not uniform, however, this increases the real estate necessary to display the parameter space if distances between values are to be accurately discerned by the user [9].

Number of parameters. The number of parameters places exponential demands on the space needed to display unique points in parameter space, and their relationships to output, for a given sample distribution.

Output

In the scope of this paper, we are interested in outputs related to spatial input data, for example, the cells identified in biomedical images. Representing output will depend on the output modality, how users will evaluate these outputs, and how many outputs there will be to evaluate.

Modality. There is a range of output modalities possible. First, an output can be a spatial image containing annotations, for example, input images annotated with outlines of cells that have been identified by an image analysis algorithm. This would require users to inspect image-based output that requires space proportional to the input data size (in pixels). Output may also be in the form of derived metric data, which includes single-valued output per input - such as the number of cells identified - or multi-valued output indexed to multiple parts of the input image - such as the area covered by each cell. The latter increases the detail to show in an output, but may also allow areas of interest to be identified so that only parts of the input data, annotated with outputs, need to be shown. This reduces the display area needed.

Evaluation. There are two approaches for evaluating output. Subjective evaluation involves users inspecting output qualitatively, by sight, to judge whether it is good, bad, or somewhere in between. As noted above, this usually requires space proportional to the input data size per output. Objective judgment is based on quantitative output. Typically, users will judge output based on a derived metric that could be a scalar or multi-dimensional (see Modality, above). A scalar, such as the number of cells detected, may be encoded compactly within the parameter representation itself, but representing a higher-dimensional output is more challenging [10].

Number of outputs. Generally the more outputs there are, the harder they are to portray due to restricted screen real-estate. This often leads to a visualization challenge in its own right. On the other hand, with a larger number of outputs it may be possible to compute meta-metrics to filter, sort, or cluster inputs and/or parameters to reveal patterns.

INTERFACE DESIGN

Scientists routinely apply automated image analysis algorithms to identify objects, such as cell nuclei, in biomedical images. Image analysis algorithms are often highly parameterized and significant human input is needed to optimize parameter settings. The conventional approach is "parameter tweaking" by trial and error. Users start with a set of test images as input. They then supply parameter settings, initialize and wait for algorithms to execute, change the input settings, and repeat the process until satisfactory outputs are achieved. Outputs are judged subjectively by visual inspection. When users are satisfied with the output, they apply the algorithms and parameter settings to large collections of images similar to their test set. Conventional parameter optimization can take hours, or even days, to complete.

To support parameter optimization for image analysis, we developed a prototype called Paramorama (see Figure 1) [3, 4]. We also developed a plug-in for CellProfiler [1], which samples parameter space and generates input data for our tool. Paramorama has three display areas dedicated to input, parameters, and output (Figure 1(a), (b), and (c), respectively). The challenge was to display these elements on a desktop monitor to assist users with parameter optimization.

Our work to date has focused on input images analyzed in biomedical laboratories, which are typically a few hundred pixels squared. Target users are generally interested in optimizing algorithm parameters for a handful of such images. Inputs are displayed as a list of images at the top right of the display, as shown in Figure 1(a). Here, the input are two micrographs of human HT29 colon cancer cells. The users' goal is to identify suitable parameter values for an image analysis system to accurately detect the cell nuclei in these images. Three samples each have been generated for five parameters (Figure 1(b)) of the algorithm used to identify cell nuclei. A user has selected three regions of parameter space (outlined in blue in Figure 1(b)), for which output images are shown in the central view (Figure 1(c)).

A typical image analysis system contains 150-200 parameters. However, many of these are only set once. Users direct most of their effort to optimizing a small number of numerical parameters, particularly those responsible for detecting objects. At any point in time only 3-7 parameters are being optimized, but the process remains time-consuming and can take several days to complete. Paramorama is typically used with several hundred combinations of values for these para-

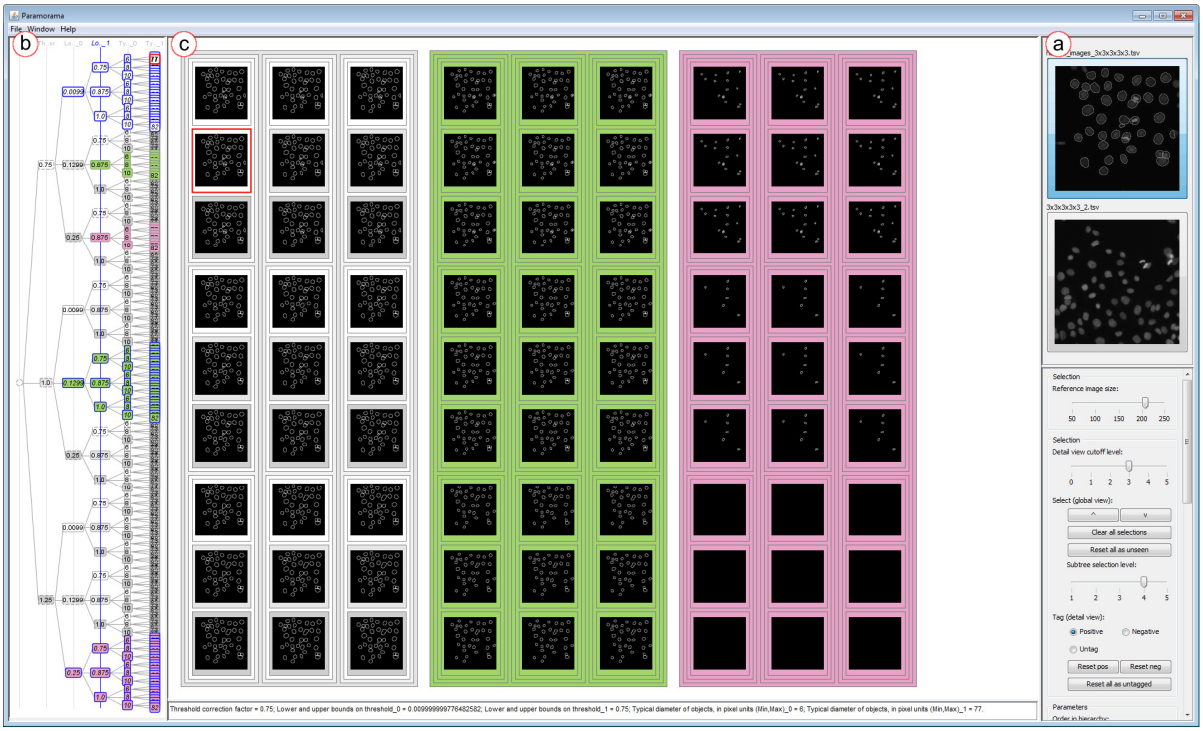

Figure 1. Visual analytics interface for parameter optimization. (a) Input images are shown at the top right of the display. (b) A structured overview of parameter space is shown at the left. Users inspect output produced for the currently selected input image by interacting with this parameter overview. (c) The output for user-specified parts of parameter space are shown as thumbnails in the central view. When users move their mouse over outputs in this view, they are displayed at native resolution in the input view instead of the input image. Tagging is an important feature and users can tag high- and low-quality output using the central view. Tagged output are color-coded in green and magenta, respectively, in the output and in the parameter space views. By tagging output, it is possible to identify contiguous regions of parameter space that yield high- and low-quality output.

meters, sampled at regular intervals by a CellProfiler plug-in (see above). This means that sampled parameters have ordinal domains.

An important design challenge was to provide users with a structured, navigable overview of the sampled parameter space. Parameter space is multidimensional, but discrete (due to sampling) and there are many existing multidimensional visualization techniques we could potentially have used [10]. Because there are few parameters (3-7) and few samples (3-6), we identified dimensional stacking as a solution. Parameter space is represented as a hierarchy: the first parameter can assume one of m values, for each of these, the second parameter can assume n values, and so forth (see Figure 1(b)). Such a structure can be interpreted accurately in terms of the parameter values it represents. Consequently, it is intuitive to navigate.

We also had to choose an appropriate layout for the tree that represents parameter space. We considered three options: directed (see Figure 1(b)), radial (Figure 2), and nested (treemap). Table 1 summarizes our comparison of these layouts. We dropped nested layouts, as both directed and radial layouts show relationships more explicitly (using connection versus containment).

To free as much space as possible for the output view, it was important to have a compact parameter representation. As Table 1 shows, directed layouts are more compact than radial ones: for a large number of leaves the numerator p^2 will dominate in the case of a radial layout. We note that radial and nested layouts are often lauded for their "ideal" aspect ratios. However, as the overview of parameter space was to be placed in an elongated area, the typically extreme aspect ratio of a directed layout is actually advantageous.

Although nodes in both a directed and radial layout can be arranged to reflect the order of the sampled values (which are ordinal), this can be shown more explicitly from top-to-bottom in a directed layout (see Figure 1(b)). Due to uniform sampling, we did not have to take account of relative distances between sibling nodes in the layout.

The number of outputs is dependent on the number of inputs, the number of parameters, and the number of samples taken per parameter. For every unique combination of parameter values and input image, our users consider a single output image. Outputs contain the outlines of cells that have been detected. These outputs are evaluated subjectively, by visual analysis, but the modality is not entirely trivial: for cell detection, users are interested both in how many cells were detected and where they were detected. Users particu-

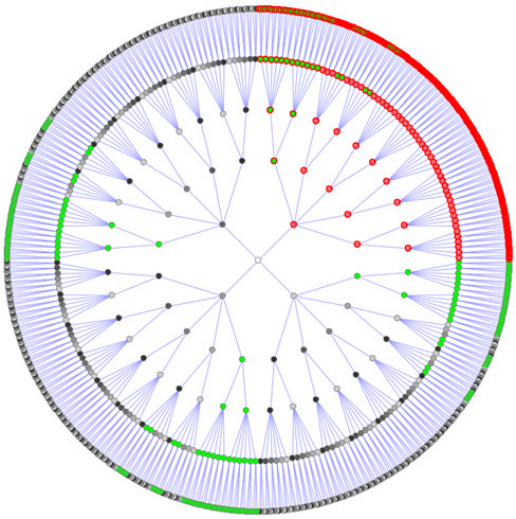

Figure 2. Radial layout of parameter space.

larly appreciate help with the visual comparison of output: this is their preferred method of analysis, but is currently done very inefficiently. We consequently decided to allocate the largest area of the display to output. In Figure 1(c), output images are displayed for 81 samples of the parameter space.

CONCLUSION

In this paper we have outlined a design approach for a visual analytics interface for parameter optimization. We have classified, from a perceptual perspective, the input, the parameters, and the output and have applied this classification to a case study. For more details on our work on parameter visualization, including related research, detailed case studies, and discussions of the scalability and limitations of our chosen design, the interested reader is referred to our previous article [4].

The perceptual analysis of our problem space suggests a number of extensions to our current work. To also cater for 3D input, our input view could be adapted to incorporate a spreadsheet interaction device, similar to VisTrails [5], where interacting with one input also changes the vantage point of all others. To cater for larger input data (such as histopathology images [8]), higher input dimensionality, and increased input size we are also eager to test our approach on wall-sized gigapixel displays.

ACKNOWLEDGMENTS

We thank Dr Anne Carpenter, Dr Mark Bray, and colleagues at the Broad Institute of MIT and Harvard for their invaluable input. This work was funded through WELMEC, a Center of Excellence in Medical Engineering funded by the Wellcome Trust and EPSRC, under grant number WT 088908/Z/09/Z.

	Directed	**Radial**	**Nested**
Relationships	Connection	Connection	Containment
Aspect ratio	Typically p:q; $p << q$	1:1	Typically close to 1:1
Area	$p \times q$	q^2/π^2	Approaches $\sqrt{q}$

Table 1. Comparison of different layout techniques for a parameter space with p parameters and q combinations of parameters. Each combination is represented by a node of unit size with zero inter-node spacing.

REFERENCES

1. Imaging Platform, The Broad Institute of MIT and Harvard. *CellProfiler developer's version website,* http://www.cellprofiler.org/developers.shtml, 2011.
2. Keim, D., Kohlhammer, J., Ellis, G. and Mansmann, F. (Editors) *Mastering the Information Age Solving Problems with Visual Analytics,* Eurographics Association, Goslar, Germany, 2010.
3. Pretorius, A.J. *Paramorama website,* http://www.comp.leeds.ac.uk/scsajp/applications/paramorama/
4. Pretorius, A.J., Bray, M.-A.P, Carpenter, A.E. and Ruddle, R.A. Visualization of parameter space for image analysis, *IEEE Transactions on Visualization and Computer Graphics,* In press.
5. Bavoil, L., Callahan, S.P., Crossno, P.J., Freire, J., Scheidegger, C.E., Silva, C.T. and Vo., H.T. VisTrails: enabling interactive multiple-view visualizations, *Proceedings of IEEE Conference on Visualization,* 2005.
6. Spence, R. *Information Visualization: Design for Interaction.* Pearson Education, Second edition, 2006.
7. Thomas, J.J. and Cook, K.A. (Editors) *Illuminating the Path: The Research and Development Agenda for Visual Analytics,* National Visualization and Analytics Center, Richland, WA, USA, 2005.
8. Treanor, D., Jordan-Owers, N., Hodrien, J., Quirke, P. and Ruddle, R.A. Virtual reality powerwall versus conventional microscope for viewing pathology slides: an experimental comparison. *Histopathology 55,* 3 (2009), 294-300.
9. Ware, C. *Information Visualization: Perception for Design.* Morgan Kaufmann, San Francisco, CA, USA, 2000.
10. Wong, P.C. and Bergeron, R.D. 30 years of multidimensional multivariate visualization. *Scientific Visualization - Overview, Methodologies, Techniques.* IEEE CS Press, Washington, DC, USA, 1997.