

3D Kiviat Diagrams for the Interactive Analysis of Software Metric Trends

Andreas Kerren
andreas.kerren@lnu.se

Ilir Jusufi
ilir.jusufi@lnu.se

Linnaeus University, DFM
SE-351 95 Växjö, Sweden

ABSTRACT

Previous techniques for visualizing time-series of multivariate data mostly plot the time along additional axes, are often complex, and does not support intuitive interaction. In this poster paper, we present an interactive visualization approach for the analysis of software metric trends that allows users to operate with Kiviat diagrams on 2D planes in the space and to intuitively extend this visual representation into 3D if needed.

Categories and Subject Descriptors

I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction Techniques; D.2.8 [Software Engineering]: Metrics—complexity measures, performance measures

General Terms

Design

1. INTRODUCTION

The visualization of time-dependent, multivariate data sets is seen as one of the most important challenges in information visualization and related fields [7]. Although time can be simply considered as an additional attribute, it can also be treated separately to underline its special importance. There are various approaches in the literature that address this problem. Many approaches use the third dimension in their visual representations to extend standard visualization techniques for multivariate data sets with an additional time axis. Prominent examples are Kiviat Tubes [4] or 3D Parallel Coordinates [5]. The resulting visualizations are relatively complex, are difficult to interact with, also use 3D if it is not needed, and cannot be smoothly embedded into other visual representations.

Currently, the input data for our approach is mainly based on measurements in software engineering, i.e., *software metrics* [2], but any time-series of multivariate data could be used instead. From the perspective of the visualization community, the measured values of a whole software metric suite form a multivariate data set. As the development of software is an evolutionary process where different versions of the source/binary code are typically stored in large repositories, we must take this time-dependency of software metrics into account. Thus, important trends of these metrics should be highlighted in some way to give the software developers the opportunity to react in short time.

Copyright is held by the author/owner(s).
SOFTVIS'10, October 25–26, 2010, Salt Lake City, Utah, USA.
ACM 978-1-4503-0028-5/10/10.

In this paper, we present a novel visualization approach that acts in accordance with the following mantra: use a 2D visual representation as long as possible and extend to 3D if really needed (e.g., to solve conceptual drawbacks of the 2D approach). The dimensional gap between both representations is bridged by smooth interaction using a *fanning-out metaphor*. Our visual representation extends Kiviat diagrams as presented in the next subsection with 3D interaction.

1.1 Previous Approach

Pinzger et al. [8] had the idea that Kiviat diagrams can be used to visualize software metrics gathered from several releases of source code. Thus, they developed an improvement of a standard Kiviat diagram in order to facilitate the visualization of multiple software releases and of release history data. Each value of a metric is measured across n releases and plotted in the diagram. Adjacent measures of metrics belonging to the same release are then connected via polylines. The resulting diagram consists of polygons for each release. However, the values of different metrics can also decrease from release to release. This results in a polygon overlapping that hides information, as our screenshot example in Figure 1 shows in the focused area at the right hand side.

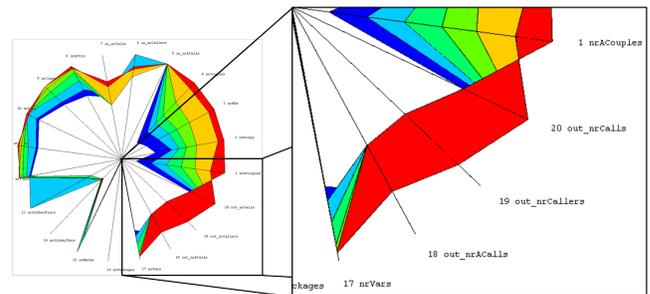


Figure 1: Screenshot example of the Kiviat diagram based on the work [8]. The metric values are hard to perceive due to overlaps in some parts of the diagram (e.g., for the metric IDs 18, 19, and 20).

2. INTERACTIVE 3D KIVIAT DIAGRAMS

A first idea of a solution could be to use transparencies in order to overcome that problem. But considering the fact that individual releases are color-coded, this approach is not very useful: by introducing a transparency, overlapped colors will change (e.g., yellow over blue will yield green) which makes the values hard to perceive.

We decided to use 3D on the contrary (further 3D approaches in software visualization were discussed in [9]). Instead of placing

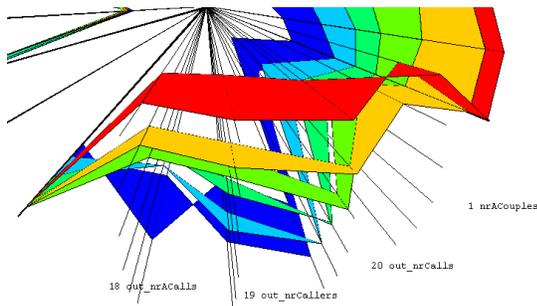


Figure 2: Users can fan out specific metric axes using the mouse wheel in order to analyze hidden information caused by overlaps (only the underpart of our 3D Kiviat diagram is displayed).

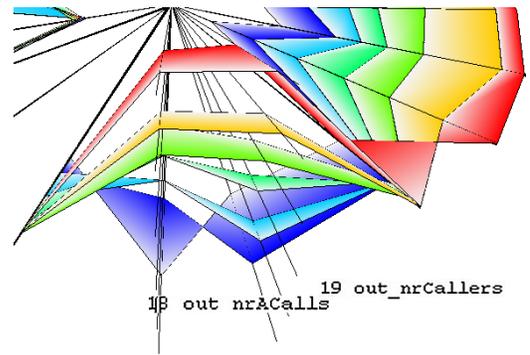
the Kiviat diagrams in a kind of stack, as discussed in the work of Fanea et al. [1], we developed an *fanning-out metaphor* which is intuitive and space-filling [3, 6]. We can interact with this diagram by *fanning-out* specific metrics axes into the third dimension in the way shown in Figure 2. This is done by selecting axes with the left mouse button and using the mouse wheel to smoothly fan them out. Hereby, each selected axis a is multiplied into N_a coplanar axes with the same origin in the center of the diagram. N_a corresponds to the number of releases, i.e., each release polygon gets its own axis level for the selected metric axes. Thus, we can examine specific metrics per release, even those, which were previously hidden by overlaps. Of course, users can rotate, zoom in/out and pan the diagram in order to get the best possible view.

Sometimes, even when our fanning-out technique is used, the direction of value changes is hard to perceive in 3D. Users have troubles to decide whether the value has increased or decreased. In order to avoid this, we introduced a gradient color coding between the releases. The darker side of the gradient shows the positive direction of a change. Thus, it is easier to see if a change is negative or positive, cf. Figure 3(a). In some cases there is also a need to filter out some metrics/releases and to focus on specific ones. Our approach supports the use of transparency to hide unimportant metrics and/or releases. One possible result of this kind of interaction is shown in Figure 3(b). Here, the metric values are highlighted as solid-colored line segments on the fanning-out axes.

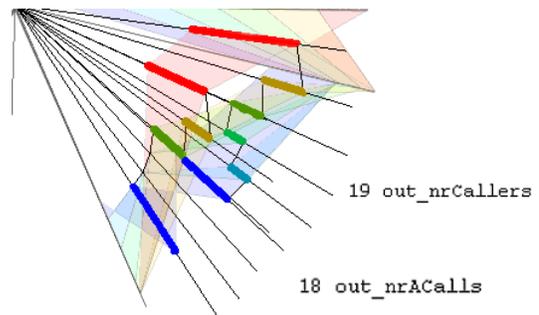
Our 3D Kiviat diagram approach using the fanning-out metaphor, combined with the gradient and transparency features, gives a possibility to see the data hidden by the overlaps and can be integrated into more complex visualization approaches, such as network visualizations.

2.1 Future Work

In this poster paper, we presented a novel 3D approach, which can support the user to discover time-dependent properties within a multivariate data set. This is an ongoing project and need to be carefully evaluated. We performed a first usability study with our students in order to get some feedback about interaction and visual appearance [6]. Here, they had to compare 2D with our 3D Kiviat diagrams. First results show that our 3D interaction possibilities are intuitive, especially the fanning-out feature. On the other hand, the test persons needed more time to become familiar with the new metaphor. Besides a more formal evaluation and a larger case study on existing software systems, our future work will be directed to analyze the possibilities that this approach will offer, and to apply it for network analysis tasks. Currently, it is still a standalone tool.



(a) A gradient helps the user to determine the directions of value changes.



(b) The transparency makes it easy to filter out unnecessary data and to focus on specific ones.

Figure 3: Different cut-out views of selected metric axes.

3. ACKNOWLEDGMENTS

We would like to thank Guo Yuhua who implemented the first prototype.

4. REFERENCES

- [1] E. Fanea, S. Carpendale, and T. Isenberg. An Interactive 3D Integration of Parallel Coordinates and Star Glyphs. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 149–156, 2005.
- [2] N. E. Fenton and S. L. Pfleeger. *Software Metrics: A Rigorous and Practical Approach*. Course Technology, 2nd. edition, 1998.
- [3] Y. Guo. Implementation of 3D Kiviat Diagrams. Bachelor’s thesis, Växjö University, Sweden, 2008.
- [4] S. T. Hackstadt and A. D. Malony. Visualizing Parallel Programs and Performance. *IEEE Comput. Graph. Appl.*, 15(4):12–14, 1995.
- [5] K. Honda and J. Nakano. 3 Dimensional Parallel Coordinates Plot and Its Use for Variable Selection. In A. Rizzi and M. Vichi, editors, *Proceedings of Compstat 2006 - Proceedings in Computational Statistics*, pages 187–195. Physica-Verlag HD, 2006.
- [6] A. Kerren and I. Jusufi. Novel Visual Representations for Software Metrics Using 3D and Animation. In I. J. Münch and P. Liggesmeyer, editors, *Software Engineering 2009 - Workshopband*, volume P-150, pages 147–154. Kaiserslautern, Germany, 2009. GI-Edition.
- [7] R. S. Laramee and R. Kosara. Challenges and Unsolved Problems. In A. Kerren, A. Ebert, and J. Meyer, editors, *Human-Centered Visualization Environments*, LNCS Tutorial 4417, pages 231–254. Springer, 2006.
- [8] M. Pinzger, H. Gall, M. Fischer, and M. Lanza. Visualizing Multiple Evolution Metrics. In *Proceedings of the ACM Symposium on Software Visualization (SoftVis ’05)*, pages 354–362, St. Louis, Missouri, 2005.
- [9] A. R. Teyseyre and M. R. Campo. An Overview of 3D Software Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 15(1):87–105, 2009.