Spatio-Temporal Visualization of Battlefield Entities and Events

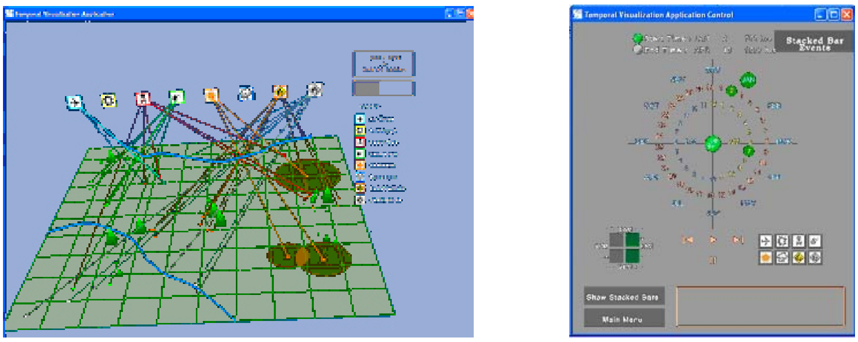
Qiyue Fong[1], Foo Meng Ng[2], Zhiyong Huang[1]

[1] School of Computer Science, National University of Singapore
fongqiyue@gmail.com
huangzy@comp.nus.edu.sg
[2] Human Factors Laboratory, DSO National Labs, Singapore
nfoomeng@dso.org.sg

Abstract. In this work, we address visualization of spatio-temporal data for military application. Four different visualization prototypes have been developed to track the movement of military entities across a land surface over time; three more have been developed to track the occurrences of numerous war events. We have implemented the prototypes in a software system with a novel clock face GUI. Usability tests have been carried out and confirmed the effectiveness of the solution.

1 Introduction

As time progresses during warfare, it becomes increasingly difficult for commanders to manually analyze and spot patterns inherent in large, complicated sets of battlefield data. Limitations of human memory and our inability to compute complex calculations simply prevent us from performing such complex tasks. It may, perhaps, be wise to make use of the powerful computing capabilities of a computer and visual computer graphics to circumvent the problems stated above. The goal of this work is thus to research and explore existing solutions which capture and visualize the spatial relationships of battlefield entities over time within a single interactive picture. Four different visualization prototypes have been developed to track the movement of military entities across a land surface over time; three more have been developed to track the occurrences of numerous war events (one snapshot in Figure 1(a)). A novel clock face GUI has also been designed and implemented to help users of the software query the system with time constraints (Figure 1(b)). Usability tests have also been carried out on all the visualization prototypes as well as the Time Query GUI to determine their applicability and usefulness and to uncover any potential human factors problems.



(a) (b)

Figure 1. Illustration of the graphics display (a) and clock face GUI (b)

2 Related work and our contributions

Different existing techniques which visualize spatio-temporal data are found in [1], [2], and [4]. In [5], a common thematic map known as the choropleth map, can represent different magnitudes of an attribute which changes over time using shaded units. Also, the Z-Axis in 3D space can be used to represent time, a technique used in [3]. Space represented by the first and second dimensions combined, while time by the third dimension in a 3D space is an innovative idea.

Our work adopts and improves on the techniques described above to serve our purposes. In general, it extends all existing ideas in 2D into 3D; and conducts experiments with all the aforementioned advanced techniques. It results in a plethora of spatio-temporal visualization prototypes.

Unlike most existing work, this work attempts not only to explore and describe the various ways of visualization, but also allows potential users to compare and contrast these various techniques through a single application and then determine empirically how preferred each technique is (or if there was simply no preference) over the others.

3 Our work

3.1 The clock face GUI

To the best of our knowledge, most time query user interfaces make use of widgets or visual metaphors such as the timeline/time slider, time dials and calendar controls to form a time query. Each of these has limitation(s) that fail to serve the purposes of the temporal data being dealt with here. Temporal data in the military context is precise and accurate. Timelines and time dials do not allow selection of such precise values. A full calendar is space-wasting. Although a dynamic calendar (such as Microsoft's calendar) is space-saving, it only allows selection of a date but not time. Time can only be selected by clicking the "increase" or "decrease" button multiple times which is inefficient. This motivated us to design a novel time query interface, the clock face GUI (Figure 1(b)), that can overcome these limitations to serve our purposes.

The clock face GUI combines circular structures with the conventional calendar resulting in a compact, space-saving UI. It consists of three concentric circles each representing a particular time unit (Month/Day/Hour). Each circle is color coded to guide the user. Two grid lines intersecting at the center of the clock interface divides it into four quadrants to reduce search times for a value.

The clock face GUI together with other controls, form the complete interface to interact with the main display. It allows selection of 1) time moment, 2) time interval, 3) time range within a day, 4) battlefield events; and also animation playback functions.

3.2 The spatio-temporal visualization prototypes

We describe the seven spatio-temporal visualization prototypes here. Four track locations (movement) of enemy battlefield entities that change over time while three track occurrences of battlefield events over time. The term "battlefield entities" refer to army units. Without loss of generality, we present the enemy data only.

Dot graph: It provides a snapshot of the distribution and strength of each enemy unit across the terrain at a particular time moment (Figure 2).

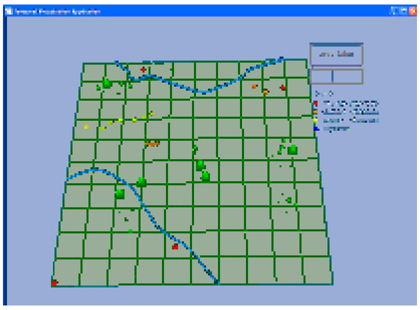


Figure 2. The dot graph

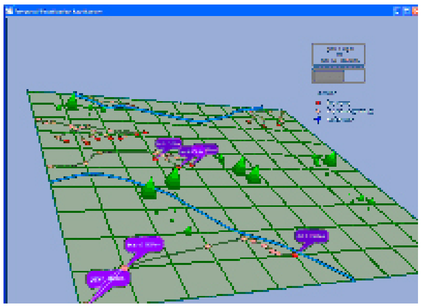


Figure 3. The trail graph (with time tags)

Dots represent enemy units marking their locations on the terrain at some time moment, t. Each dot is colored which describes the strength of each unit. Dots on the map change color and increase in number over time due to the splitting of a unit. Such colored dot maps, hence, are an instance of a choropleth map.

Trail graph: It gives a snapshot of both the current as well as historical locations of each enemy unit in a given time interval. Dots mark a unit's location on the terrain. Curved lines (trails) connect a unit's historical and current locations together, showing movement. Trails approximate the path that a unit moves along. It solves a limitation posed by Dot Graph (ie. inability to show historical information).

This prototype, hence, is an instance of the 2D line plot graph extended into 3D. Users may click on the dots to display time tags to reveal time associated with each dot. Plotting these time tags on the terrain is analogous to the scatter plot graph. With these time tags, it is possible for user to estimate the speed at which a unit is moving.

Spoke graph: It functions like the trail graph with some enhancements. A spoke is a rod-like structure that extends along the y-axis in 3D space. Spokes were initially used to solve a drawback posed by the Trail Graph – inability to distinguish between two units that traveled along the same path at different times.

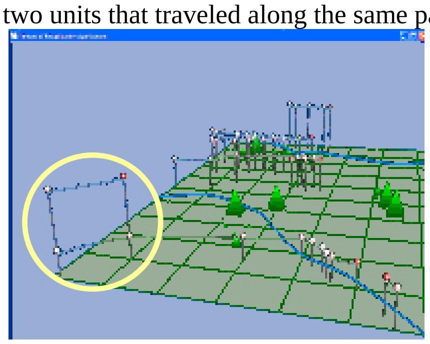


Figure 4. Spokes to differentiate the enemy units

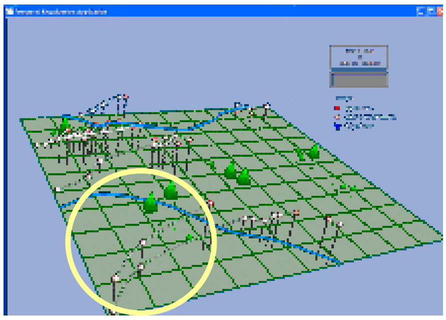


Figure 5. Spokes to depict time value

Figure 4 shows how each dot along any trail is elevated by a spoke at a fixed height. The height of each spoke was used to identify one enemy unit (moving along a trail) from another (moving along another trail). As such, two different units (circled) moving along the same path can be distinguished.

However, it may not seem obvious that height is used to differentiate units. Hence, the spokes idiom was modified to depict the time value that an enemy unit was found located at some position on the terrain (Figure 5).

From Figure 5, each spoke drawn along a unique trail now take on different height values. User can have an intuitive feel of how much time has elapsed before a particular unit has moved on to a new location and to predict its speed. The spokes still preserve its ability to differentiate among units moving along the same path (circled).

Raster graph: Multiple snapshots of locations of enemy units at successive time moments (Figure 6) is shown. Raster is a horizontal translucent plane in 3D space. This idea came from 2D chess maps.

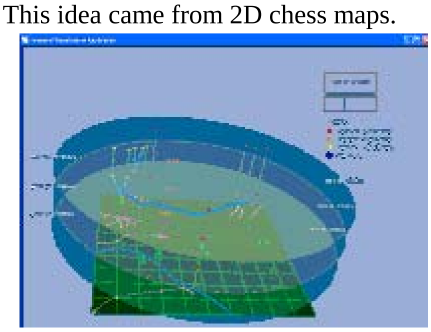


Figure 6. The raster graph

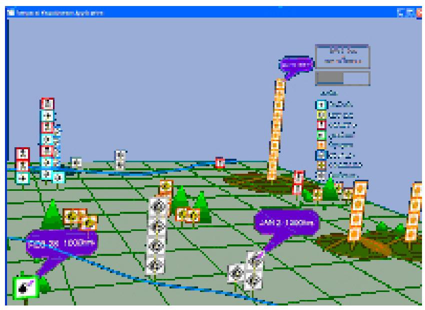


Figure 7. The stacked bar graph

Each raster offers a snapshot of the locations of each enemy unit at a particular time. Vertical grey lines drawn through the rasters connect units representing the same army unit on each raster. Cyan colored lines depict splitting behavior. Rasters are displayed in successive times, allowing comparison of the movement and/or splitting behavior of the enemy units between two or more successive times. Such comparison is useful in seeing if any unit has remained in its position for a long time by looking down at the line projected from the unit in space to the unit on the map. If the unit is found on each n raster along this projected line, it means that it has not moved/split for the past n successive times.

The following three graphs are designed to track the occurrences of battlefield events.

Stacked bar graph: Stacked Bar Graph retrieves battlefield events found to be associated with a time moment or interval. Then, they are stacked in a vertical bar (Figure 7). Each vertical bar "holds" all the events that occurred at the same spatial location at a particular time moment or interval. In other words, it is able to convey what events occurred and where they occurred at some time moment or interval. To some extent, this prototype was inspired by the 2D bar graphs

Such a simple representation is capable of conveying lots of useful information. First, it can depict event recurrence at a location by observing the number of icons found in a stacked bar. Second, the events found in a stacked bar are sorted according to time, thus, the order of events occurring at a location can be inferred. Third, by clicking on an event icon in stacked bar, time tags are displayed. Time tags allow the inference of causality relationships between events.

Grouped links graph: Grouped Links Graph is capable of showing the distribution of events across a land surface effectively. It does so by grouping links that link to the same type of events on the terrain together (Figure 8). Links are lines joining an event icon to a particular location on the map. These links have the same color as that associated with the event icon it is joined to and are thus associated with the event that the event icon refers to. A link represents an occurrence of the event it is associated with. A thicker link means there are more than one occurrence of an event.

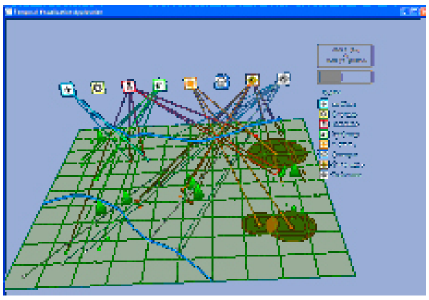


Figure 8. Grouped links graph

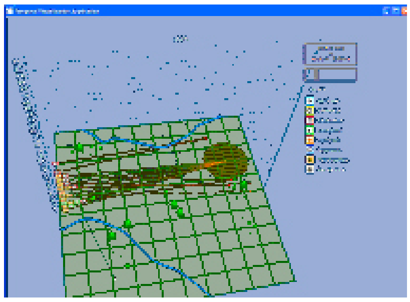


Figure 9. The calendar graph

One useful feature of this visualization technique is that comparison of the number of occurrences of an event with another can be made intuitively (from the degree of the "meshing" of links) without having to count the exact numbers.

Calendar graph: It retrieves data about battlefield events found to be associated with a given time moment or a month and plots them onto a calendar drawn perpendicular to the map (Figure 9).

The calendar displays all the days of a month in rows. All event icons found in a row on the calendar are sorted according to time. This graph was inspired by GeoTime[3]. Some modifications were made to GeoTime to adapt to the type of data this project deals with.

In GeoTime, event icons are drawn directly above the location at which they occurred on the map. The height of the event icons corresponds to the date on the calendar. Two drawbacks can be observed in GeoTime. First, when the main display is rotated, it becomes hard to determine which date each event icon corresponds to because no link is drawn to connect the icon to the calendar. Second, GeoTime is unable to represent the scenario where two different types of events are occurring at the same location and at the same time. This is because only one event icon can be drawn above a location and mapped to a date on the calendar at any one time. The Calendar Graph attempts to improve on these drawbacks to serve its own purposes.

To solve the first drawback, event icons are directly plotted onto the calendar. To solve the second drawback, different types of events that occur concurrently at the same locations are represented by their respective icons plotted side by side on the same row in the calendar. Each of these icons is connected by a colored line to the location where it occurred. With a calendar, time information becomes obvious. It is like a neat report showing which events have occurred and at what date/time.

4 Usability study

Our usability tests aim to determine 1) ease-of-use of the clock face GUI, 2) effectiveness of each prototype in conveying spatio-temporal information and 3) to uncover any potential human factors problems inherent in the system. A test methodology described by [6] was adopted for the tests.

Medium: Low-interaction method was used in tests on the clock face GUI to determine if the participants were able to complete certain tasks without any help or prior training. A mixture of low- and high-interaction methods was used in the tests on the prototypes to collect as many different interpretations of the visualizations as possible to determine if they are interpreted according to the original intentions.

Test specifications: A test has two participants: the test participant and the tester. Each test participant has to complete 10 test cases followed by a mini questionnaire. The test cases are briefly described as follows:

<u>Test Cases 1 - 2 on the clock face GUI</u>: They aim to determine the ease of use of the clock face GUI. For each task in a test case, the participant selects a given time query. Time taken for the participant to do a correct selection is measured using a stopwatch and recorded.

<u>Test Case 3 on the complete clock face GUI</u>: This aims to determine how easily the complete clock face GUI can be manipulated to do a given list of tasks.

<u>Test Cases 4 -10 on the 7 spatio-temporal visualization prototypes</u>: These require the participant to carry out a given task related to the prototype. The participant was then asked to describe what the visual graphics is that appear on the main display.

Choice of participants: Due to the classified nature of the project, only fellow colleagues and HYP students within the DSO organization were asked to participate. More information about the 8 participants is shown in Figure 10.

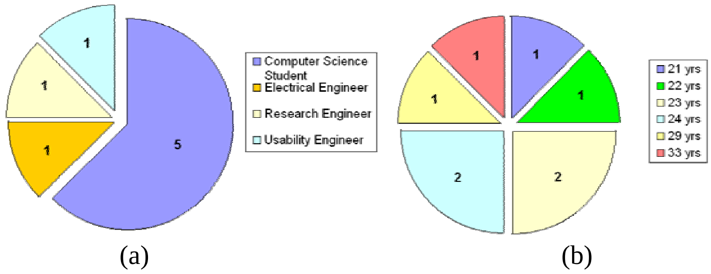


Figure 10. Participant information: (a) occupation and (b) age distribution

Test environment: Human Factors Engineering Lab at DSO@Kent Ridge. No audio or video recording was used.

Test data obtained are collated and analyzed empirically. To make these conclusions apply for a general population, a statistical method called the Student's T-Test for two independent samples was used. In Student's T-Test, a value known as the test statistic, t, is calculated and compared with a value, c, in the T-Table. For any two independent samples A and B, if $t < c$, then the null hypothesis (i.e. meanA = meanB) is accepted at some level of confidence. Otherwise, the null hypothesis is rejected (i.e. meanA < meanB or meanA > meanB, depending on the values of meanA and meanB from the data). From these results, we can make conclusions, at some level of confidence, about some observation that we are investigating over a general population.

Summary of test results:

Test Cases 1-2: Time taken to select time units across 1, 2 or 3 quadrants in the clock face GUI appears to be the same. When a time interval was to be selected, time taken to do so generally increases by two to three times. Participants who managed to figure out certain shortcuts in selection took less time – at least by half of the time taken by non-shortcuts.

Test Case 3: Participants did not experience any difficulties using the complete clock face GUI. Positive responses were given to the clock face GUI in terms of aesthetics, ease-of-use, intuitiveness, effective feedback and usefulness.

Test Cases 4-10: Generally, all test participants gave interpretations for each of the 7 prototypes similar to the original intentions of the visualization design. Participants felt that all prototypes were useful and informative, but not necessarily communicative or intuitive. This concurs with the observation that participants usually take some time to study the visualization before giving their interpretations. However, once the "studying" stage was over, comments came flowing in. ${}^{4}C_{2}$ binary comparisons were made between 2 prototypes from the set: Dots, Trails, Spokes and Rasters. At 95% confidence level, users across a population found Trail Graph more informative and useful than Dot Graph but expressed no preference over any prototype in the other binary comparisons. ${}^{3}C_{2}$ binary comparisons were made between 2 prototypes from the set: Stacked Bar, Grouped Links and Calendar. At 95% confidence level, users across a population found Calendar more informative and useful than Grouped Links but expressed no preference over any prototype in the other binary comparisons.

5 Conclusion

In this paper, we have proposed and implemented a software system to visualize spatio-temporal data of a battlefield. Four different visualization prototypes have been developed to track the movement of military entities across a land surface over time; while three more have been developed to track the occurrences of numerous war events over time. A novel clock face GUI has also been designed and implemented. Usability tests have also been carried out to determine the applicability and usefulness as well as to uncover any potential human factors problems.

References

1. Andrienko, N., Andrienko, G., Gatalsky, P. (2000). Visualization of Spatio-Temporal Information in the Internet. In Proceedings of the 11th International Workshop on Database and Expert Systems Applications, IEEE Computer Society, Washington, DC, USA, 2000, pp 577-585.
2. Elvins, T.T. (1997). VisFiles: Presentation Techniques for Time-Series Data, ACM SIGGRAPH Computer Graphics, New York, USA, Volume 31, Issue 2, May 1997, pp 14-16.
3. Kapler, T., Wright, W. (2004).Visualization for Tracking Battlefield Events in Time and Space for C2. 2004 Command and Control Research and Technology Symposium, 2004, pp 1-11.
4. Müller, W., Schumann, H. (2003) Visualization Methods for Time-Dependent Data – An Overview. In Proceedings of the 2003 Winter Simulation Conference, New Orleans, Louisiana, 2003, pp 737-745.
5. Slocum, T.A., McMaster, R.B., Kessler, F.C., Howard, H.H. (2004). In Thematic Cartography and Geographic Visualization. Prentice Hall Series, Upper Saddle River, NJ07458, 2nd Edition, 2004.
6. Weinschenk, S., Jamar, P., Yeo, S.C. (1997). In GUI Design Essentials. Wiley Computer Publishing, New York, USA.