Poster: Tweeting Visualizations for Collaborative Visual Analysis

Aidan Slingsby* Jason Dykes† Jo Wood‡

giCentre, Department of Information Science
City University London

ABSTRACT

We demonstrate a new approach to collaborative visual analysis, in which descriptions of graphics are shared online. HiVE – a high-level descriptive language – captures the way in which graphics are built and configured. HiVE statements and commentary can be micro-blogged. Compliant software clients can interpret HiVE, provide user-interfaces to allow users to explore data through graphical reconfiguration and can generate HiVE expressions that can be used in a collaborative context. We demonstrate this approach using examples from our work with the insurance industry.

Index Terms: E.3 [Data]: Coding and Information Theory—Formal models of communication; K.4.3 [Computers and Society]: Organizational Impacts—Computer-supported collaborative work

1 INTRODUCTION

'*vizTweets*' is a new approach to collaborative visual data analysis in which high-level descriptions of graphics are blogged or tweeted. We are engaging with diverse users, datasets and case studies through the Willis Research Network [4]. Analysts can share *descriptions* of the processes of visual design and analysis using the Hierarchical Visualization Expression (HiVE) language [2]. HiVE statements explicitly state how the graphic is constructed or configured through a notation that makes reference to data channels, but leave the precise geometry and aesthetics unspecified. HiVE can be generated and interpreted by software designed to suit the needs of the user, domain and hardware, whilst retaining the integrity of the information graphic. Rather than creating a new web infrastructure for collaboration and communication we use existing blogging mechanisms to communicate HiVE statements and commentary. This approach is in line with the aims of the JISC *Virtual Research Environment Rapid Innovation* programme that supports the development of virtual research environments through existing infrastructures and technology.

2 HIERARCHICAL VISUALISATION EXPRESSION (HIVE)

Existing languages for describing graphics capture geometrical and aesthetic qualities. HiVE is different in two important respects. Firstly, it emphasises the mappings between data variables and visual variables, leaving display and aesthetics to HiVE-compliant software. Secondly, it captures the reconfiguration of graphics as part of the data exploration process, important for collaboration.

HiVE is suited for describing hierarchical graphics where categorical variable values condition data at the next hierarchical level. This approach suits many types of graphic including scatterplot matrices, small-multiple maps (e.g. Fig. 2) and barcharts within maps (e.g. Fig. 5). This is achieved through of a *hierarchy of conditioning variables*. Variables or constants that control *layout*, *size*, *order/position*, *shape* and *colour* can be applied to each level.

*e-mail: a.slingsby@soi.city.ac.uk
†e-mail:jad7@soi.city.ac.uk
‡e-mail: jwo@soi.city.ac.uk

HiVE consists of *states* (preceded with 's') which describe the state of a graphic and *operators* (preceded with 'o') which describe how an existing graphic is reconfigured and always leads to a new state as demonstrated by the caption of Fig. 2 (states are omitted from subsequent figures). We have extended HiVE over the past year to meet needs of our users, including bar charts, dot maps, animation. Documentation is available online [1].

3 HIVE-COMPLIANT SOFTWARE CLIENTS

We envisage a diverse set of HiVE-compliant software clients to cater for different types of user, dataset and task. These interpret HiVE and then may: display appropriate graphical encodings of HiVE, provide a suitable user-interface with which to explore the data through interaction with the graphic and/or output HiVE so that the visualization can be shared. We encourage the use of our open HiVE specification and interfaces to produce new HiVE-compliant software to suit different needs. Since HiVE does not specify *aesthetics* [3], HiVE-compliant clients have some freedom in how to interpret HiVE. For example, we have produced a HiVE-compliant client that converts a HiVE statement into natural language. This may make HiVE more interpretable to a general audience and be more suitable for devices with limited graphics.

4 EXAMPLES AND APPLICATIONS

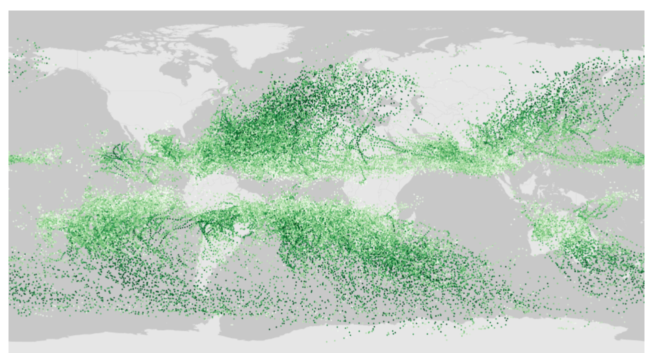

Figure 1: Map of hurricane tracks points, coloured by windspeed. *State:* `sHier(/,$pt); sSize(/,FIX); sShape(/,PNT); sOrder(/,[$lon,$lat]); sLayout(/,CA); sColor(/,$aWnd).`

We are using a variety of insurance-related datasets in our work. Here, we illustrate the approach using a dataset of hurricane tracks. This is typical of the kind of event-sets used by catastrophe (CAT) models, important tools in the insurance industry for estimating likely losses incurred through natural disasters.

Maps such as Fig. 1 are popular means of studying datasets of hurricane tracks. Dots are *coloured* by average windspeed (`$aWnd`). A Cartesian *layout* is used (`CA`) and track points are *ordered* (positioned) by longitude and latitude for the *x* and *y* axes. Fig. 2 shows how this varies by month as small multiples conditioned by year. This is obtained by *inserting* the month variable into the conditioning hierarchy, using a rectangular (`RCT`) *shaped*

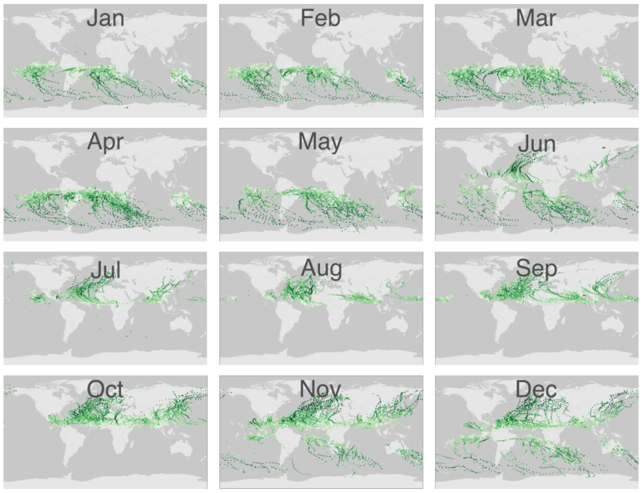

Figure 2: Small-multiple maps of track points by month.
Operator: `oInsert(/,1,$mn); oOrder(/,1,$mnOrder)`
State: `sHier(/,$mn,$pt); sSize(/,FIX,FIX); sShape(/,RCT,PNT); sOrder(/,$mnOrder,[$lon,$lat]); sLayout(/,SF,CA); sColor(/,Ø,$aWnd)`

space-filling (`SF`) *layout* and temporal *order*. Animation is a popular technique for showing change over time. In Fig. 3 we change the *layout* at level 1 to arrange the months over time (as an animation; `AN`). `oFocus(/Jun/)` changes the time-step to June. We have also *sized* the track points (level 2) by average vorticity (`$aVrt`).

Cutting the track points from the hierarchy so it only contains month, aggregates the data by month in Fig. 4. Bars are still *coloured* by average windspeed and ordered temporally, but are *sized* by the number of track points on the *y* axis and are of fixed width. To study geographical variation, the 'country' variable can be *inserted* into the base of the hierarchy and can be spatially *ordered* in a space-filling (SF), rectangular *layout*, where each country is shown at a fixed *size* (Fig. 5). Countries are coloured by the average windspeed of track points within.

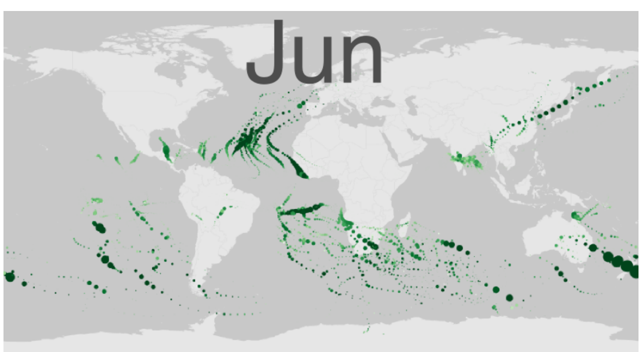

Figure 3: One frame (June) from a month-by-month animation, where points are sized by vorticity.
Operator: `oLayout(/,1,AN); oFocus(/Jun/); oSize(/,2,$aVrt)`

5 Conclusion and further work

This poster shows potential for collaborative data analysis using HiVE through virtual research environments based on existing blogging infrastructures. We use the insurance industry as a case study because of its reliance on collaboration and efficient knowledge dissemination amongst a diverse set of users. Other areas in

Figure 4: Number of track points by month, coloured by windspeed.
Operator: `oCut(/,2); oColor(/,$aWnd); oSize(/,1,[FIX,$numPt]); oShape(/,1,RCT); oLayout(/,1,CA)`

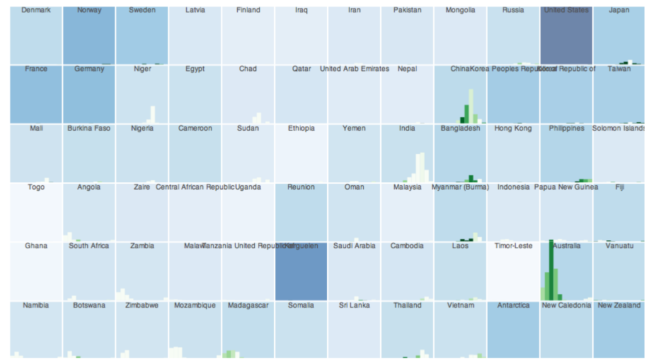

Figure 5: Barcharts by country; countries spatially arranged.
Operator: `oInsert(/,1,$cnty); oLayout(/,1,SF); oOrder(/,1,[$lon,$lat]); oColor(/,1,$aWnd); oSize(/,1,FIX)`

which we are working include model sensitivity analysis, demographics and property rebuild costs. We propose demonstrating a number of clients alongside our poster, including encodings of HiVE as QR codes (2D barcodes) enabling, for example, visualisations to be shared by photographing barcodes on posters (Fig. 6) and will show how *vizTweets* can be communicated between users and clients through micro-blogs such as *Twitter* (Fig. 7).

Figure 6: QR codes containing HiVE statements for Figs. 1–5.

Figure 7: HiVE statements stored and communicated using *Twitter*.

Acknowledgements

vizTweets is funded by a JISC Virtual Research Environment Rapid Innovation grant. The authors are grateful to Strachan and Vidale (University of Reading) for providing data and the Willis Research Network for funding Slingsby's position.

References

[1] giCentre. HiVE: Hierarchical Visualization Expression Language. http://gicentre.org/hive/, 2010.

[2] A. Slingsby, J. Dykes, and J. Wood. Configuring hierarchical layouts to address research questions. *IEEE Transactions on Visualization and Computer Graphics*, 15:977–984, 2009.

[3] L. Wilkinson. *The Grammar of Graphics (Statistics and Computing)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[4] Willis. The Willis Research Network. http://www.willisresearchnetwork.com/, 2010.