

Exploring Italian Wine: a Case Study of Aesthetics and Interaction in a Generative Information Visualization Method

Luca Buriano
Telecom Italia Lab
Via Reiss Romoli, 274
10148 Torino, Italy
luca.buriano@telecomitalia.it

ABSTRACT

Interactive applications can greatly benefit from Information Visualization (Infovis) methods addressing aesthetic and creative design aspects, to help in effectively conveying the meaning of complex data. We present a novel Infovis design and method, applied to the interactive visual exploration of Italian wines' properties. This work adopts a generative approach, based on automatic creation of the visual layout according to functional as well as aesthetic and perceptual criteria.

Categories and Subject Descriptors

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

General Terms

Algorithms, Design, Human Factors.

Keywords

Information visualization, graphic design, generative art, generative design, visual recommender systems, interactive data exploration, wine culture.

1. INTRODUCTION

Aesthetic and creative design aspects are essential to the development of a successful Information Visualization (Infovis) project, helping to immediately and effectively convey the meaning of complex data [10]. To this end, we think that inspiration can be found in the approach of generative visual art and generative graphic design, where the artist/designer, after envisioning a set of aesthetic, functional and semantic criteria, models them as a process and lets the resulting system organize into the actual, emergent visual patterns [3][5][6]. This kind of flexibility can greatly benefit interactive applications, where the visualized data are dynamic by nature. Following this framework, we present a demonstration of a novel Infovis visual concept and method, applied to an interactive application scenario. This concept emphasizes, in addition to its specific visual design: a) visual layout creation driven by a fitness function taking into account not only data relationships, but also aesthetic, perceptual

and graphic design aspects; b) immediacy and easiness in the interactive exploration of the resulting visualization.

2. RELATED WORK

The aesthetic features of an Infovis project depend primarily on the human designer's creative design insights and skills [10][9][1]. When dealing with dynamic and interactive applications, where the data to visualize are selected on the fly, the designer faces the additional challenge of managing the layout and aesthetics of unpredictable data configurations. In the Infovis field, this is usually addressed in essentially two ways: a) letting the aesthetic features emerge from layout algorithms designed to show data properties and relationships with clarity (see, for example, graph layout algorithms[4][11]); b) directly embedding aesthetic criteria into the visualization algorithm (see, for example, circle packing layout algorithms [2]). However, the combined use of a) and b) and the embedding of multiple, potentially complex aesthetic factors remain challenging and relatively unexplored areas. Our work aims to integrate the approaches a) and b) in a single generative framework. The generative process is applied to a visual concept with many degrees of freedom, related to the family of grid visualizations (see for example [8]). The process is driven by a fitness function, able to take into account multiple data-driven, aesthetic and graphic design visual factors. The flexibility and modularity of the fitness function allows the designer to experiment with different aesthetic criteria and styles. Moreover, the generative approach naturally leads to the creation of a diverse set of visual solutions for a given data set, enriching the user's experience with a source of visual novelty.

3. APPLICATION SCENARIO

Our work's example application scenario is a visual recommender system on the domain of Italian wines from the Piemonte region, based on a database of these wines and their properties related to smell, taste, grapes and production locations. This application scenario has been envisioned in the context of the "PIEMONTE Project"¹, a research project whose core concept is that "smart things" can play the role of gateways for enhancing the interaction between people and a territory with its cultural heritage.

When the user chooses a wine from the list, the system extracts from the database a set of wines with similar properties (according

Presented at EuroHCIR2012. Copyright © 2012 for the individual papers by the papers' authors. Copying permitted only for private and academic purposes. This volume is published and copyrighted by its editors.

¹ <http://www.piemonte.di.unito.it/index.html>

to a given similarity function) and creates an interactive visualization of this set, allowing visual exploration of wines' properties.

4. VISUALIZATION

Our visualization method works on a set of n items, where each item, identified by a name label, is linked to a list of properties taken from a domain-specific dictionary and categorized in property groups. In the wine recommendation scenario, items and item properties become wines and wine properties, categorized in the following property groups: smell, taste, primary grape, secondary grape and production location.

The algorithm is composed by the following steps:

1) An initial visualization layout is built as a $n \times m$ matrix, where each row corresponds to a distinct item and each column corresponds to a distinct property (m is the number of distinct properties appearing in the given item set). Each element in this matrix, corresponding to a item-property combination, is assigned a color, determined by the property group the element's property belongs to, according to a given color table. For the wine domain, we chose the following color table association: taste = red, smell = indigo, primary grape = green, secondary grape = cyan, production location = yellow. Elements corresponding to item-property combinations not appearing in the visualized item set (called in the following "empty elements") are assigned the background color. A special column in the matrix will contain item's name labels.

2) A large number of candidate visualization layouts is generated, where each candidate is obtained by randomly rearranging rows and columns in the matrix (keeping individual rows and columns intact, for data coherence). Candidate layouts are assigned a score according to a fitness function; the actual visualization layout is randomly selected among the candidates with the highest scores. The fitness function takes into account a set of factors related to the visualization's functional, aesthetic and graphic design factors, including:

- minimization of the distance between rows corresponding to similar items (the similarity between two items is calculated as a function of the properties they share);
- preference for visual grouping of columns linked to the same property group (i.e., of the same color);
- preference for specific shape properties of local or global visual emergent patterns (e.g., preference for spatially compact patterns, minimizing the number of gaps between colored matrix elements);
- preference for specific color pairings of matrix elements, according to a given color theory (e.g., preference for complementary colors of adjacent elements).

The fitness function is calculated as a weighted sum of components, where each component calculates the score assigned to a given layout with respect to a different design factor. This modular structure allows for seamless modelization and inclusion of other design factors when needed, for example when the designer wants to experiment with different visual and interaction styles.

During this step, groups of two or more columns can be optionally fused together into one column, provided that non-empty elements

from these columns won't overlap in the resulting column. These column fusion operations generate new candidate layouts.

3) Finally, the chosen visualization layout is displayed (Figure 1). Each matrix's element is drawn as a borderless tile painted with the color associated to the element. Item names are displayed in their column, with the tile belonging to the chosen item highlighted by a thin border. Optionally, the brightness and width of adjacent columns of the same color is slightly modified, in order to increase visual diversity and allow the user to tell these columns apart more easily.

5. INTERACTION

Interactive exploration of the resulting visualization can be performed in several ways, including:

- selecting a wine in the name column: all the wine's properties in the corresponding row will be highlighted, by showing their labels (e.g. in Figure 1, the 5th wine from the top is selected) ;
- selecting a property column: the property's label will be shown at the interaction point and all the wines sharing that property will be highlighted, emphasizing their name labels' font and/or coloring their tiles with the same color as the property tiles (e.g. in Figure 2, the "amarognolo" (slightly bitter) taste property is selected). It should be noted that, by default, properties' labels are intentionally not shown until they (or related wine names) are selected, in order to encourage visual data exploration and reduce information overload. At any moment the user can, of course, ask for permanent display of a given label, or all labels;
- asking for alternate layouts (from the best candidates found in the step 2) of the visualization algorithm);
- making a new recommendation/visualization query, choosing one of the visualized wines, or a wine from the global list;
- changing the visualization's settings, e.g. setting the number of similar wines shown as the result of a query, or setting the minimum number of wines a property must belong to, in order for the property to be shown;
- manually rearranging the layout of rows and columns with drag-and-drop actions.

When the user makes a new query, a smooth visual transition between visualization layouts is provided by animations "dissolving" the current layout (by moving tiles outwards in random directions and fading them out) and "assembling" the new one (by moving tiles from random outscreen positions to their actual position in the layout, and fading them in).

6. IMPLEMENTATION

The Infovis method described in this paper has been implemented as a prototype application for the wine recommendation scenario using Processing², an open source programming language and environment for creating images, animations, and interactions [7]. The availability of a powerful language for building visual structures, together with the compatibility with the Java programming language, made Processing an excellent rapid

² <http://processing.org>

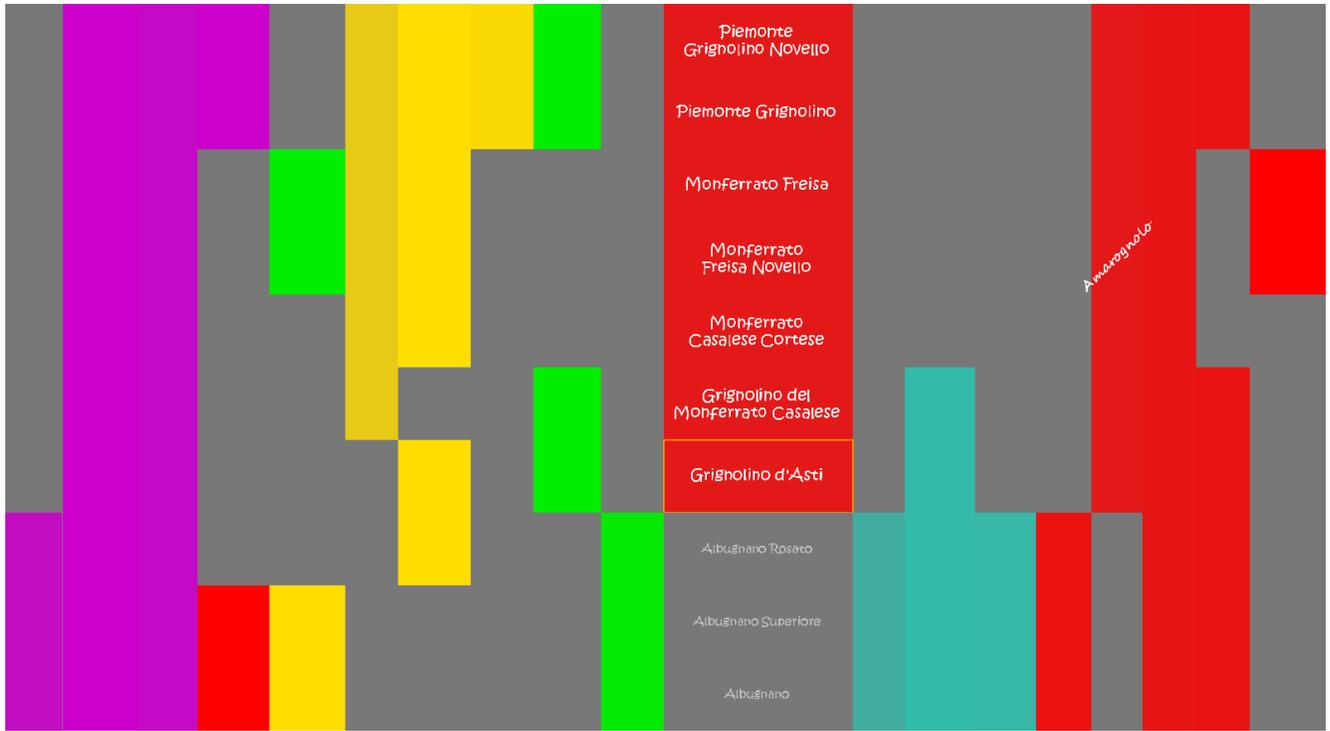


Figure 2.