Automating the Design of Graphical Presentations of Relational Information

By Jock Mackinlay, Tableau (then Stanford)
Overview

• Automate the design of 2D graphical presentations for relational data

• Why is this hard?
  – Must express design criteria so that the machine can understand them (*expressiveness criteria*).
  – Must express information in an understandable way given the capabilities of the output medium (*effectiveness criteria*).
Goal

Given application data, a presentation tool (APT) creates an image that effectively displays the data.
Credo

“...an important responsibility of a user interface is to make intelligent use of human visual abilities and output media whenever it presents information to the user.”
The Graphical Presentation Problem

• What is an effective encoding of information?
• Example:
  Given a car database:
    • Present the *Price* and *Mileage* relations.
    • The details about the set of *Cars* can be omitted.
• Many ways of encoding any information.
Approach: Expressiveness Criteria

• Graphical presentations -
  - “sentences of graphical languages”
  - essentially logical statements

• A set of facts is expressible in a language
  - if and only if it contains a sentence that describes exactly that set of facts
  - no more, no less.
Expressiveness Criteria
Expressiveness Criteria

A *graphical sentence* \( s \) describes objects and their locations.

- Object: Square, Circle, Triangle, Plus, Minus, Star, Smiley Face, etc.
- Location: \( X_{\text{min}}, X_{\text{pos}}, X_{\text{max}}, Y_{\text{min}}, Y_{\text{pos}}, Y_{\text{max}} \).

\[
s \subseteq \{ o, l \} : o \in O \land l \in L \}
\]
Expressiveness Criteria

• Example: You can describe a 1-D horizontal graph using a “horizontal position” language

• A graphical sentence can belong to this language if it describes either the horizontal axis or a “+” mark on that axis:

\[
\text{HorzPos}(s) \iff s = h \cup m \land \langle o, l \rangle \in m \Rightarrow \begin{cases} o = \text{plusobj} \land & Y \max(h) \leq Ypos(l) = \text{const} \land X \min(h) \leq Xpos(l) \leq X \max(h) \end{cases}
\]
Expressiveness Criteria

*Encodes* relation: Given a language for presenting information (like HorzPos), *Encodes* is the relationship between the facts you are encoding and the objects on the screen.
Expressiveness Encoding

Example: given a relation r with tuples \((a_i, b_i)\) (where \(a\) is an element of the set of marks and \(b\) is an element of the set of positions) for the HorzPos language, there are three \textit{Encodes} relations:

- Range of locations are encoded by the horizontal axis:
  \[
  \text{Encodes}(h, \{b_1\ldots b_n\}, \text{HorzPos})
  \]

- Each located object \(o_i\) in the set of marks \(m\) encodes an \(a_i\):
  \[
  \text{Encode}(o_i, a_i, \text{HorzPos})
  \]
Expressiveness Encoding

The relation $r$ can be encoded by the position of each mark along the horizontal axis in the horzPos language

$$\text{Encodes(position(m,h), r, HorzPos)}$$
Effectiveness Criteria

• How do we design an effective presentation automatically?

• Based on empirically verified knowledge, not mathematical rigor.

• A ranking of perceptual tasks is used to decide which graphical language to employ (Fig 14-15).
Expressiveness Criteria
Jock Mackinlay, 1986

[Quantitative]
- Position
- Length
- Angle
- Slope
- Area
- Volume
- Density
- Saturation
- Hue
- Texture
- Connection
- Containment
- Shape

[Ordinal]
- Position
- Density
- Saturation
- Hue
- Texture
- Connection
- Containment
- Length
- Angle
- Slope
- Area
- Volume
- Shape

[Nominal]
- Position
- Hue
- Texture
- Connection
- Containment
- Density
- Saturation
- Shape
- Length
- Angle
- Slope
- Area
- Volume

[Decreasing]

[Mackinlay, Automating the Design of Graphical Presentations of Relational Information, 1986]
Effectiveness Criteria

Principle of Importance Ordering:
Encode more important information more accurately (use information higher in the ranking to encode more important information)
Composition

• How do you design new presentation designs?
  – Can just have a laundry list of them
  – But it is better to take a bunch of simple “primitive” ones and combine them.

• Principle of Composition:
  – Compose two designs by merging parts that encode the same information.
Composition

Merge different encoding techniques not usually combined
Axis Composition

• Example: ozone measurements in two different cities.
  – Y-axis: ozone density
  – X-axis: date
  – first figure, from Yonkers, second from Stamford: overlay them.

• Only can do this if the axes encode the same information
Axis Composition

• Formally:

\[ v_i = v_j \neq \{ \} \land h_i = h_j \neq \{ \} \land \]
\[ \text{Encodes}(h_i, x, l_i) \land \text{Encodes}(h_j, x, l_j) \land \]
\[ \text{Encodes}(v_i, y, l_i) \land \text{Encodes}(v_j, y, l_j) \]

• Similar for single axis composition
Mark Composition

Merges mark sets if the sets encode the same information in the same way
- position: positions of objects along existing axes are same
- retinal: retinal properties must be the same
Implementation

• Uses logic programming to determine possible designs given the formalisms.

• Uses divide and conquer algorithm:
  – Partition
  – Selection
  – Composition
Implementation

• Partitioning
  • A divide and conquer algorithm
  • Partition on most important element

• Selection
  • For each partition, a list of graphic design is generated based on expressiveness criteria
  • Then, the list is ordered by the effectiveness criteria

• Composition
  • Each partition’s graphic design is tested to see if they both can be applied, if not the next most effective graphic design is used
Partition

- **Partition (divide)**
  - order the attributes by importance
  - divide them up into groups that match expressiveness criteria
  - `<Price, Mileage, Repair, Weight>` can be partitioned into `<Price>`, `<Mileage, Repair, Weight>`.
  - `<Mileage, Repair, Weight>` must be repartitioned recursively until something that can be encoded is obtained
Selection and Composition

• Selection
  – For each partition, filter out incompatible design criteria
    • e.g., cannot use maps to encode <Price, Mileage, Repair, Weight>

• Composition
  – Composes the individual designs into a unified presentation of all information
Summary

• Formalizes Bertin’s graphical presentation scheme
• Shows that machine generated presentations are feasible
• Develops a formal model for analyzing graphical representations of data
Discussion/Critique

• Strengths:
  – Was the first to develop a framework for automating graphical presentation creation
  – Defined criteria for evaluating presentation tools (effectiveness, expressiveness)

• Weaknesses:
  – Not clear that APT is particularly useful
  – Are APT generated presentations effective?
  – Can only do limited types of presentations
Car price for 1979
Car mileage for 1979
Repair record for 1977
Car weight for 1979