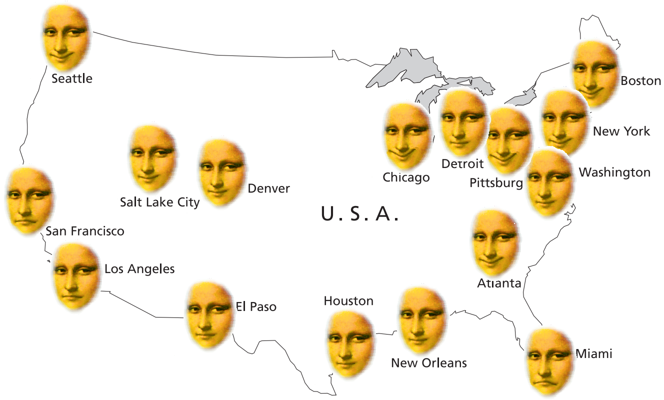
Chapter 6

Applications in Visualization



The visualization of scalar and multivariate quantitative data involves the mapping of data onto a visual scale. The principles of such a mapping of scalar data to scales of visual attributes are well-known for a number of basic scales. In color mapping, data values are mapped to appropriate hue of lightness values. Scatter plots are based on the principle of mapping data values to positions, or more exactly, to distances from an axis. Other variables often applied for such purposes are scale, form, and texture. Bertin [1983] describes a general methodology of how to select an appropriate mapping to these visual variables and how to combine them. Cleveland gives a ranking of their effectiveness [Cleveland 1985]. For multivariate data with two upto five dimensions more complex color and texture scales can provide solutions in some cases. For the visualization of local variations higher dimensions data glyphs have been proposed and been applied successfully. Chernoff faces [Chernoff 1973] and stick figures [Pickett & Grinstein 1988] are well-known examples for this visualization approach. Tufte gives a good overview of relevant visualization techniques and discusses their effectiveness in certain application areas [Tufte 1983]. Nevertheless, a generally accepted set of visualization rules does not exist.

All the techniques mentioned above represent fundamental approaches to the visualization of scalar and multivariate data. The general understanding is that no single of these visualizations is effective in all possible situations. One visualization may – and should of course – produce new insights and, by this, new questions which again produce the need of different views to the data. A good number of applications have proven that a user can gain knowledge about some unknown data more effectively when provided with highly interactive techniques [Rheingans 1992]. Techniques such as Focusing and Linking [Buja et al. 1991] extend this interactivity even further by connecting different views to the data using interactive feedback.

However, effective visualization still is very, very difficult. There are a number of reasons for this. First, a mapping of application data to these fundamental variables involves an abstraction. If the user is familiar with the idea of this mapping, he may understand the generated visualization good. However, often the application context is lost and the visualization which was applied to simplify the analysis of the data involves an analysis step or experience by its own.

Second, it is not easy to visualize a number of parameters using these fundamental mappings only. Usually, the visualization of multi-parameter data involves the generation of application specific models and solutions. General solutions and general scales do not exist for this purpose. Consequently, the user needs methods to define a visual scale for such applications very quickly and easily. Such methods do hardly exist to date.

Last, and may be even most important, it is still difficult to change visualization parameters and to produce a new, appropriate visualization intuitively. For example, to highlight a specific data value one has usually to modify the data filtering or to completely redefine the mapping of the data to visual attributes. A direct and interactive modification of local data mappings is hardly provided by any visualization technique today.

We have introduced a new approach for the visualization of scalar and multivariate data, which addresses the problems presented above [Alexa & Müller 1999b]. This approach is based on the direct and interactive specification of local data mappings.

6.1 Visualization by Examples

The general paradigm of our visualization approach is to enable the user to visualize some data by specifying the mapping of a small number of selected data values. We call this Visualization by Example.

This approach can be explained best with an example. A simple mapping of some scalar quantitative data to color can be defined by linking two arbitrary data values with appropriate color values. This results in a linear mapping. Further links can be supplied to adjust the mapping locally, resulting in a more sophisticated mapping function. The visualization of any local feature and its neighborhood is directly and intuitively controlled by the user and may be easily changed when provided with appropriate visual attributes or objects on which the data may be mapped. Note that this strategy is fundamentally different from the selection of a color scale and applying this scale to all data.

While the direct and interactive linking of data values with visual representations is not new, it has not been combined with appropriate methods for the approximation of data mappings based on the supplied correspondences.

In the example presented above this approach seems simple and easy to follow. However, the generalization of this approach calls for a mathematical model describing the data objects, flexible visual scales, and the mapping between this values based on a small number of parameters and features.

In addition, the user will have some additional knowledge about the data in many cases, which can be exploited with our approach. Moreover, the user might want to highlight several data values by mapping them to special representations.

There exist approaches to describe data spaces based on mathematical models [Brodlie 1993]. However, in the context of this work mathematical models are proposed to describe the spaces of visual scales and morphing is used to construct the corresponding graphical objects for this purpose [Alexa & Müller 1998b; Müller & Alexa 1998]. This approach allows to define rich sets of useful visual representations from only a few graphical base objects. As such, this method provides the appropriate foundation for a Visualization by Example.

In the following section we will discuss the fundamentals and characteristics in more detail.

6.2 Visual representations from morphing

For our visualization technique, the visual representations have to be structured as a multidimensional space. That is, a visual object has to be element of an n-dimensional space and represented by a vector $r \in \mathbb{R}^n$.

For many visual scales such a representation is quite natural:

- Color can be represented by real values in $[0,1]^3$, color scales might be represented by real numbers in $[0,1]$.
- Size or position is naturally a real number.
- For textures one defines a number of real valued parameters, which control their appearance.
- In general: If the visual representations are used to depict quantitative data there has to be a reasonable understanding in terms of real valued vector spaces.

We have proposed a more flexible way to define spaces of visual representations [Alexa & Müller 1998b; Müller & Alexa 1998]:

Figure 6.1: A smiling scale produced by morphing a mona lisa face.

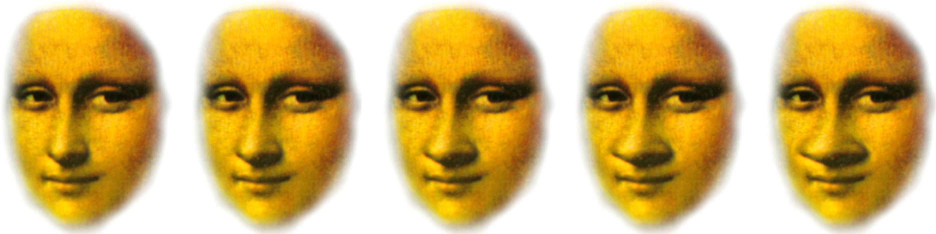


Figure 6.2: Another scale produced by morphing a mona lisa face.

Given a number of graphical objects of any class (images, polyhedra, etc.) we construct a space by morphing among these objects. Morphing between two objects produces one-dimensional visual scales. Two such scales are depicted in Figures 6.1 and 6.2. Here, the degree of smiling could be used to represent a scalar value. While this scale might not be as visually strong as e.g. the size of a dot it is easier to connect to the real-world phenomenon behind the data values. A smile is obviously representative for "good" values in the context of the data, while abstract representations need a legend, which has to be learned and remembered.

In this approach, a morph among multiple objects by performing several morphing operations between two objects subsequently defines a multidimensional space of objects (see previous chapter).

Since morphing is applicable to produce scales such as color, position, size, etc. we understand this to be a generalization of these techniques to define spaces of visual representations (e.g. glyphs or icons [Beddow 1990; Pickett & Grinstein 1988]). Note that our technique allows to represent glyphs or icons as an n-vector.

The strength of using morphing techniques to generate visual representations of data becomes evident when applied to multivariate data. As mentioned before, it has been proven difficult to find intuitive visual representations for multivariate data and multidimensional objects. By morphing among multiple objects one could visualize multivariate data as elements of a space of graphical objects.