6.3 Mapping Data to Coordinates

The main idea of our approach is to let the user define several relations between data values and graphical representations. These correspondences are used to construct a mapping from data to visual representations. We want to allow the user to define any number of correspondences, usually beginning with only a small number of correspondences. Depending on the application the user might decide to generate an affine mapping in any case, or, if no simple affine mapping can be found, to accept a non-linear function to depict the mapping.

To define a single correspondence, the user first chooses a data value and then searches the space of graphical representations for a suitable element. That is, the user gives an example for the intended relation between data and visualization.

We denote the space of data values as V^d, i.e. each vector consists of d variates. Assume a number e of data vectors $\mathbf{q_0}, \mathbf{q_1}, \ldots, \mathbf{q_{e-1}}$ should be mapped to a coordinate $\mathbf{r_0}, \mathbf{r_1}, \ldots, \mathbf{r_{e-1}}$ in the space of visual representations. If $e \leq d$ then an affine mapping $a : V^d \to \mathbb{R}^n$ exists that satisfies $a(\mathbf{q_i}) = \mathbf{r_i}$ for all $i \in 0, \ldots, e-1$. However, if $e > d$ that mapping does not necessarily exist.

We suggest to use a linear mapping whenever possible, i.e. in case $e \leq d$. The reason for this is that visual scales are still meaningful, properties of the data variates are preserved in the visualization, and the order of the data values is not changed.

If e is greater than d we offer three choices:

1. An affine mapping that fits the given correspondences as far as possible.
2. A non-linear mapping that satisfies all relations by locally changing a linear approximation.
3. A non-linear mapping that satisfies all relations by globally interpolating the relations.

The second and third mapping are constructed using the same approach. Both need an approximation of the affine mapping similar to the one used in the first approach. In the upcoming subsection we explain how to calculate that affine mapping, independent of the number of given relations.

6.3.1 Finding an affine mapping

We want the mapping a to be represented by a matrix multiplication. Thus, we are searching for a $n \times d$ matrix A that maps from V^d to coordinates in the space of visual representations. In case $e \leq d$, A has to satisfy the simultaneous equations

$$\begin{aligned} A\mathbf{q_0} &= \mathbf{r_0} \\ A\mathbf{q_1} &= \mathbf{r_1} \\ \vdots &= \vdots \\ A\mathbf{q_{e-1}} &= \mathbf{r_{e-1}}, \end{aligned} \tag{6.1}$$

in case $e > d$ we like to minimize the residual

$$(\|A\mathbf{q_0} - \mathbf{r_0}\|, \|A\mathbf{q_1} - \mathbf{r_1}\|, \ldots, \|A\mathbf{q_{e-1}} - \mathbf{r_{e-1}}\|) \ .$$

We now first solve the first case. The techniques we employ here will automatically produce a solution to the second case.

If we look at the i-th row $\mathbf{a_i}$ of A we get the simultaneous equations

$$\begin{aligned} \mathbf{a_i q_0} &= r_{0_i} \\ \mathbf{a_i q_1} &= r_{1_i} \\ \vdots &= \vdots \\ \mathbf{a_i q_{e-1}} &= r_{e-1_i}. \end{aligned} \tag{6.2}$$

We define the $d \times e$ matrix

$$B = \begin{pmatrix} - & \mathbf{q_0} & - \\ - & \mathbf{q_1} & - \\ & \vdots & \\ - & \mathbf{q_{e-1}} & - \end{pmatrix} \tag{6.3}$$

to rewrite the simultaneous equations in 6.2 as a matrix equation:

$$Ba_i^T = (r_{0_i}, r_{1_i}, \cdots, r_{e-1_{i,}})^T \tag{6.4}$$

The solutions of these n systems of linear equations yield the rows of A. In order to solve one of these systems we use the Singular Value Decomposition (SVD, [Golub & Van Loan 1989]). The SVD has several nice properties that are interesting for our problem [Press et al. 1992]

1. It gives a stable solution in the quadratic case, even in the presence of degeneracies in the matrix.

2. It solves the under-specified case in a reasonable way, i.e. out of the space of solutions it returns the one closest to the origin.

3. It solves the over-specified case by minimizing the quadratic error measure of the residual.

Using the SVD we can compute all rows of A and thus have found the affine mapping we were searching for.

6.3.2 Non-linear mappings

We want to find a mapping a that satisfies all equations $a(\mathbf{q_i}) = \mathbf{r_i}$. This could be seen as a scattered data interpolation problem where we try to find a smooth interpolation between the values $\mathbf{r_i}$ given at locations $\mathbf{q_i}$. Contrary to some other application domains of scattered data interpolation, we deal with different and high dimensions of the vectors and typically the number of relations is close to the dimension of the input data.

As explained before, it seems desirable to have an affine mapping from the data values to the space of visual representations. Therefore, we always start with a linear approximation of the mapping (as calculated in the previous section) and then fit the relations in the mapping by tiny adjustments. For these adjustments we use radial sums. The idea of combining an affine mapping with radial sums for scattered data interpolation is considered in e.g. [Arad & Reisfeld 1994] and [Ruprecht & Muller 1995] (for two-dimensional vectors, only).

Hence, we define a by

$$a(\mathbf{q}) = A\mathbf{r} + \sum_j \mathbf{w_j} f(|\mathbf{q} - \mathbf{q_j}|), \mathbf{q} \in V^d, \mathbf{r} \in \mathbb{R}^n \tag{6.5}$$

where $\mathbf{w_i} \in \mathbb{R}^n$ are vector weights for a radial function $f : \mathbb{R} \to \mathbb{R}$. We consider only two choices for f:

1. The Gaussian $f(x) = e^{-x^2/c^2}$, which is intended for locally fitting the map to the given relations.
2. The shifted log $f(x) = \log\sqrt{(x^2 + c^2)}$, which is a solution to the spline energy minimization problem and, as such, results in more global solutions.

We compute A beforehand as explained in the previous section. Thus, the only unknown in (5) is a pure radial sum, which is solved by constituting the known relations

$$\mathbf{r_i} - A\mathbf{r_i} = \sum_j \mathbf{w_j} f(|\mathbf{q_i} - \mathbf{q_j}|) \tag{6.6}$$

This can be written in matrix form by defining

$$F = \begin{pmatrix} f(0) & f(|\mathbf{q_0} - \mathbf{q_1}|) & \dots & f(|\mathbf{q_0} - \mathbf{q_{e-1}}|) \\ f(|\mathbf{q_1} - \mathbf{q_0}|) & f(0) & \dots & f(|\mathbf{q_1} - \mathbf{q_{e-1}}|) \\ \vdots & \vdots & \ddots & \vdots \\ f(|\mathbf{q_{e-1}} - \mathbf{q_0}|) & f(|\mathbf{q_{e-1}} - \mathbf{q_1}|) & \dots & f(0) \end{pmatrix}$$

and separating 6.6 according to the n dimensions of $\mathbf{r_i}$ and $\mathbf{w_i}$:

$$F \begin{pmatrix} w_{0_i} \\ w_{1_i} \\ \vdots \\ w_{e-1_i} \end{pmatrix} = \begin{pmatrix} r_{0_i} - \mathbf{a_i r_0} \\ r_{1_i} - \mathbf{a_i r_1} \\ \vdots \\ r_{e-1_i} - \mathbf{a_i r_{e-1}} \end{pmatrix}, i \in 0, \dots, n-1 \tag{6.7}$$

Again, we solve these n equations by calculating the SVD of F. This time we are sure that an exact solution exist, because the solvability for the above radial functions f can be proven [Dyn 1989].

6.4 Results

We will demonstrate the techniques at two examples. These examples show two principally different application scenarios:

- The first example shows the mapping from multivariate data onto low-dimensional visual representation. That is, the dimension of the data is much higher than the dimension of the representations.

- The second example shows a mapping from scalar data onto either basic or more complex, multi-parameter representations. Here, specific aspects of the scalar data set are mapped to a specific channel of the visual attribute enhancing the expressiveness of the visualization.

6.4.1 Visualizing city rankings

In this example we visualize an overall (scalar) ranking of cities in the USA. Suppose we want a visual aid for a decision which of the major cities would be nice to live in. In order to quantify the different amenities and drawbacks of these cities we use data from "The places rated almanac" [Boyer & Savageau 1985]. This data contains values for nine different categories. That means, we need to project nine-valued vectors onto scalar values.

To visualize the ranking of the cities we use a Chernoff-like approach. The faces are generated by morphing among a standard set of facial expressions (as explained in the following chapter). In this example we make use of only a smile and a grumble, defining a one-dimensional visual scale. Thus, the degree of smiling represents the living quality determined by a combination of the nine data attributes from [Boyer & Savageau 1985].

One way to find this mapping might be to inspect the nine different categories and try to find some weights for the values. This requires not only to define nine values, also the correlation to the outcome of this mapping does not take into account the user's knowledge about the cities.

A more intuitive approach is to allow the user to supply a ranking based on personal experience. Remark that a ranking of a subset of all cities is sufficient. In figure 6.6 only three examples were given to generate an affine mapping. Namely, Chicago was thought to be nice to live in and was mapped to a smiling face, whereas Miami was unacceptable and mapped to a grumble. Additionally, Washington appeared nice but expensive and, therefore, mapped to a neutral face.

6.4.2 CT scan data

In this example, we inspect CT scan data from the "Visible Human"-project. The data is given as 16-bit data values on a 512 by 512 grid. A standard linear mapping of the relevant CT data to gray values is depicted in figure 6.7. Note, that this image could be produced by picking the two boundary values to define an affine map.

Figure 6.6: Cities of the united states represented by mona lisa faces. The representation is generated from 9-valued ranking vectors. The mapping was defined by mapping Chicago to a smile, Washington to neutral face, and Miami to a grumble.

In figure 6.7 the soft tissue is display relatively bright. We can adjust this for a better distinction of bones and soft tissue by simply selecting one of the data values from the soft tissue and assigning a dark gray to it. This time an affine mapping is a bad choice, because the three correspondences cannot be satisfied. Instead, we fit the mapping globally to the data value - gray value pairs by using radial basis sums with the shifted logarithm as the radial function. The resulted is depicted in figure 6.8 and clearly shows the advantage in comparison with figure 6.7.

If we take a closer look at figure 6.8 we find a brighter substructure in the stomach. We would like to bring this region of data values to better attention in the visual representation. We do this by mapping a data value of this region to a red color. That is, instead of using gray values in the visualization we now use RGB color. Note, that it is not necessary to use specific two-dimensional color scales: We simply specify which data value maps to which RGB triple. The gray value representations of the three correspondences defined earlier are mapped onto corresponding RGB values. The resulting mapping is shown in figure 6.9. Note, how the empty structures are colored in the complementary color of red. This gives a nice distinction of empty spaces and tissues.

6.5 Conclusion

We present a new approach to the construction of visual scales for the visualization of scalar and multivariate data. Based on the specification of only a few correspondences between data values and visual representations, complex visualization mappings are produced, hereby introducing a Visualization by Examples.

This approach exploits the user's knowledge about the data in a more intuitive way. Moreover, the user is enabled to adapt the visualization interactively and easily. The technique of Visualization by Examples can be used in combination with any visual representation

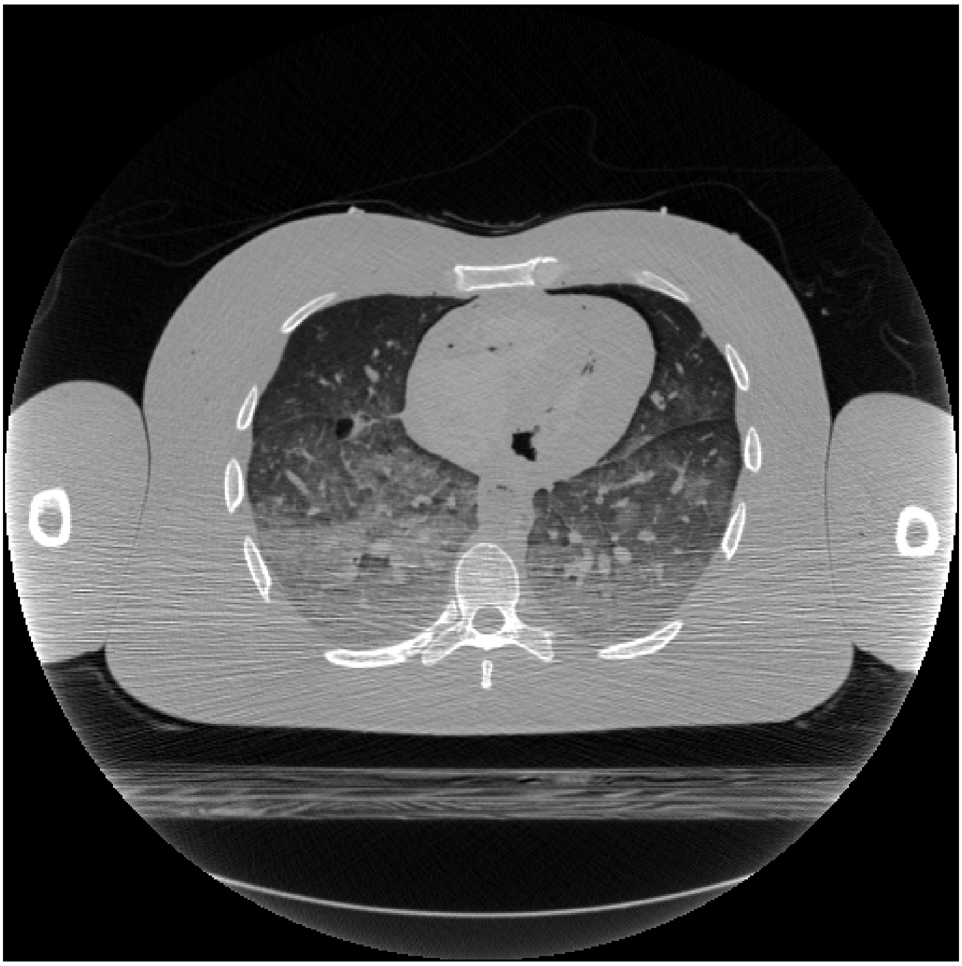

Figure 6.7: The CT-scan of the chest of a man. This image is generated from the raw CT-data by linearly mapping the range of useful CT-data values to a grey-scale

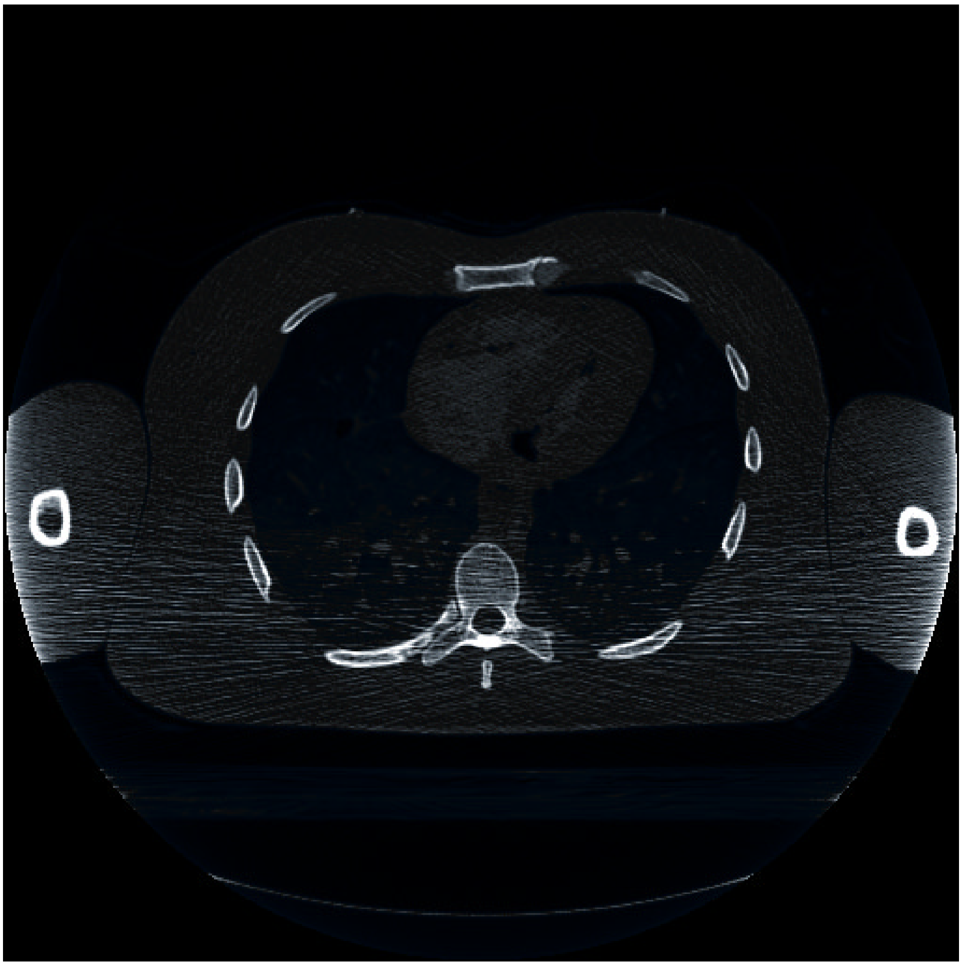

Figure 6.8: Here, the CT scan was generated by a mapping defined from three correspondences. The background was mapped to black, the bones were mapped to white, and the soft tissues surrounding the lung were mapped to dark grey.

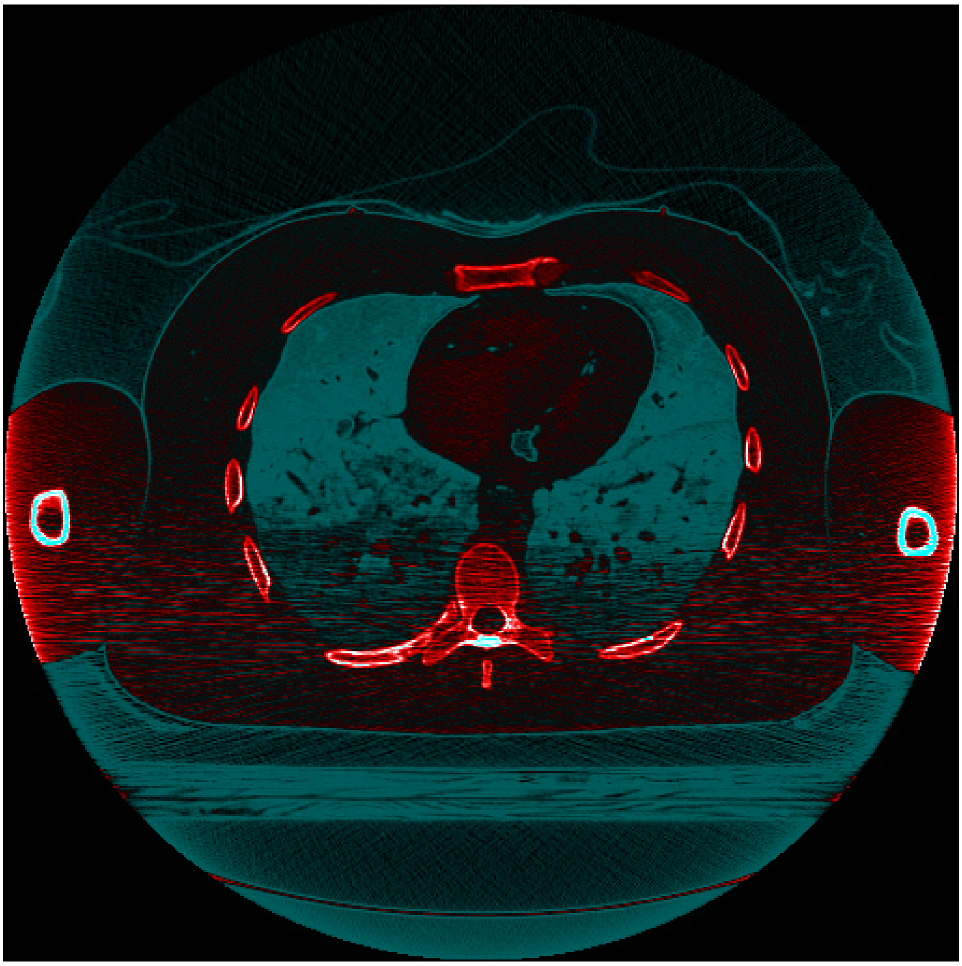

Figure 6.9: This image demonstrates the benefits of displaying scalar data with multidimensional visual representations. In addition to an already defined gray-scale, the soft structures of the bones were mapped to red color.