DURCHEINANDER
UNDERSTANDING CLUSTERING VIA INTERACTIVE SONIFICATION

Till Bovermann[1], *Julian Rohrhuber*[2], *Helge Ritter*[1]

Bielefeld University[1]
Neuroinformatics Group
Bielefeld, Germany
[tboverma|helge]@techfak.uni-bielefeld.de

Cologne University and
Academy of Media Arts[2]
SFB/FK 427
Cologne, Germany
julian.rohrhuber@uni-koeln.de

ABSTRACT

With *Durcheinander* we present a system to help understand Agglomerative Clustering processes as they appear in various datamining tasks. *Durcheinander* consists of a toy dataset represented by several small objects on a tabletop surface. A computer vision systems tracks their position and computes a cluster dendrogram which is sonified every time a substantial change in this dendrogram takes place. *Durcheinander* may be used to answer questions concerning the behavior of clustering algorithms under various conditions. We propose its usage as a didactical and explorative platform for single- and multi-user operation.

Figure 1: Video stills from the presentation of a prototype of *Durcheinander* at Animax, Bonn in late 2007.

1. INTRODUCTION

Durcheinander is an interactive system designed to support learners in understanding Agglomerative Clustering (AG) processes and researchers in investigating methods of their sonification. AG is a datamining approach that is mainly used to unveil structural relations in high-dimensional datasets[1]. It particularly facilitates the discovery of compact clusters of data items in high-dimensional vector spaces. Structures found by AG are assembled into a *dendrogram* that recursively interconnects single data items by means of their location (see Figure 2 for an example). Though the general behavior of AG with a given set of meta-parameters (which includes the used distance metric) can be easily understood, the parameters' relation to the algorithm's result in a *specific* case is more difficult to grasp. Participants in datamining courses can achieve greater understanding by exploring the answers to the following questions: *(a)* Under what variations in the data does the AG dendrogram change its configuration and how does it change? *(b)* What happens when data items are in a special configuration? *(c)* What are the differences between the various distance metrics? *(d)* What are the differences between the various cluster metrics?

*Durcheinander*s purpose is to help answer these questions by means of tangible interaction [2] and auditory displays [3, 4]. It provides the opportunity to physically *grasp* the data and, at the same time, allows auditory exploration of the effect of different clustering parameters. *Durcheinander*s tangible objects are laid out on a table and so the sound is delivered in a spatial sound environment. Learners have turned out to particularly benefit from this collaborative multiuser nature of the system—it invites to discuss the results of AG *in the process* of cooperative exploration, instead of in hindsight. Futhermore, its *interactive programming* approach allows researchers to experiment with different sonification methods during interaction.

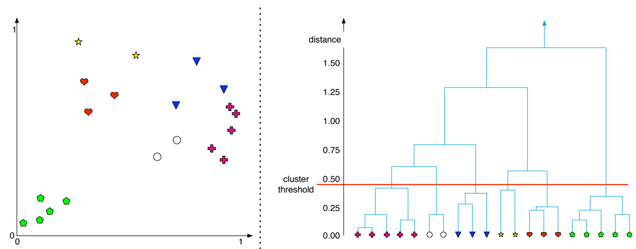

Figure 2: A 2D-plot of a toy dataset with its corresponding dendrogram. The red line indicates a specific clustering that defines the colors of the data items.

2. AGGLOMERATIVE CLUSTERING

Clustering can help unveil hidden structures of a specific kind in possibly high-dimensional data sets. It is especially suitable for compact structures in the sense of the used distance metric. Agglomerative Clustering (AG) is a special approach for clustering and produces so-called *dendrograms* of inter-cluster distances by the following rule-set:

1. Initially, all data items x_i are considered to be clusters c_i, so that $\forall x_i \in X : x_i = c_i$

2. Compute distances between all pairs of clusters and find the smallest distance:

$$\begin{aligned} \text{minpair} &= \text{argmin}_{i,j} \;\; d(c_i, c_j) \\ \text{mindist} &= \min_{i,j} \;\; d(c_i, c_j) \end{aligned}$$

3. Join c_i and c_j at the distance mindist. This joint $\langle c_i, c_j \rangle$ represents the new cluster c_k
4. Add c_k to the list of clusters, remove c_i, c_j from this list
5. If more then one cluster is in the list of clusters GOTO 2, else END.

A cut at a specific distance in the resulting dendrogram represents one possible clustering of the given dataset. For example applying AG to the dataset shown in Figure 2a results in the dendrogram shown in Figure 2b. The red line represents a possible cut.

Though it seems natural to use the standard euclidean metric to measure object distances, it is also possible to use other metrics which may fit better to the domain of the given dataset. The choice of the inter-object metric as well as the choice of how to determine cluster distances[1] heavily affects the structure of the AG's outcome and therefore the resulting dendrogram. Although it is relatively easy to understand the general global behavior of the clustering algorithm, it is difficult to understand the way in which local variations such as the exact position of data items affect the algorithm's output. This is particularly interesting since AG is usually applied to data with measuring errors that cause variations in data item locations.

A dynamically changing structure may not necessarily be best represented in form of a visual dendrogram; sonification allows us to explore its recursive (re-)configuration without a projection onto the plane of geometry.

3. IMPLEMENTING DURCHEINANDER

As a basis for *Durcheinander* we use the tDesk [5], a tabletop tangible computing environment designed and built in the interaction laboratory at Bielefeld University. By design, the dimensions of the surface allow groups of people to work on tangible applications, providing each member direct access to the physical objects. We use a digital camera below the tDesk to capture the 2D positions of the objects used as the data set in our system. This method prevents possible visual object occlusions by the users such that all 20 objects are all the time recognizable by the vision-engine. A blob recognition algorithm then detects number and position of the objects, which is fed into the actual clustering algorithm which computes the dendrogram.

The dendrogram structure is translated into a corresponding sound synthesis graph which may be triggered externally by knocking on the surface of the tDesk. The resulting sound is played in real time to the users by the multi-channel audio system surrounding the table. Here, each physical data item produces a sound that is spatially related to its position on the surface—every object sound again consists of sub-sounds determined by other nodes of the dendrogram.[2] The graph structure is being continually updated, and whenever its configuration differs substantially from its predecessor, the system generates a trigger that propagates through the synthesis graph; a series of reconfigurations may be heard as a series of differing sounds in context. The sonification algorithm constructs a computation graph where each node (representing a cluster c_i) takes an n-tuple of streams as input, provided by its enclosing cluster. In addition to this, a variable number of arguments allow parametric control and triggering of each node (see Figure 3). Each node passes on an n-tuple of streams to both of its two adjacent nodes. The algorithm that defines this flow-graph can be rewritten conveniently at runtime, so that different synthesis techniques may be experimented with while listening.

```
{|in, trig, freq, dist, id, lagTime|
    freq = freq.lag(lagTime.max(0.05));
    freq = freq * (3 ** dist);
    [
        in + Decay2.ar(trig, 0.01, 2.5, 0.1) * SinOsc.ar(
            freq,
            SinOsc.kr(Rand(1, 4), 0, 0.05, Rand(0, pi))
        ),
        trig,
        freq
    ];
};
```

Figure 3: Definition of a node's sound subgraph.

4. FIRST INSIGHTS

In late 2007 we presented *Durcheinander* at a workshop[3] for children at the Animax in Bonn. There we had the chance to extensively work with visitors to adjust *Durcheinander*s auditory display in real time (Figure 1) At ICAD 2008 we present a live demonstration of *Durcheinander*, and explain its particular usefulness in more detail. A video showing *Durcheinander* in action can be viewed at *http://tuio.LFSaw.de/durcheinander.shtml*.

5. REFERENCES

[1] B. S. Everitt, S. Landau, and M. Leese, *Cluster analysis*, Arnold, London, 4. edition, 2001.

[2] Hiroshi Ishii and Brygg Ullmer, “Tangible bits: towards seamless interfaces between people, bits and atoms,” in *CHI ’97: Proceedings of the SIGCHI conference on Human factors in computing systems*, New York, NY, USA, 1997, pp. 234–241, ACM.

[3] G. Kramer, Ed., *Auditory Display*, Addison-Wesley, 1994.

[4] T. Hermann and A. Hunt, “Interactive sonification,” *IEEE MultiMedia*, vol. 12, no. 2, pp. 20–24, 04 2005.

[5] T. Hermann, T. Bovermann, E Riedenklau, and H. Ritter, “Tangible computing for interactive sonification of multivariate data,” in *Proceedings of the 2nd Interactive Sonification Workshop*, February 2007.

[6] J. McCartney, “Supercollider hub,” URL, July 2004, http://supercollider.sourceforge.net/.

[1] These metrics differentiate AG into e.g. single-linkage, complete-linkage, or average based clustering.

[2] In order to realize such a framework, we implemented a modular sound architecture in SuperCollider, a higher-level programming language that is specially suited for real time sound rendering [6].

[3] This workshop was held by the DFG-funded research project *Artistic Interactivity in Hybrid Networks* as part of the German *Jahr der Geisteswissenschaften 2007.*