# nepDroid: An Intelligent Mobile Music Player

Sebastian Huber
Department of Computational Perception
Johannes Kepler University
Linz, Austria
se.huber@gmx.at

Markus Schedl
Department of Computational Perception
Johannes Kepler University
Linz, Austria
markus.schedl@jku.at

Peter Knees
Department of Computational Perception
Johannes Kepler University
Linz, Austria
peter.knees@jku.at

## ABSTRACT

Mobile music consumption has been spiraling during the past couple of years. The interaction techniques provided to sift through the ever increasing amounts of music available on smart devices unfortunately have not. In this paper, we address this issue and present an intelligent mobile user interface that enables the user to browse her mobile music collection in a joyful and informed way.

## Categories and Subject Descriptors

H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems

## General Terms

Algorithms, Human Factors, Performance

## Keywords

Content-based Music Similarity, User Interface, Clustering, Visualization, Music Information Retrieval

## 1. MOTIVATION

The ever increasing amount of music available on mobile devices, such as smartphones, demands for intelligent ways to browse music collections, either available as digital files on the device or as music stream. Although current retrieval methods based on meta-data search perform well when the user has a specific information need in mind and can express it as a text query, they are incapable of organizing a collection according to acoustic similarities and in turn enable its joyful exploration by applying clustering and visualization techniques. We therefore present a user interface for the `Android` platform that allows to explore music collections in a game-like manner. Please also note that we have an `iOS` version available. However, the `Android` version is more mature, which is the reason why we focus on it here.



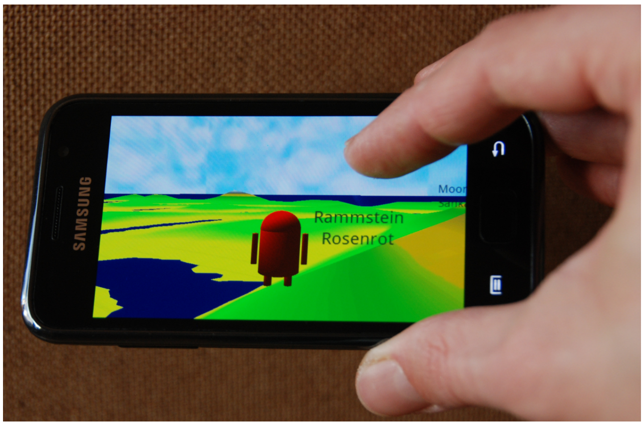


**Figure 1: Screenshot of the *nepDroid* user interface.**

## 2. THE NEPDROID PLAYER

Based on the idea of our `nepTune` player [2, 3], we present a similar user interface for the `Android` platform, taking into account the specific properties and demands of a mobile device. Figure 1 shows the player in action. In the front, we see a cluster, represented as an island, grouping together hardrock music. Such island visualizations are created by first extracting *audio features* from the digital music, subsequently applying a *clustering* approach, and eventually *visualizing the distribution of the music items* in the three-dimensional output space, where the height of the terrain corresponds to the density of the items.

### 2.1 Technological Background

Given a set of music files, `nepDroid` first extracts content-based audio features, more precisely *Mel Frequency Cepstral Coefficients* (MFCC) [1] that describe timbre and *Fluctuation Patterns* (FP) [5] that reflect rhythmic properties. The FP feature describes the strength of beats recurring at certain periodicities over a set of perceptual frequency bands. The output of both feature extractors is linearly combined and a *Self-Organizing Map* (SOM) [4] is subsequently employed to cluster the music pieces. We visualize the resulting clusters via a *Smoothed Data Histogram* (SDH) [6], by means of which the density of the data items over the map is estimated. The island-like appearance is eventually obtained by using the density estimates as height information and mapping the height values to colors.

### 2.2 Implementation Aspects

Due to performance limitations on mobile devices, feature extraction and pre-computation of the SOM and SDH are performed on the PC. To this end, a `Java` tool, run on the PC, creates a compressed XML file containing a 3-D model of the island landscape and meta-data for the songs. This file is subsequently transferred to the `Android` device. For visualizing the computed landscape, the `jPCT-AE` framework[1] is used. When the user starts the `nepDroid` application, she is presented the landscape generated from her music repository. An `Android` figure is used to visualize the current position of the user. Navigation is implemented through touch inputs. While moving around on the landscape, labels of the nearest songs are presented to the user. When the user navigates close to a song, `nepDroid` automatically starts playing the song. This interaction approach fosters the seamless exploration of the music landscape.

### 2.3 Performance Aspects

While implementing `nepDroid`, we encountered several performance issues concerning mobile devices. First, performing feature extraction and similarity computation is not feasible on current smart phones. It would take too much time and battery power. As already mentioned, we bypassed this by relying on the user's PC to carry out these tasks. But even on a state-of-the-art PC, these two tasks take around four minutes for just 500 songs. Figure 2 shows the runtime behavior of the major parts of the landscape generation process. Considering that today's average music repositories easily consist of several thousand songs, runtime is still a crucial factor.

Another performance issue is that the possible size of the landscape is strongly limited by the memory available for the application. On modern devices it is possible to show a few hundreds of songs, until the size of the landscape's 3-D model exceeds the available memory. On some devices also the rendering power of the GPU is a limiting factor. A solution would be to load the landscape not as a whole, but splitting it up into smaller pieces, which can be loaded on demand.

Surprisingly, the playback of the songs was also a performance issue. On `Android` a song has to be prepared before it can be played. The preparation step needs up to 500 ms for one song, which is too much to perform it during game play. We hence prepare the five nearest songs asynchronously and start playing them on demand. When the user is navigating fast over the island, no songs are prepared, because it would cause stutter, besides not being necessary in this case.

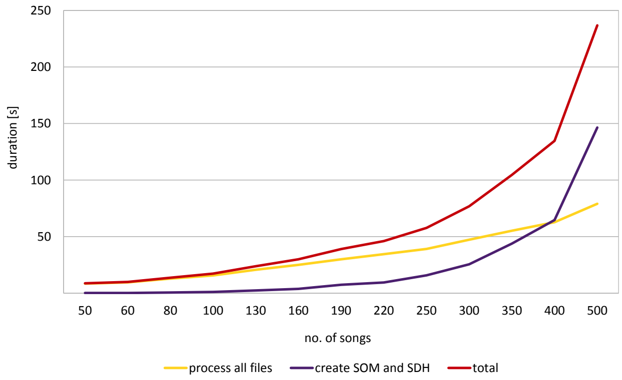


**Figure 2: Runtime behavior of the landscape generation process.**

## 3. CONCLUSIONS AND FUTURE WORK

In this paper, we presented `nepDroid`, an intelligent user interface to explore music collections on mobile devices. `nepDroid` offers a joyful way to discover music according to acoustic similarity rather than meta-data based filtering. We employ a clustering technique on the audio features (SOM), a density estimation approach on the resulting clustering (SDH), and we eventually visualize the music collection using a topographic color map. Exploring the landscape is possible via standard touch control.

As part of future work, we will explore methods to visualize *huge* amounts of music on mobile devices, as the current implementation is restricted to several hundreds of pieces. We may rely on some ideas we presented in [7]. The recent trend towards music streaming services demands for even higher computing and visualization capabilities, when building intelligent music browsing interfaces. We will further extend the interface towards a social music player, where listeners are connected and may mutually create, share, discover, and modify their islands.

[1]`http://www.jpct.net/jpct-ae` (access: Dec 2011)


## 4. ACKNOWLEDGMENTS

This research is supported by the Austrian Science Funds (FWF): P22856-N23.


## 5. REFERENCES


[1] J.-J. Aucouturier and F. Pachet. Improving Timbre Similarity: How High is the Sky? *Journal of Negative Results in Speech and Audio Sciences*, 1(1), 2004.

[2] P. Knees, M. Schedl, T. Pohle, and G. Widmer. An Innovative Three-Dimensional User Interface for Exploring Music Collections Enriched with Meta-Information from the Web. In *Proc. ACM Multimedia*, Santa Barbara, CA, USA, Oct 2006.

[3] P. Knees, M. Schedl, T. Pohle, and G. Widmer. Exploring Music Collections in Virtual Landscapes. *IEEE MultiMedia*, 14(3):46–54, Jul–Sep 2007.

[4] T. Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Germany, 3rd edition, 2001.

[5] E. Pampalk, S. Dixon, and G. Widmer. On the Evaluation of Perceptual Similarity Measures for Music. In *Proc. DAFx*, London, UK, Sep 2003.

[6] E. Pampalk, S. Dixon, and G. Widmer. Exploring Music Collections by Browsing Different Views. *Computer Music Journal*, 28(3), 2004.

[7] M. Schedl, C. Höglinger, and P. Knees. Large-Scale Music Exploration in Hierarchically Organized Landscapes Using Prototypicality Information. In *Proc. ACM ICMR*, Trento, Italy, Apr 2011.