A MUSIC RETRIEVAL SYSTEM BASED ON USER-DRIVEN SIMILARITY AND ITS EVALUATION

Fabio Vignoli **Steffen Pauws**

Philips Research Laboratories
Prof. Holstlaan 4
5656 AA Eindhoven (The Netherlands)
Fabio.Vignoli@philips.com, Steffen.Pauws@philips.com

ABSTRACT

Large music collections require new ways to let users interact with their music. The concept of finding 'similar' songs, albums, or artists provides handles to users for easy navigation and instant retrieval. This paper presents the realization and user evaluation of a music retrieval music that sorts songs on the basis of similarity to a given seed song. Similarity is based on a user-weighted combination of timbre, genre, tempo, year, and mood. A conclusive user evaluation assessed the usability of the system in comparison to two control systems in which the user control of defining the similarity measure was diminished.

Keywords: music similarity, user evaluation.

1 INTRODUCTION

End-users identify the concept of finding 'similar' songs, albums, or artists as one of the most appreciated features for future music players to get access to large music collections. They recognise the direct use of it in their daily music listening practice [1].

Though similarity in music is intuitively meaningful, it needs further definition. From a user perspective, judging similarity of songs either involves the comparison of two songs or the comparison of a set of alternative songs to a referent or ideal (e.g., a seed song). A straightforward method is to list all features of the songs involved and find the overlap in features. The reality is more complicated; similarity judgement seems to come down to the computation of a 'psychological function' of shared, distinctive, and comparable features of the songs involved.

Evidently, similarity needs to be explained with respect to a feature or a set of features. Simply stating that two songs are similar is not sufficient; we need to say that two songs are similar because of their instrumentation, their compositional style, their performers. Numerous research efforts have already been devoted to timbre similarity in music [2-5], in which timbre refers to the spectral information that correlates with instrumentation and articulation in musical performance. Unquestionably, timbre similarity is grounded by perception; non-musicians rather choose instrumentation over correct melody and harmony in similarity judgement of music by mere listening [6]. But still, timbre is only one facet of music similarity.

Similarity judgement is also afflicted with cognitive processes and reasoning using knowledge and conventions from the real world. It may even result into the observation that two songs are actually incomparable because, for instance, they originate from different cultural traditions, music idioms, or just because of personal conviction. Music psychology has already pointed out that besides instrumentation, at least tempo and genre information are indispensable for similarity judgments of music [7].

However, Only a few papers from engineering have addressed the problem of integrating the multiple facets of music similarity into a single objective function [8;9,18]. An interesting approach is presented in [9] where a music retrieval system combining similarity on timbre, lyrics as well as genre was tested with users. Results showed that users chose different combinations of these aspects to convey their music preferences.

Besides the involvement of various features, the contribution of each individual feature to the overall similarity needs to be weighted. Some features are more important than others to the end-user, the application context, or the set of songs under consideration. Given that the importance of features is heavily dependent on the context and the listening intention at hand, the user should be empowered to have total control on this weighting procedure. A similar approach is presented in [19] where the user can interactively combine measures of periodicity with measure of timbre similarity, although no evaluation was performed.

This paper presents the realization and user evaluation of a music retrieval music that sorts songs on the basis of similarity to a given seed song. Similarity is based on a weighted combination of timbre, genre, tempo, year, and mood. The end-user can specify her personal definition of similarity by weighting these aspects on a graphical user interface. A conclusive user evaluation assessed the usability of the system in comparison to two control systems in which the user control on defining the similarity function was diminished.

2 SONG SIMILARITY

The proposed system adopts a broad definition of music similarity by taking into account various features that contribute to the overall similarity of two songs: timbre, tempo, genre, mood, and year (respectively, denoted by s_s, s_t, s_g, s_m, s_y). In many cases, each feature adds new information to the similarity function. Italian Pop and Spanish Rock are for many people quite different genres, though their music can be classified as sounding similar in timbre, because of the instrumentation used by both. But in principle, the features do not need to be independent.

In this section, we describe the proposed algorithms to compute our features for music similarity.

2.1 Timbre similarity

Current state-of-the-art methods to compute timbre similarity between songs are usually based on a combination of signal processing and statistical tools, which boils down to the following schema: (i) computation of timbre-related features for each song, (ii) computation of timbre model for each song, and (iii) comparison of timbre models. A more detailed analysis of this topic and an exhaustive list of references can be found in [2].

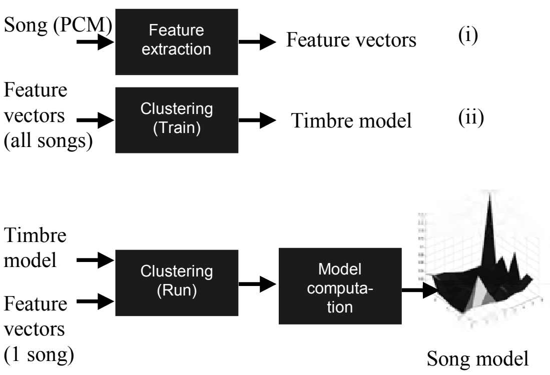


Figure 1: Block schema of computation of similarity (i) computation of feature vectors for each song, (ii) computation of a song model.

Our approach to compute timbre similarity is based on the same principles. The main difference lies in the computation of the song model, which, instead of being independent for each song, is computed relatively to the entire music collection. The main advantage with respect to the state-of-the-art methods is in the speed factor during the comparison step. The comparison method that achieves the best performance to date uses the Monte-Carlo approach (e.g. as described in [2]), which means comparing the timbre feature distribution of two songs by computing the likelihood of N points of the first distribution against the other, where N is usually in the range of a few thousands (4000 in [5]). The method we propose uses only a few multiplications for each song. However simple, this song model contains more information about the timbre of the song than a single averaged feature vector and it is able to provide sensible comparisons for songs with a complex structure (e.g., songs with an *intro* that is totally different from the rest of the song).

In Figure 1, a block schema of the process is shown. The first step is the computation of features. The music is converted to PCM format and a frame-by-frame analysis is performed. For each frame a feature vector v_c is computed as reported in [10].

The second step is split in two parts: first a model of the entire collection is made, and then the relative model for each song is computed. A Self Organizing Map (SOM) [11] is used to cluster the set of feature vectors for all songs in the collection in a unsupervised manner. Although other clustering algorithms can be used, we have chosen the SOM for its topology preservation property (i.e., features close in the N-dimensional Euclidean space are still close in the resulting *2*-dimensional space). For our purpose, a map size of *16x16* (i.e., *256* clusters) gave the best performance.

The set of centroids that results from the clustering can be called a *timbre-space* model. Each element of this model is a vector with the same dimensions as the v_c feature vector. The *timbre-space* model is a representation of the timbre of the entire music collection. The song model, which is also a *16x16* matrix, is computed as follows. Each element is computed by accumulating the response of the SOM (the closest cluster) for each feature vector of the song. The resulting matrix is normalized to represent a probability distribution.

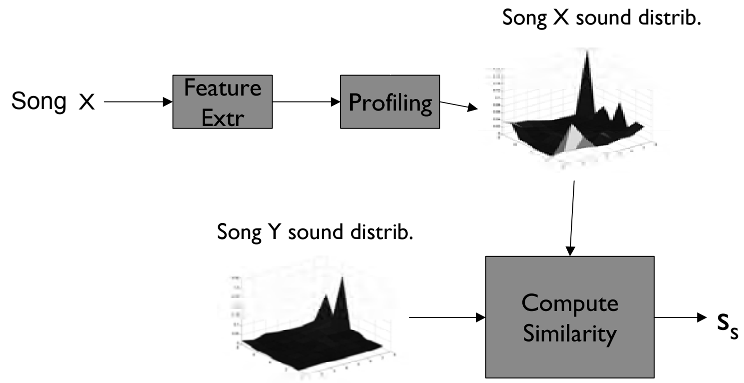
The last step, the comparison of song models, is performed by computing the Kullback-Leibler (KL) divergence L, which for two distributions $p(x)$ and $\tilde{p}(x)$ can be written as:

$$L = -\int p(x) \ln \frac{\tilde{p}(x)}{p(x)} dx.$$

For discrete distributions, the integration becomes a summation over the bins of the two distributions. The KL divergence is regarded as a measure of the extent to which two probability density functions agree. The KL divergence is not symmetrical, but in the context of computing song similar to a seed song, this property is desired. Reasons to believe in non-symmetric measures can be found in [12], where it is argued that humans tend to select the prototype (i.e., the seed song) as referent and the variants (i.e., the similar songs) as the subject of the similarity judgements.

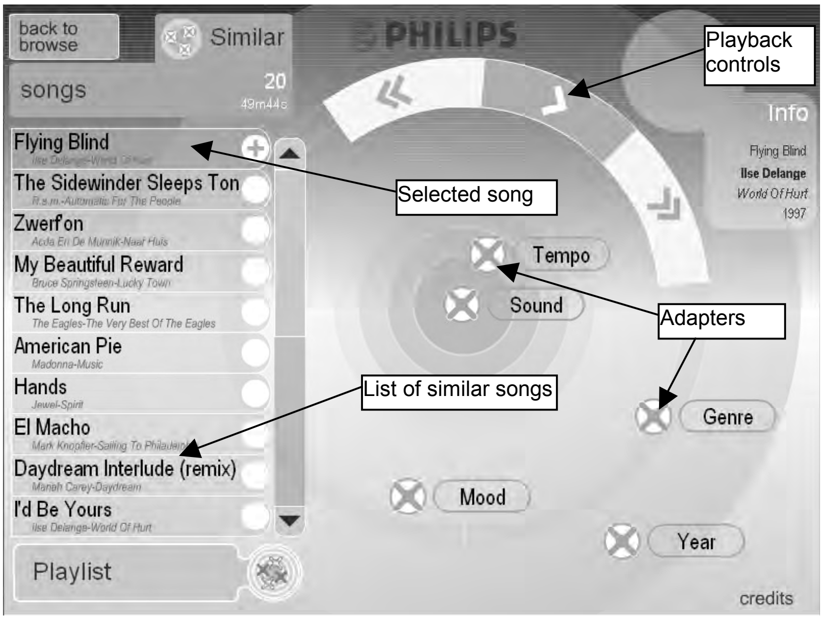
2.2 Genre, mood, year and tempo (dis-)similarity

Genre, mood, year, and tempo dissimilarities are computed by using the distance defined by Gowda and Diday in [13], which was initially developed for the task of clustering symbolic object. For our purposes, we convert these dissimilarities into similarities by taking their complement. According to the definition, which applies to both quantitative and qualitative features types, the dissimilarity between two objects A and B can be written as



- $S = w_S s_S + w_t s_t + w_g s_g + w_m s_m + w_y s_y$
- s=timbre, t=tempo, g=genre, m=mood, y=year

Figure 2: Block schema of the method used to compute “timbre similarity” and to combine the various similarities (s_s, s_t, s_g, s_m, s_y) into a single measure S.



Flying Blind
Ilse Delange-World Of Hurt

The Sidewinder Sleeps Ton
R.e.m.-Automatic For The People

Zwerf'on
Acda En De Munnik-Naaf Huis

My Beautiful Reward
Bruce Springsteen-Lucky Town

The Long Run
The Eagles-The Very Best Of The Eagles

American Pie
Madonna-Music

Hands
Jewel-Spirit

El Macho
Mark Knopfler-Sailing To Philadelphia

Daydream Interlude (remix)
Mariah Carey-Daydream

I'd Be Yours
Ilse Delange-World Of Hurt

Flying Blind
Ilse Delange
World Of Hurt
1997

Figure 3: The E-Mu jukebox in "Similar Songs" mode. The user can define the similarity function applied to the music collection by dragging the *sound/tempo/mood/genre/year* adapters on the screen. An adapter that is close to the center is weighted more than when it is positioned on the periphery.

$$D(A,B)=\sum_{k=1}^{d} D(A_k,B_k),$$

where A_k and B_k are the components of the k-th feature for the objects A and B, respectively.

For quantitative interval type of features (e.g., 1997-2000),

- au denotes the upper limit and al denotes the lower limit of interval A_k,
- bu denotes the upper limit and bl denotes the lower limit of interval B_k,
- n denotes the length of the intersection of A_k and B_k.

For qualitative type of features (e.g, genre and mood labels),

- l_a denotes the length (number of elements) of A_k,
- l_b denotes the length of B_k,
- n denotes the number of elements common to A_k and B_k,
- $l_s= l_a + l_b - n$ denotes the span length of A_k and B_k combined.

The distance between the two object with respect to the k-th feature is computed as $D(A_k,B_k)=D_p(A_k,B_k)+D_s(A_k,B_k)+D_c(A_k,B_k)$, where $D_p(A_k,B_k)=\frac{|al-bl|}{|U_k|}$ and U_k is the length of the maximum interval for the k-th feature. This dissimilarity component is due to the *position*, which arises only when the features are quantitative; it indicates the relative position of two feature values on the real line.

The expression $D_s(A_k, B_k) = \frac{l_a - l_b}{l_s}$ denotes the dissimilarity component due to the span. It indicates the relative size of the feature values without referring to common parts between them.

The expression $D_c(A_k, B_k) = \frac{l_a + l_b - 2n}{l_s}$ denotes the dissimilarity component due to the content. It is a measure of the non-common parts between two feature values.

When considering ratio/absolute type of features (such as year and tempo) the following applies: *al=au*; *bl=bu*, *la=lb=n=0*. Thus, the dissimilarity between two tempi ends up being proportional to the interval of tempi encountered in the collection under analysis. The tempo was manually tapped for each song used in this experiment.

Genre and mood are, for the current experiments, manually annotated with a single label (although automatic genre and mood classification could be used [14]). In this case, the definition of dissimilarity ends up being based on identity: if two songs are labelled with the same genre (mood) their similarity is equal to 1, otherwise it is set to 0. This definition of dissimilarity could also be used to compare features made of multiple labels as in the case of the style information of 'All Music Guide' [15].

2.3 Combining similarities

The overall similarity is given by the weighted sum of the different similarity components. Each similarity component is a real value between 0 and 1 and is weighted by a weight also in the range between 0 and 1.

1. Sound: the timbre of the song, it is based on content analysis as discussed in Section 4.1. Its weight is denoted by $w_s \in [0,1]$
2. Mood: the mood of the song (based on MoodLogic [16] data). Its weight is denoted by $w_m \in [0,1]$
3. Genre: the genre of the song (depends on the genre of the artist). Its weight is denoted by $w_g \in [0,1]$
4. Year: the year in which the song was released. Its weight is denoted by $w_y \in [0,1]$
5. Tempo: the tempo of the song (fast-slow). Its weight is denoted by $w_t \in [0,1]$

Finding a sensible weighting of the various components into a generic similarity function for all contexts, listening intentions, and songs under consideration is hard to do. Therefore, we let the user interactively decide what is the best weighting for her current purpose. Moreover, weighting the similarity components provides an interesting way to explore and navigate through a music collection.

2.4 ThE EXPRESSIVE MUSIC JUKEBOX

The Expressive Music (E-Mu) Jukebox has been created as an experimental platform to test algorithms and interaction concepts. The E-Mu Jukebox enables the user to browse a music collection by selecting genre/artists and albums and to search for songs based on similarity. When a user asks for songs similar to a seed song, the jukebox displays the screen shown in Figure 3.

The similarity components are represented by adapters. These adapters can be dragged on the bull's eye (as showed on the right-hand side of the screen). The radial distance of an adapter to the centre determines the weight of its corresponding similarity component in the similarity function. In this way, a user has the possibility to change the similarity function that is applied to the music collection. For instance, when the *genre* adapter is close to the centre, the *genre* component will be highly weighted in the similarity computation. Consequently, songs with similar labelled genres will pop up high in the list. If, on the other hand, the adaptor is far away from the centre, genre is not highly valued. Songs from different labelled genres are likely to appear in the list.

The list of songs on the left-hand side of the screen is sorted according to the degree of similarity; the songs that are closest to the seed are positioned at the top of the list.

3 USER TEST

To assess the usability and the user benefits of the similarity concept, a user evaluation has been carried out. We assessed *user task performance*, perceived *ease-of-use* and *usefulness*, and *user preference* in a music playlist creation task using a test system and two control systems:

1. User-Driven Similarity system (*UDS*) with a fully controllable similarity measure,
2. Control system 1 (*Control1*) with only timbre similarity,
3. Control system 2 (*Control2*) with a fixed combination of timbre and the other similarity components.

Participants in the test worked with the same visual interface for all systems, with the exception that the similarity manipulation was only available for the *UDS* system. Additionally, the colour of the logos was different for the three systems (i.e., red, blue, green), which allowed us to refer to the systems in the post-experiment questionnaires. For *Control2*, the weights were fixed (according to empirical experimentation) as follows:

$$w_s = 1, w_m = 0.3, w_t = 1, w_g = 0.3, w_y = 0.3$$

3.1 Research questions

The hypotheses that we want to verify are the following:

- The *UDS* system supports users in creating playlists **more rapidly** than the control systems do. It is expected that more control on the similarity definition helps users to find preferred music more easily.
- The use of the *UDS* system gives the user a **better perceived control** in comparison to the control systems.
- The *UDS* system will be perceived **more difficult** to use than the control systems.

- The *UDS* system is **preferred** to the control systems.

3.2 Participants

Twenty-two (22) persons (15 male, 7 female) participated voluntarily to the experiment. All participants were frequent listeners to popular and rock music with an average age of 28 years (min: 22 max: 40). All participants had completed higher vocational education. About two third of the participants said that they make or had made playlists in their private life.

3.3 Method

A factorial within-subject design with one independent variable *system* (UDS, Control1, Control2) was used: all participants had to work once with each system. To compensate for order effects, participants were randomly assigned to one of the six possible permutations of admission to the three systems.

A music collection comprising 2248 popular songs extracted from 169 CD albums from 111 different artists covering 7 different musical genres released in the period from 1963 to 2001 served as test material. The test equipment consisted of a personal computer, on which the systems were running, a touch screen tablet, on which the users could control the system and a Philips Streamium MCI-250 audio set to render the music.

3.4 Procedure

Participants were invited for the experimental session in a prepared office room. A few days before the test, they were provided with a paper list of all artists whose music was used as material in the experiment. They were asked to indicate what artists they knew and like, did not like or did not know. This task was primarily done to get participants acquainted with the music used in the test.

At the start of the session, participants were handed over the general and the detailed instructions for each task of the experiment. Together with the instructions, two example tasks were provided. Participants were encouraged to replicate the examples to get acquainted with the systems. After performing the tasks, participants were given the opportunity to practice until they felt comfortable with the system.

The user task consisted of a playlist creation task for each of the three different systems. Participants were asked to create playlists with 10 different songs while imaging the same listening situation. The playlist created in the three trials should be different. While performing the task, music could be listened to as many times as participants desired. No clues were given on how the task should proceed, or how music should be examined and evaluated. Songs could be added, removed, or reordered individually to or from the playlist under construction. Time to perform the task was unlimited and speed of operation was not presented as a criterion of success. Quality of the playlist was presented as the sole optimization criterion. Participants were not told about the nature of the systems and any questions to the experiment leader on this topic were not answered.

After each playlist creation task, participants completed a (adapted) Technology Acceptance Model (TAM) questionnaire [17], as shown in Figure 4, assessing *perceived ease-of-use* and *perceived usefulness* of the interactive system. In our experimental setting, the term *perceived ease-of-use* refers to the extent to which a user finds a playlist creation system easy to learn and use (questions Q1 to Q4). The term *perceived usefulness* refers to the extent to which a user finds the system to be an aid for music selection (questions Q4 to Q5). Participants were asked to rate the questions from 1 (totally disagree) to 7 (totally agree).

At the end of the experimental session, participants ranked the systems according to their preference of use. After the experiment, participants received an e-mail with a link to the three playlists made, from which they could listen to the songs. They were asked to rate the playlist on a scale from 1 (extremely bad) to 7 (excellent).

Please indicate to what extent you agree or *dis*agree with each of the following statements by using a 7-point scale.

Q1. I find learning how to use the system easy.
Q2. I find it easy to get the system to do what I want it to do.
Q3. I find it easy to become skilful at using the system.
Q4. I find the system easy to use.
Q5. I find that by using the system I can make good playlists.
Q6. I find that by using the system I am able to create a playlist rapidly.
Q7. I find that by using the system I enjoy the making of a playlist.
Q8. I find this system useful at home.

Figure 4: Adapted Technology Acceptance Model (TAM) questionnaire

3.5 Measures

Three measure were used in the evaluation, (i) Task performance, (ii) Quality of playlists, (iii) Order of preference.

Task performance was measured by *time-on-task* and *number-of-actions*. *Time-on-task* measured time in seconds that elapsed from the first button press to the last button press. *Number-of-actions* measured the number of clicks performed on the interface (scrolling on the list was also taken into account) by the participant. A rating score on a scale from 1 (extremely bad) to 7 (excellent) measured the perceived *quality of a playlist*. The *order of preference* for the three systems was assessed by asking the participants which system they liked most and which system they liked least.

4 RESULTS

4.1 Task Performance

In a first analysis, we did not find any statistically significant effect on the *time-on-task* and on the *number-of-actions* measures. However, a more detailed analysis showed that the participants could be divided into two groups: the *fast* (14 participants) and the *slow* (8 participants). A k-means cluster analysis was used to identify these two group of people based on the *time-on-task* and the *number–of-actions* measures. The group of slow people manifested a strong explorative behaviour: their music selection strategy was influenced by suggestions provided by the system. Instead, the group fast participants focused more on their target and music preferences. In Figure 4, data on *time-on-task* are plotted for the two identified groups: the slow ones are distinct from the fast one by a *time-on-task* of about 9 minutes (540 seconds).

4.1.1 Time-On-Task

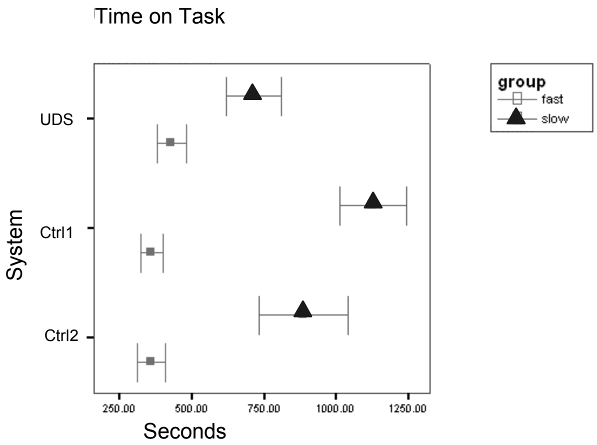


Figure 5: Mean values of *time-on-task* of the fast and slow participants for the three systems (error bars show the standard error of the mean).

A MANOVA (Multivariate ANalysis Of VAriance) with repeated measures was conducted in which the *time-on-task* measure was used as dependent variable and *system* was a within-subject independent variable. The graph shown in Figure 5 suggests that making a playlist by using *UDS* was on average faster than with *Control1*. We found, indeed, a main statistically significant effect (F(3,7) = 5.18, p < 0.05) for the *time-on-task* measure with respect to *system* for the set of eight *slow* participants. On average, it took about 11 minutes (700 seconds) to make a playlists of ten songs with the aid of *UDS* and 19 minutes (1126 seconds) to make the playlist with the aid of *Control1*.

For the *fast* group, it took about 7 minutes (421 seconds) to make a playlist with the aid of *UDS* and about 6.5 minutes (397 seconds) with the aid of *Control1*; these results were not found to be significant.

4.1.2 Number-Of-Actions

A MANOVA with repeated measures was conducted, in which the *number-of-actions* measure was used as dependent variable and *system* was a within-subject independent variable. Figure 6 shows the means and standard errors for the slow participants. We found a main statistically significant effect (F(3,7) = 3.6, p < 0.05) on the *number-of-actions* measure with respect to *system* for the set of eight slow participants. Making a playlist with *UDS* requires fewer actions (on average 236 actions) than making a playlist with *Control1* (on average 395). For the fast group, it took on average 160 actions to make a playlist with the aid of *UDS* and on average 176 actions with the aid of *Control1*; these results were not found to be statistically significant.

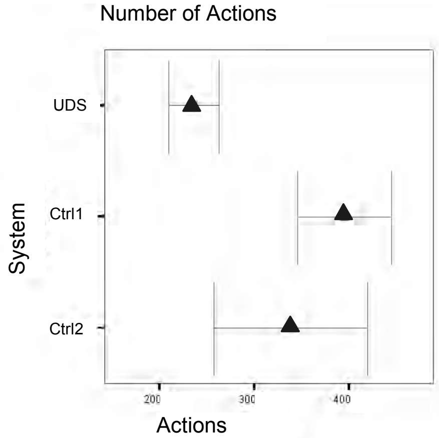


Figure 6: Mean values for *number-of-actions* for the slow participants for the three systems (error-bars show the standard error of the mean).

4.2 Quality of the playlists

A MANOVA with repeated measures was conducted on the full set of twenty-two participants. The *rating-score* was used as dependent variable and *system* was a within-subject independent variable. We found a significant effect (F(3,21) = 3.7, p < 0.05) on the ratings of the playlists with respect to *system*. The playlists generated with the *UDS* system were rated higher (5.8, on average, on a scale from 1 to 7) than those generated with *Control1* (5.0, on average). We did not find significant effects for *UDS* and *Control2*, or between *Control2* and *Control1*. No significant effect between *fast* and *slow* participants was found

4.3 System preference

The results of system preference are shown in Figure 7. Most participants preferred *UDS* and substantiated their choice by saying that they felt more in control in the selection of the similarity. A substantial number of participants (15/22) did not express any preference difference between *Control1* and *Control2*.

4.4 Perceived ease-of-use and usefulness

Responses to the adapted TAM questionnaire for all participants were subjected to a two-dimensional nonlinear principal component analysis (PCA). The eight items in the questionnaire were treated as active variables and the three different systems were treated as passive variables to label the plot (i.e., *Control1*, *Control2*, *UDS*). The responses were treated as ordinal categories; only the order of the 7-point scale was considered important.

The visualisation of the PCA solution of the TAM questionnaire is shown in Figure 8. It displays the mean transformed item responses related to the three different systems, together with the mean scores to the individual items (i.e., Q1 to Q8). The arrows go through the origin and the mean scores of each group of items. These lines represent the 'mean' axes along which the transformed ordinal response categories of the items (i.e., the 7-point scale of the questionnaire) are located.

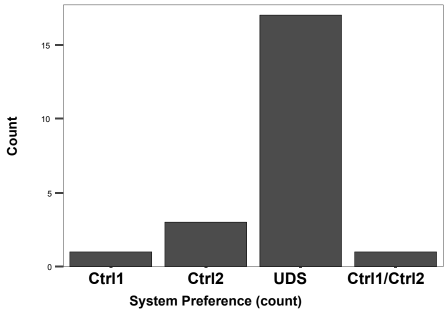


Figure 7: Expressed preferences for the tested systems.

The scores for the items Q1, Q2, Q3 and Q4 are highly correlated as well as the scores for the items Q5, Q6, Q7 and Q8. The high correlations mean, indeed, that the two sets of four items load on different factors that can be labelled as *perceived ease-of-use* and *perceived usefulness*. Figure 8 suggests that *UDS* is perceived as the most useful of the three tested systems, but it is also perceived as slightly more complex to use than the two control systems. In contrast, *Control1* is perceived as the least useful, though there is not much difference in the *ease-of-use* dimension with *Control2*.

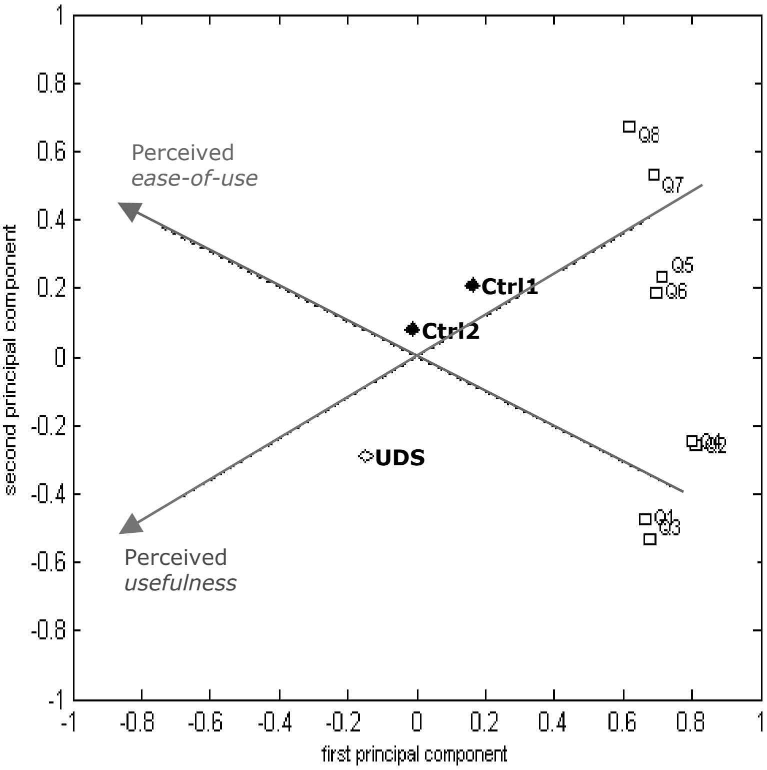


Figure 8: The perceived ease-of-use and usefulness of the tested systems.

5 . DISCUSSION

This experiment evaluated user task performance, perceived *ease-of-use* and *usefulness*, and preference of use of the *UDS* system (with a fully user controllable song similarity feature) in comparison with two control systems: *Control1* and *Control2* (with a non-user-controllable similarity feature).

It was expected that less time and fewer actions are required to make a playlist when using the *UDS* system than when using the control systems. The test revealed that, slow participants needed less time (on average, 7 minutes or about 38% less time) and fewer actions (160 or 40% fewer actions) to complete a playlist with the aid of *UDS* than with the aid of *Control1*. For *Control2*, it took about 4 minutes, equivalent to 21% less time with respect to *Control1* and about 3 minutes more than *UDS*. Based on these results, the hypothesis could not be rejected. Note that the *UDS* system contained an additional repertoire of actions to manipulate the similarity; this additional set of actions did however not negatively influence the total number of actions. For the fast participants, no effects on time and actions were observed. Probably, these participants were already acting at maximal performance level, while leaving little room for improvement by using a different system.

Quality of the playlist was explained to the participants as their sole optimisation criterion with no restrictions on time. Hence, we would not expect a quality effect in the playlists. Nevertheless, playlist made with UDS were rated higher that those made with *Control1*. This suggests that the *UDS* system allowed participants to create better playlists than at least one of the two control systems did.

It was expected that the *usefulness* of *UDS* was perceived higher than the two control systems and that *UDS* would score less on *ease-of-use*. The TAM questionnaire indicated that *UDS* was, indeed, perceived most useful and slightly less easy to use. The two control systems were perceived equally easy to use. Based on these results, the hypothesis could not be rejected.

It was expected that the *UDS* system is preferred to the control systems because of the better control that it offers. The order of preference task found out that, indeed, the *UDS* system was preferred over the *Control1* and the *Control2* systems.

6 CONCLUSION

This paper presented the realization and user evaluation of a music retrieval music that sorts songs on the basis of similarity to a given seed song. The notion of music similarity has been identified as an appreciated tool for end-users to find preferred music in large collections. We believe that music similarity involves the comparison of different song features like timbre, genre, mood, tempo, and year. Moreover, given that the importance of features is heavily dependent on the context and listening intentions at hand, we proposed a system in which the users had complete control on the contribution of each feature to the overall similarity. We believe that our approach to music similarity could provide the tools to break through the glass ceiling discussed in [2]. We have evaluated the usability of such a system in comparison to two control systems in which the user control on the similarity function was diminished. Findings were that users with a more explorative nature who work with the proposed system are able to make better playlists in less time and less effort than in the case of the control systems. Only because of the additional effort to learn to work with the user-driven similarity function during first-time use, most users find the proposed system somewhat less easy to use than the other systems. In conclusion, providing users with complete control on their personal definition of music similarity is found to be more useful and preferred than no control.

REFERENCES

[1] Vignoli, F., Digital Music Interaction concepts: a user study, *Proceedings of International Conference on Music Information Retrieval, ISMIR 2004*, pp. 415-420, 2004.

[2] Aucouturier, J.-J. and Pachet, F., Improving timbre similarity: how high's the sky, *Journal of Negative Results in Speech and Audio Science*, vol. 1, no. 1, 2004.

[3] Berenzweig, A., Logan, B., Ellis, D. P. W., and Whitman, B., A Large-Scale Evaluation of Acoustic and Subjective Music Similarity Measures, *Proceedings of 4th International Conference on Music Information Retrieval, ISMIR 2003*, pp. 99-105, 2003.

[4] Logan, B. and Salomon, A., A music similarity function based on signal analysis, *Proc.of IEEE Int.Conf.on Multimedia and Expo (ICME)*, 2001.

[5] Pampalk, E., Dixons, S., and Widmer, G. On the evaluation of perceptual similarity measures for music. Proceedings of the 6th International Conference on Digital Audio Effects (DAFx-03), 2003.

[6] Wolpert, R. S., Recognition of melody, harmonic accompaniment, and instrumentation: musicians vs. non-musicians, *Music Perception*, vol. 8, no. 1, pp. 95-106, 1990.

[7] Cupchik, G. C., Rickert, M., and Mendelson, J., Similarity and preference judgment of musical stimuli, *Scandinavian Journal of Psychology*, vol. 23, pp. 273-282, 1982.

[8] Pauws, S. and Eggen, B., Realization and evaluation of an automatic playlist generator, *Journal of New Music Research*, vol. 32, pp. 179-192, 2003.

[9] Baumann, S., Pohle, T., and Shankar, V., Towards a socio-cultural compatibility of MIR systems. *Proc. of 5th Int. Conf. on Music Information Retrieval. 2004*, pp. 460-465, 2004.

[10] McKinney, M. and Breebaart, J., "Features for audio music classification", *Proceedings of 4rd International Conference on Music Information Retrieval, ISMIR 2003*, pp. 151-158, 2003.

[11] Kohonen, T., *Self-organizing maps,* Springer, 1995.

[12] Tversky, A., Features of similarity, *Psychology Review*, vol. 84, no. 4, pp. 327-352, 1977.

[13] Gowda, K. C. and Diday, D., Symbolic Clustering using a new dissimilarity measure, *Pattern Recognition*, vol. 24, no. 6, pp. 567-578, 1991.

[14] Tzanetakis, G. and Cook, P., Musical Genre Classification of Audio Signals, *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293-302, 2002.

[15] All Music Guide. All Music Guide. http://www.allmusic.com, last accessed 20-04-2005.

[16] MoodLogic. http://www.moodlogic.com, last accessed 20-04-2005.

[17] Davis, F. D., Perceived usefulness, perceived ease-of-use, and user acceptance of information technology, *Management Information Science Quarterly*, vol. 18, pp. 189-211, 1989.

[18] Whitman, B. and Smaragdis, P. Combining Musical and Cultural Features for Intelligent Style Detection, *Proceedings of International Conference on Music Information Retrieval, ISMIR 2002*, 2002.

[19] Pampalk, E. and Dixon, S. and Widmer, G. Exploring Music Collections by browsing different views, *Proceedings of International Conference on Music Information Retrieval, ISMIR 2003*, pp 201-208, 2002.