Musical Mood-Based Mobile Gaming

A. HODGE, K. COLLINS and P. J. TAILLON

Abstract—**This paper explores ways of using mood-based extraction methods on player-selected music to drive content in mobile video games. Specifically, we describe the methods employed in the development of a game engine adapting the CLAM C++ Library for the Apple iPod.**

I. INTRODUCTION

Customized music can be used to drive game elements. This concept was first introduced by *Vib-Ribbon* in 1999, in which the music chosen by the player created the structure of the gameplay through altering the game's generation of level mapping. The game scanned the user's CD and made two obstacle courses for each song (one easy and one difficult), so that the game was as varied as the music the player chose. Since *Vib-Ribbon*'s release, there have been a handful of games to use the player's music as a way to alter game content. *Phase*, for instance, was designed by Harmonix for the fifth generation iPod. *Audiosurf,* a puzzle-racing game that uses the player's input music files to design the game maps gained much attention after its release in 2008. These games rely on extracting information from the player's music to procedurally create gameplay elements. Until recently, such games have primarily relied on beat mapping and tempo. Rosso, Tzanetakis and Gooch describe a method for analyzing player actions and translating it into a beat sequence synchronized to an existing musical track [1]. Arrasvuori and Holm developed a more extensive approach in the creation of two game prototypes where they explore a wide variety of musical properties that could be used to drive gameplay, but do not mention mood [2].

In this paper, we propose a method of adding the concept of musical mood to drive gameplay. Here, we describe our approach to adapting open source software library CLAM for the iPod, and explain some of the methods used for developing musical mood extraction algorithms. Our engine is still in prototype stage, but we anticipate having a working model ready in 2011.

II. TECHNICAL DESCRIPTION

The Apple iPod makes a logical choice for a music–driven game, since players will already have their own musical selections stored on the device, it is a popular device for both music listening and casual gaming, and there is ample support for game developers. We are using a 4th generation iPod Touch (iOS 4.3.3). We ported the existing CLAM C++ library to the device. CLAM is an audio signal extraction and analysis library with a strong tonal analysis implementation. We have opted to use the best elements of the CLAM library alongside existing hardware-optimized frameworks. Specifically, Apple's Accelerate framework contains many DSP calculations necessary for audio and mood feature extraction, and so these are favoured over CLAM's existing FFTW (Fastest Fourier Transform in the West)-based analysis.

Our approach to mood extraction is currently focused on two areas: 1) descriptive tonal analysis, both on the instant tonality (chord) and the overall tonality (key) of the piece, and 2) a tempo and time signature analysis, which follows a similar implementation but filters high frequency ranges and performs basic onset detection on the resulting low frequency spectrum. An overview of this algorithm is presented in Fig 1.

Before any analysis begins, the source file is converted to 16 Bit Linear PCM WAV, which is a popular uncompressed digital music format. The Accelerate Framework's vDSP functions are used to perform a Forward Fast Fourier Transform on a split complex vector representation, and the output returns a split real vector. In this form the bin's frequency can be calculated by squaring the magnitude of the bin value. The Constant Q Transform, where Q is the ratio of center frequency to bandwidth, is implemented following CLAM's approach and that described by Brown and Puckette [3]. The result of this transform, and the feature that makes it particularly useful to us, is the returned bin values are quantized to the logarithmic frequency scale used to measure sound. This can result in accuracy of one bin per semitone, or 12 bins per octave, and a pitch class profile is built on the resulting power spectrum This is the foundation on which we can make deductions about the tonal content and associated moods of a user's iPod Library.

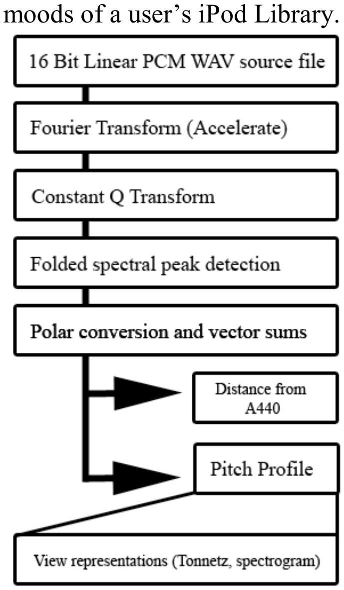

Fig. 1 – An overview of an audio feature extraction algorithm being implemented for research on music-driven games.

III. Developing Mood Detection: Methods

Musical mood is largely a culturally specific phenomenon. While certain aspects have a universal element to them (due primarily to bioacoustic properties: for instance, low bass is heard as threatening), most of the semiotics of music are due to cultural convention. In the West, this convention has evolved through a combination of folk musics and European art music. We have chosen to use these Western conventions for our purposes of determining mood. In other words, our mood detection may only work on Western musical styles. We developed a database of musical elements (harmonic content, instrumentation, tempo, articulation, timbre, pitch range, and dynamics) that contribute to each mood through a number of different related projects.

A. Distributed Classification Games

Distributed classification is a method of collecting a large number of responses of multiple users, commonly used for meta-data. Users tag media objects with text keywords in a free-association fashion. Tags can then be combined into non-hierarchical groups of associated terminology. In order to engage the audience and increase the amount of tags collected, tagging games can be created [4]. Approaches that encourage players to tag musical data have been developed [5][6]. We developed two distributed classification games and collected data over two years [7].

B. Sheet Music Analysis

Photoplay scores are the sheet music used by the pianist that would accompany so-called "silent" films. These short pieces of music were typically arranged according to genre of movie and mood, and were accompanied by keywords that would represent the intended type of scene or mood associated. For example, the 4-bar *Mysterioso* by Julian Rutt was labeled "burglars and creepy business" [8]. We converted the sheet music into usable MIDI files and stored the keywords as metadata associated with the file.

C. Existing Databases

We drew on existing databases that categorize popular songs according to mood, such as Moodstream, AllMusic and Aupio. In Aupio, songs are categorized according to mood keyword. We obtained as many MIDI files as possible of these songs and stored the keywords as metadata in a database.

D. Artificial Neural Networks

Using the MIDI files and affiliated metadata, we used the MIDI Toolbox plug-in for Matlab to develop artificial neural networks to scan and compare tagged MIDI files [9]. A neural network uses software to explore data and search for patterns, such as if there are any melodic or harmonic patterns in all of the 'sad' songs. MIDI is limited in that it cannot represent timbre, spatialization or voice. However, we were able to explore patterns of harmony, rhythm and melodic contour within given mood groups.

IV. Next steps and future work

We have a number of challenges before the system is ready to be implemented into prototype games. We are currently exploring different methods of both beat mapping and timbre analysis (using harmonic distortion and reverb tails) which is not part of the CLAM library, and we are testing our iPod system for accuracy. The iPod is limited in its processing and memory, and simultaneously analysing music and generating graphics/gameplay will likely introduce some lag. A pre-scan of the player's song will solve this problem, but if the player wishes to change music in the middle of gameplay, this may cause problems.

Games that procedurally generate or alter content based on music can be limited. Certain types of music work better than others in these types of games, of course, particularly when it comes to beat mapping—abstract ambient music tends to throw off these types of games, and so the player's experience can vary greatly, depending on their choice of music. Using musical mood and other musical parameters is one way to overcome this problem, but how effective it will be in generating interesting gameplay remains to be seen. Moreover, we have not yet determined how we will map the musical mood to game parameters. Arrasvuori and Holm describe a list of game parameters that could respond to music, including for instance speed of game, type of objects, location, properties or behaviour of objects, camera angle, rules, scoring, and so on [2]. We will be drawing on their list in developing musical mood-based game prototypes.

Our next step is to design some games that will allow us to best use musical mood as a parameter in the game, and explore ways of mapping mood to gameplay events that will make gameplay interesting and unique.

References

[1] S. Rosso, G. Tzanetakis and B. Gooch, "Adapting Personal Music for Synesthetic Game Play." *FDG 2010*, June 19-21, Monterey, CA, USA.

[2] J. Arrasvuori, and J. Holm. "Background Music Reactive Games," *MindTrek 2010*, October 6th-October 8th 2010, Tampere, Finland.

[3] J. Brown, M. Puckette, "An efficienct algorithm for the calculation of a Constant Q Transform" June 16 1992 http://www.wellesley.edu/Physics/brown/pubs/effalgV92P2698-P2701.pdf

[3] L. von Ahn and L. Dabbish, "Labeling images with a computer game". *ACM Conference on Human Factors in Computing Systems*, 2004, pp. 319–326.

[4] E. Law, L. von Ahn, R. Dannenberg and M. Crawford, "Tagatune: A Game for Music and Sound Annotation**"** *Proceedings of the International Conference on Music Information Retrieval* (2007) http://ismir2007.ismir.net/schedule.html .

[5] D. Turnbull, R. Liu, L. Barrington and G. Lackriet, "A Game-Based Approach for Collecting Semantic Annotations of Music" *Proceedings of the International Conference on Music Information Retrieval* (2007) http://ismir2007.ismir.net/schedule.html.

[6] K. Collins, "Generating Meaningful Sound: Quantifying the Affective Attributes of Sound Effects for Real-Time Sound Synthesis." *Audio Engineering Society 35th International Conference: Audio for Games*, London, UK. February 11-13. 2009. London, UK.

[7] E. Rapée. *Motion Picture Moods for Pianists and Organists*. New York: G. Schirmer, 1942.

[8] O. Lartillot, P. Toiviainen, and T. Eerola. A matlab toolbox for music information retrieval. In Preisach, C., Burkhardt, H., Schmidt-Thieme, L., and Decker, R., editors, *Data Analysis, Machine Learning and Applications*, Data Analysis, Machine Learning and Applications: Studies in Classification, Data Analysis, and Knowledge Organization, pp. 261–268, Berlin, Germany. Springer. (2008).