Autonomous Robot Dancing Synchronized to Musical Rhythmic Stimuli

Catarina B. Santiago
DEI, FEUP and ROBIS, INESC – Porto
Porto, Portugal
catarina.santiago@fe.up.pt

João Lobato Oliveira
DEI, FEUP, NIAD&R - LIACC and UTM INESC- Porto
Porto, Portugal
joao.lobato.oliveira@fe.up.pt

Luís Paulo Reis
DEI, FEUP and NIAD&R - LIACC
Porto, Portugal
lpreis@fe.up.pt

Armando Sousa
DEEC, FEUP and ROBIS, INESC-Porto
Porto, Portugal
asousa@fe.up.pt

Abstract—**This paper presents a system that is able to control in real time a humanoid robot to perform dance movements to the rhythm of a music. The movements' coordination is performed with the aid of a music analyzer that estimates the beat of the music and calculates a prediction of the next inter-beat-interval (IBI). During the robot dancing performance, a dance movement is chosen from a pre-defined dance library and is executed on-the-fly by the robot. The movements' velocity, as well as the attended metrical-level, are adjusted in real time so that the movement can be executed within the time interval of two beats, and at the same time taking into consideration the robot motor-rates limitations. Results evince a good synchrony of the movement towards the analyzed rhythm, but points to the need of a careful design of the dance movements so they can be naturally executed on time.**

Keywords-robotics; robot dancing; musical rhythm analysis.

I. INTRODUCTION

Dance had always played an important role on human development and behavior. From a complex point of view dance can be seen as a mean of nonverbal communication, although people are usually only interested in its artistic and playful side. Most of the times dance is strongly associated with music and its movements depend on well-defined music properties. Rhythm is the music element that most influences the dancing performance, and it includes several aspects such as the beat, the meter and the tempo.

Although dance is something commonly done by humans and some animals (*e.g.* bees and birds) it seems appropriate to extend it to robotics. This extension will for sure provide new forms of entertainment. The basis for extending dance into the robotics field lies on the fact that in recent years robotics' manufacturers have devoted a great effort on developing robots that resemble either animals [1], [2] or humans [3], [4], [5], not only in shape but also in the ability to perform movements.

In this paper we present an intelligent system based on a humanoid robot that is able to perform dance movements attuned to the musical beat and conditioned by the surrounding environment. The remainder of this paper is divided in 4 sections. The first section provides some background information and related work. The following section provides a detailed explanation of the implemented architecture. The last two sections show the results achieved so far and draw some conclusions and future work.

II. RELATED WORK

Robotics has suffered a huge development since the first robots appeared and nowadays robots are being used even for entertaining activities. Applications mixing dancing with robotics have been developed with all kinds of robots, from wheeled to legged robots, and finally humanoids.

Auconturier *et al.* [6] developed a simple wheeled mp3-player-robot, Miuro. Their approach is not based on a predefined library but rather on commands generated on-the-fly that control the robot movements. The beats generated by their music analyzer are propagated through a FitzHugh-Nagumo neural network to generate a train of pulses. This train of pulses is afterwards converted into velocities for the right and left wheels of the robot. Indeed they don't have a pre-programmed dancing library, but the physical framework they use, Miuro, is very simple when compared with a humanoid robot (only has two servos, one for each wheel).

A more sophisticated wheeled robot was used by Oliveira *et al.* [7]. Their robot besides having two wheels also used servos to control the arms, the hip and the head of the robot. Their system analyzed rhythmic qualities of the music, in the form of note-onset events, using an onset detection algorithm with adaptive peak-picking, implemented on the MARSYAS framework [8]. Depending on the triggered threshold considered by their peak-picking algorithm, three rhythmic events are sent to the robot: strong, medium and soft. These events in conjunction with two other sensorial events (dance-floor colors under the robot and proximity to an obstacle) define the robot movements to be executed, according to the dance composition user-defined *a priori*.

Several researchers use humanoid robots due to their similarity with humans. For example, Nakahara *et al.* [9] used a "Tai-chi" humanoid robot with 21 servos, which comes with a motion editor that enables the definition of key positions that are saved as text files (21 motor angle positions). To generate the actual movements, angle positions and execution-time between positions must be sent to the robot. Their music analyzer is able to predict the inter-beat-interval (IBI) time of the given musical input on-the-fly, and uses this information to calculate the time needed for transiting between key-positions.

Eldenberg *et al.* [10] also used a low-cost humanoid robot. Their work uses predefined gesture libraries, one for the arms and another for the legs. Based on the song tempo and beat, gesture commands for the right arm, left arm and legs are passed to the robot. The gesture commands are sent to the robot before the predicted beat-time in order to maintain synchrony with the music. To determine the moment to issue the commands they take into consideration the specific execution time of each gesture. This approach can carry some problems namely during the execution of slow commands. If a new gesture command arrives while still performing a slow gesture the new gesture command is ignored and will not be performed.

We also propose to develop a system based on a low-cost humanoid robot that is able to dance attuned to the musical beat, while overcoming its motor-rate limitations. The platform chosen to implement this solution is based on Robonova-I [11] and uses the IBT real-time audio beat-tracker [12] to predict the beats of a musical input.

III. Architecture

As stated before, our robot dancing framework was tested on Robonova and uses a Music Analyzer that is able to track the beat-times of a musical input file. The choice of Robonova is related with being an off the shelf solution and having enough degrees-of-freedom (sixteen) to enable the creation of structured dance movements.

The system is based on a modular architecture composed of two sub-systems: the Music Analyzer, that is responsible for analyzing the musical rhythm; and the High Level Control that performs the bridge between the Music Analyzer and the physical robotic platform, Robonova. An overview of the system architecture, decomposed on its functional modules is presented on Fig.1.

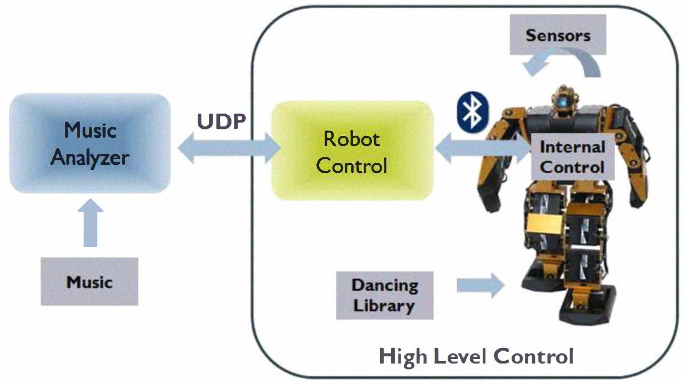

Fig. 1. Robot dancing system architecture.

A. *Robotic Platform*

Robonova [11] is a small humanoid robot (31cm height) developed by Hitech which is controlled by 16 HSR-8498HB digital servos. It uses a simple microcontroller, CR3024 that runs at 7.81 MHz, making it restrict for performing basic arithmetic and decisions. Yet, it supports serial communication, which enabled the use of a Bluetooth dongle for bi-directional communication between its internal robot control and an external processing device, assuring higher CPU processing by a computer.

B. *Music Analyzer*

The Music Analyzer consists on an audio real-time beat tracker, IBT [12], based on a competing multi-agent system which considers multiple tempo and beat hypotheses. It communicates its beat estimates and inter-beat-interval predictions to the High Level Control on a real-time basis, which on its hand, if necessary, replies with requests for locking the beat tracking onto a certain metrical-level, in order to assure the robot accomplishes the considered dancing motion in phase (robot's "preferred tempo"), overcoming its motor limitations.

The metrical-level interchange is handled by IBT by generating a leading agent which may follow a tempo hypothesis double or half the one pursuit by the best agent at that moment.

C. *High Level Control*

In order to control the Robonova movements' execution we integrated a High Level Control interface responsible for analyzing each beat estimate from the Music Analyzer and the robot internal feedback, issuing the necessary adjustment commands for keeping the motion-beat coordination. The communication between these two sub-systems (Music Analyzer and High Level Control) is performed via UDP sockets in order to allow for a fast message interchange. Since this is a local network, and the message flow is not that high reliability issues are not problematic.

The High Level Control sub-system combines four sub-modules: Dancing Library, Sensors, Internal Control, and Robot Control which are described in the following subsections:

1) Dancing Library

The dancing library is composed by several basic dance movements, which are accessed through the Internal Control module, and are stored inside the robot microcontroller.

This library was manually built using the motion editor software provided by the manufacturer. With this software it is possible to capture all motor values by placing the robot in a desired position. The capture of all the intermediate positions, as key-poses, allowed to basically describing the intended movements, since the robot takes care of the transition between the captured positions, via linear interpolation. In order to create a fluid motion sequence, the movements obeyed to two main characteristics: they were cyclically repeated; and they were composed by four ordered steps, where each step corresponds to the transition between two key-poses, which must be performed within an inter-beat-interval. Fig. 2 exemplifies a basic movement for an arm of the robot, described into a cyclic repetition of four ordered key-positions.

Each step of a movement is in such way characterized by its initial and final positions, by a normal velocity and a delta velocity. The normal velocity corresponds to the velocity on what a given step should be performed in order to be completed within 0.5s (a "medium-sized" IBI). The delta velocity allows increasing or decreasing the velocity at which each step is performed in order to accommodate the robot velocity to the next predicted IBI.

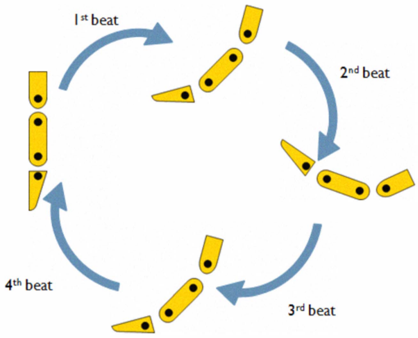

Fig. 2. Example of a cyclic movement described for the robot arm.

2) *Sensors*

Robonova can be equipped with several different sensors that include gyroscopes, distance, light, tilt, touch and sound sensors. The developed application made use of a sonar (MaxSonar EZ1 ultrasonic range finder) for detecting objects that may be near the robot, in order to provide a sort of interaction between the robot and its surrounding environment. Therefore, whenever the robot detects a close-by object, it stops dancing and starts tapping one of its feet in order to inform that it doesn't have enough space to dance. This kind of behavior provides some personality to the robot, enhancing the animacy of its live performance.

3) *Internal Control*

The Internal Control module receives commands directly from the Robot Control and acts accordingly. Since the Internal Control is encoded in the robot microcontroller and the Robot Control on a desktop computer, the communication between the two modules is wireless, so that the robot can move freely without having any wires. As stated, the communication is performed through a Bluetooth module using the Cable Replacement Protocol (RFCOMM) which allows emulating a RS-232 serial communication.

The Internal Control can receive "issue movement" and "change velocity" messages and reply with "movement done" and "impossible to change velocity". There is also a sub-command of the "issue movement" which calls for "continue movement". Whenever a serial command arrives, the Internal Control first of all checks the sonar value and verifies if the robot has space to dance. If an object is too near the robot it starts tapping its foot and ignores the command. Otherwise it will analyze the command received from the Robot Control module, as referred above.

In case of a "change velocity" request, the delta velocity variable (see Dancing Library subsection) is updated so that the robot can perform the movement faster or slower in order to keep up with the beat estimate. This delta velocity can range within defined upper and lower limits so that the robot continues to move while keeping its balance. Whenever the maximum or minimum delta velocities are reached the Internal Control module informs the Robot Control module by issuing an "impossible to change velocity" command.

If the command received is an "issue movement" then the Internal Control requests the robot to perform the basic movement. The "issue movement" command can be of different sub-types which depend on the dancing movement selected from the Dancing Library. The Internal Control module is not responsible for choosing the basic movement but only to issue the instructions needed to perform it. The Robot Control module is the one responsible for choosing the specific movement.

Fig. 3 illustrates the BASIC code of a sample movement described in the Dancing Library, including the handling of the motion velocity adjustment and the signaling of completed steps and movements.

4) *Robot Control*

The interface between the Music Analyzer and the Internal Control modules is performed via the actual Robot Control module. When the first beat information arrives from the Music Analyzer an "issue movement" command is sent to the Internal Control through the serial communication channel.

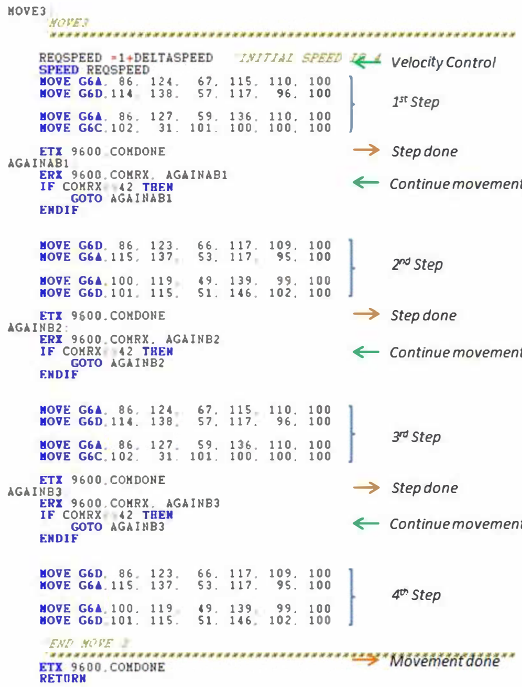

Fig. 3. BASIC movement code sample.

Once the robot finishes the first step of the movement it informs the Robot Control by issuing a "movement done" command. With this exchange of messages it is possible to determine the time the robot took to perform the step (physical time) which corresponds to the current time minus the time when the issue command was sent. By comparing this physical time with the next inter-beat-interval predicted by the Music Analyzer it is possible to determine if the robot is synchronous with the beat or not, and if there is the need to change the velocity of the robot.

Although the synchrony is verified in all the four steps of a movement, the actual "change velocity" command is only issued when the entire movement has been completed, this way avoiding to have a too nervous control of the robot velocity. The "change velocity" command can either request to increase or decrease the velocity. Besides evaluating the need to change the robot's velocity, the Robot Control module is also responsible for issuing the "issue movement" or the "continue movement" commands, depending if the entire movement has been completed or not. These commands are only sent to the robot if meanwhile a beat arrived from the Music Analyzer. In fact, if the robot finishes the step/movement before a new beat arrival it will stay stopped until then.

The Robot Control can also receive an "impossible to change velocity" command that signals the impossibility of the robot to further increase or decrease its motor velocity. If this is the case the Music Analyzer module is informed to adjust its attended metrical-level. Diagram of Fig. 4 illustrates how the Robot Control module works and gives an insight of the data-flow and messages' exchange between the three main modules: Internal Control, Robot Control and Music Analyzer.

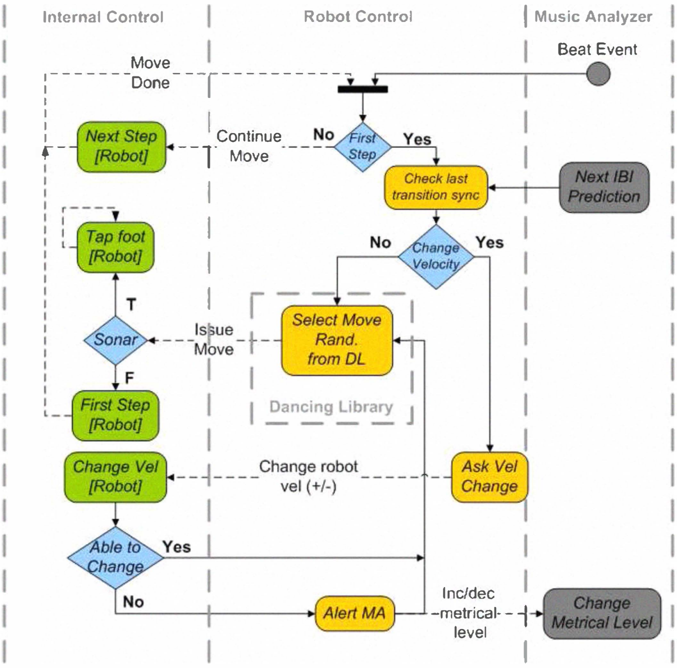

Fig. 4. Data-flow and messages' exchange between the main modules of the proposed robot dancing control architecture.

IV. Results

Preliminary tests shown that, although it is easy to capture the key-positions using the motion editor software provided by Robonova, special attention must be paid during the interpolation between poses so that the robot can keep its balance. When composing the gesture library it is also necessary to keep in mind that the motion steps must be simple enough to be transited within one inter-beat-interval, in order to keep the rhythmic synchrony.

In order to evaluate such synchronism between the robot performance and the estimated beat-times, retrieved by IBT, we tested not only if the robot started the movement in phase with the beat but also how well the robot and IBT could adapt to each other (change motors velocity versus change metrical-level). Tests were conducted with two different movements and two different songs. The four key-positions of the two evaluated dance movements, *mov1* and *mov2*, can be seen on Fig. 5. Both songs correspond to excerpts of Pop/Rock music with rather stable tempi, identified as *music1* and *music2*.

Fig. 5. *mov1* (up) and *mov2* (down) description in key-positions.

Besides *mov2* being more complex than *mov1* it also takes more time for each of the steps to be executed. Figs. 6 to 9 show plots comparing the phase alignment between the estimated beats (in blue), by the Music Analyzer, and the triggered key-poses by the Robot Control (in red). The green arrows describe requests of the Robot Control to the Internal Control for increasing (by arrows without borders) or decreasing (by arrows with borders) the velocity of the movement execution. Finally, the orange dots represent requests made to the Music Analyzer for increasing (by dots without borders) or decreasing (by dots with borders) its attended metrical-level.

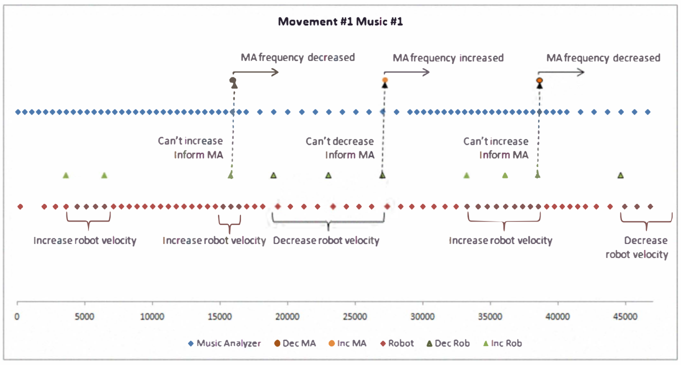

Fig. 6. Performance of the robot for *mov1* with *music1*.

On the first music (Fig. 6), the robot started to dance but it was not able to keep in phase with the beat estimates, which caused the Robot Control to request a velocity increase to the robot actuators. During the following 9 seconds the robot was able to keep up with the beat. After these 9 seconds the Robot Control detected that the robot was again not synchronous and issued another velocity increment of the robot movement. However, this time, the robot was not able to increase its velocity further, which caused the Robot Control to inform the Music Analyzer to decrease its attended metrical-level. Since the robot was at its maximum speed when the Music Analyzer decreased its velocity this caused the robot to become not synchronous again, which resulted on a new "decrease robot velocity" request. This cycle of increase and decrease velocity (robot)/metrical-level (Music Analyzer) continued until the music ends.

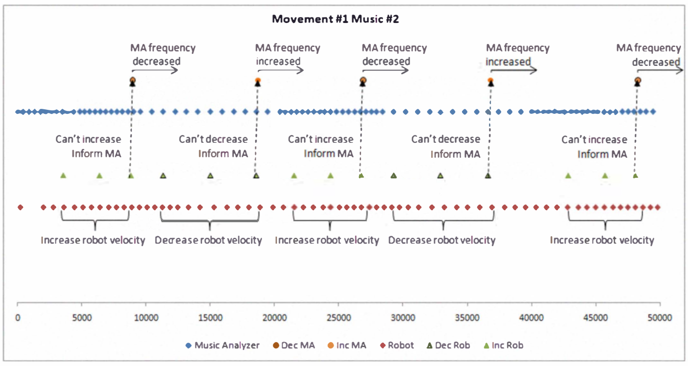

Fig. 7. Performance of the robot for *mov1* with *music2*.

The second music has a more variable tempo which caused more changes on both the velocity of the robot movement as well as on the considered metrical-level by the Music Analyzer (Fig. 7).

As stated before, *mov2* is more complex and each step takes more time to be executed. These two factors contributed to a worse performance over both *music1* and *music2*, as evinced in Figs. 8 and 9.

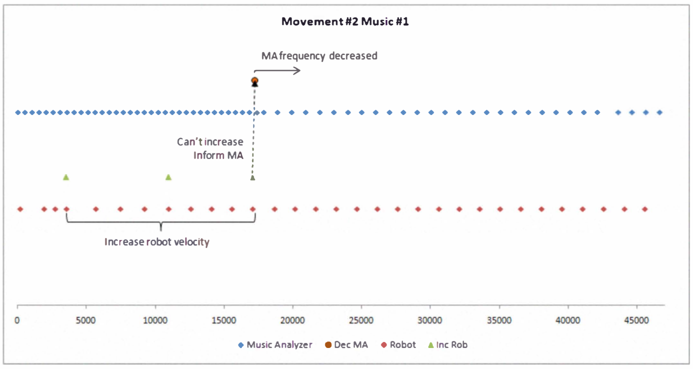

Fig. 8. Performance of the robot for *mov2* with *music1*.

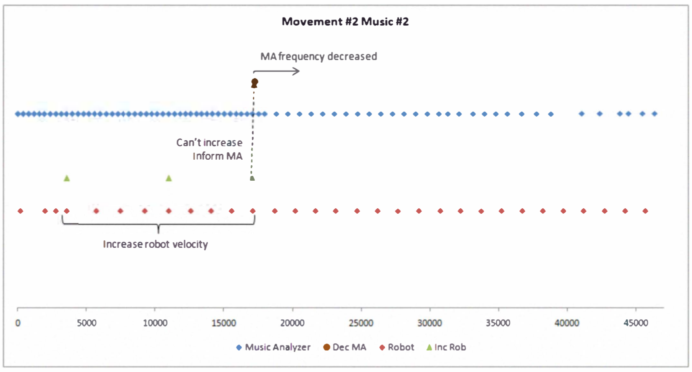

Fig. 9. Performance of the robot for *mov2* with *music2*.

In fact on both musics some beats were simply ignored by the robot since the time of step execution took half or more the time of the current inter-beat-intervals.

V. CONCLUSIONS

In this paper we presented a method for controlling a dancing Robonova. The main objectives were to develop a system that could control the robot movements based on real-time beat estimates (phase) and inter-beat-interval predictions (period). The developed system is composed by two main sub-systems, one responsible for analyzing the musical rhythm, the Music Analyzer and the other responsible for controlling the dancing performance – High Level Control. The High Level Control sub-system on its hand combines several modules: Dancing Library, Sensors, Internal Control and Robot Control.

The Gesture Library is composed by a set of simple movements, each composed by four key-steps. The Internal Control module is responsible for sending low-level commands to the robot motors as well as to collect information related with the end of a movement, or the velocity of the robot, and even to assess information from the robot sensors. With this information as well as with the beat estimates supplied by the Music Analyzer, the Robot Control module is able to control the robot with musical synchrony. This control consists not only on requesting the robot to perform a movement but also to change its velocity in order to keep it in phase with the beat. Such coordination is achieved in two ways: either by changing the velocity at which a movement is executed or by changing the Music Analyzer metrical-level.

Results showed that in order to have a satisfactory control of the robot the movements must be carefully designed, taking into consideration not only the balance of the robot, but also the time each step takes to execute. It was also possible to verify that a control based not only in movement commands but also in velocity commands can achieve very good results. These results are further improved with the ability to change the Music Analyzer metrical-level accordingly.

This work established the basis for an autonomous Robonova Dancing platform, synchronized to musical rhythmic qualities. Despite using the sonar for interacting with the robot during its dancing performance, it would be interesting to explore other kinds of sensors, namely gyroscopes to detect if the robot is standing up or cameras to detect possible obstacles or objects the robot could interact with. It would also be interesting to enlarge the Dancing Library with more dance movements, also enhancing their level of expressiveness and style.

Although we were able to plot metrics for quantitatively evaluating the level of beat-synchrony of the robot dancing performance, we believe that visual effect plays an important role on this kind of work, and therefore we are aware that we still need to perform a qualitative analysis of the system by collecting people's opinion about the robot overall performance.

Acknowledgment

This work was supported by a PhD fellowship endorsed by Fundação Calouste Gulbenkian, with ref. 104410, and another endorsed by FCT, with ref. SFRH/BD/43704/2008.

References

[1] "Genibo", http://www.plasticpals.com/?p=2111, accessed 15 February 2010.

[2] "Sony AIBO Europe – Official Website", http://support.sony-europe.com/aibo/index.asp , accessed 15 February 2010.

[3] "Honda Worlwide | ASIMO", http://world.honda.com/ASIMO/, accessed 20 February 2010.

[4] "Aldebaran Robotics, the creators pf NAO | Aldebaran Robotics", http://www.aldebaran-robotics.com/eng/, accessed 20 February 2011.

[5] "Robonova – MABL_Wiki", http://mablresearch.rit.edu/wiki/index.php/Robonova, accessed 20 February 2011.

[6] J. J. Aucouturier, Y. Ogai, T. Ikegami, "Making a Robot Dance to Music Using Chaotic Itinerancy in a Network of FitzHugh-Nagumo Neurons", in Neural Information Processing, Lecture Notes in Computer Sciences, 2008, vol. 4985, pp. 647-656.

[7] J. Oliveira, F. Gouyon, L. P. Reis, "Towards an Interactive Framework for Robot Dancing Applications", in Proceedings of the 4th International Conference on Digital Arts, 2008, Porto, Portugal.

[8] "Marsyas", http://marsyas.info/, acessed on 15 February 2010.

[9] N. Nakahara, K. Miyazaki, H. Sakamoto, T. Fujisawa, N. Nagata, R. Nakatsu, "Dance Motion Control of a Humanoid Robot Based on Real-Time Tempo Tracking from Musical Audio Signals", in Entertainement Computing – ICEC 2009, Lecture Notes in Computer Science, Springer, Heidelberg, 2009, vol. 5709, pp. 36-47.

[10] R. Ellenberg, D. Grunberg, Y. Kim, P. Oh, "Exploring Creativity through Humanoids and Dance", in Proceedings of the 5th International Conference on Ubiquitous Robotics and Ambient Intelligence, Seoul, South Korea, 2008.

[11] Hitec Robotics, "Robonova-I 16 Servo Edutainment Robot Box Kit Instruction Manual", English Language 1.00.

[12] J. L. Oliveira, F. Gouyon, L. G. Martins, L. P. Reis, "IBT: A Real-time Tempo and Beat Tracking System," in International Conference on Music Information Retrieval (ISMIR), 2010, Utrecht, The Netherlands, pp.291–296.

[13] J. Oliveira, "Towards an Interactive Framework for Robot Dancing Applications", Master Thesis at Faculty of Engineering of University of Porto, 2008