Interactive Music 2.0:
User-driven ad hoc dance clubs

Yago de Quay
yagodequay@gmail.com

Short Term Scientific Mission Report COST IC0601 - WG3Art

Home: Faculty of Engineering, University of Porto – Portugal
Host: FourMs Laboratories, University of Oslo – Norway

August 1st – September 31st 2010

Abstract

This short term scientific mission investigated the role of interactive media in multi-participatory environments. The research focused on show casing two systems, the *Interactive Dance Club* installation and the *Xsens Performance*. The installation consisted of online-services and on-site motion capturing technologies that enabled people in the club to manipulate sounds. The 15 minute performance marked the first use of the Xsens motion capture technology in the sonic arts, thus requiring the creation of a new sonic interaction model, music composition, and software. Results were seen in four areas: 1) Communication; 2) algorithms; 3) sound and motion; and 4) promotion. During the mission, three exhibitions suggested that this line of research is not that far from providing a new form of interactive entertainment in popular venues and a framework for future works.

Keywords

Interactive music, dance night clubs, web 2.0, motion capture, Xsens

1. Introduction

The term Interactive Music 2.0 is defined here as a system that draws upon the Web 2.0 model of user driven content (1). Interactive Music 2.0 aggregates users' social behaviors and individual attitudes towards dancing in clubs, enabling them to create, modify, reconfigure and share music in real time through live motion sensing technologies. As pointed out by Carlheinz Brandenburg, Director of the Fraunhofer Institute and MP3 co-creator, "we will see forms of listening to music and creating music: intelligent applications will replace our current idea of album or playlist. Recommendation and access to the artists will allow a new way of enjoying music. Interactivity will blur the lines between listening, creating music and computer games" (2). Much has been learned in the area of interactive music up to now, hatching new areas such as music information retrieval, generative music, adaptive music, virtual instruments and the like. However, group movement descriptors and respective music information remain a challenge for most researchers (3) (4).

This mission aimed on filling this research gap by experimenting with sound and body movement in three interactive events using *Motion Capture* (MoCap) technologies. The first was a sonic performance, entitled "Dance Jokey", premiered on August 25, 2010, at the Oslo University's "Department of Musicology's Semester Opening Concert". The second event was an installation, named "Interactive Dance Club", demonstrated on September 4, 2010, at the Gabler night club. While the installation was open for anyone to use, the performance was prepared by the author and a colleague from IFI at Oslo University, Ståle Andreas Van Dorp Skogstad. The third put together the first two events in one single happening held on September 24, 2010, at the Gabler night club. The *Dance Jockey* system was based on the Xsens MVN suit that was supplied by the fourMs Lab at Oslo University. The *Interactive Dance Club* comprised of: 1) two Wii remotes that provided auditory feedback when handled; 2) a user-generated online playlist; 3) an optical marklerless MoCap system for the purpose of triggering notes; 4) a user-driven video projections; and 5) real-time online video stream if the dance floor and regular twitter updates.

MoCap is the process of recording body movements and storing them in a digital format (5). This process is applied in various fields ranging from medicine, sports, computer games and computer animation. MoCap rests on various technologies and these can be either *optical* or *non-optical* (6). Inside the latter we can find the affordable *inertial sensor* systems that encapsulate gyroscopes, accelerometers and magnetometers. A MoCap system that relies on various sensors normally performs *sensor fusion* –the act of combining the data of all the sensors to provide holistic information such as the position of a limb. Sensors can be inside devices such as the Nintendo wii remote or strapped to a suit. *Optical* Mocap systems offer the advantage of being non-intrusive and remove the burden of the dancers carrying wires and devices attached to their bodies. This group is divided in *markerless* and *marker based* systems. Markers are either balls coated with retroreflective material that reflects light, or LEDs that emit their own light. Both systems rely on cameras for tracking body parts and movements. Marker based solutions are more accurate and faster; the markerless solutions are more affordable, less invasive and are easily complemented by relevant tools such as Max/MSP/Jitter and EyesWeb (7). This group is well suited for creative applications. Within the non-optical systems, we find the Xsens MVN technology which is based on miniature inertial sensors. It offers some advantages over other technologies in that is it: 1) relatively lightweight and portable (Suit: 1.9 kg, full system: 11 kg); 2) quick setup time (5-15 min); 3) communication with 3rd party programs; and 4) in my experience, a robust system for on-stage performances (8). Its application was limited to industries, training and simulation, film, games, and medicine –never for sonic interaction.

Body oriented technologies that do not concern movement involve extracting data directly from inside the users such as brain waves, heart rate and skin conductivity (9).

Figure 1 - Xsens suit and suitcase

However, the before mentioned technologies used by some performers had the consequence of widening the gap between the artist and the audience. Modern day dance night clubs and other performances that do not harvest the potential creativity of the audience miss out on the knowledge, div ersity and innovation that user-driven ad hoc dance clubs may provide.

This STSM hoped to create an interactive music model for dance clubs and the first sonic performance using the Xsens technology. To this end, work was done on their development, implementation and diffusion in popular dance night clubs.

The next section will examine more closely the development of the software for the *Xsens Performance* and the installation of the *Interactive Dance Club*. In the following section, preliminary results will be presented and followed by discussion and conclusions.

2. Work

2.1 Xsens Performances

To this end, a line of communication between the Xsens MVN software and the sound engine had to be created in C++ and Max/MSP. The content consisted of a music composition and sound effects. A choreography was rehearsed and the result was performed in two venues. As depicted in **figure 2**, information was channeled through the various organs of the system starting from the dancer to the sound output.

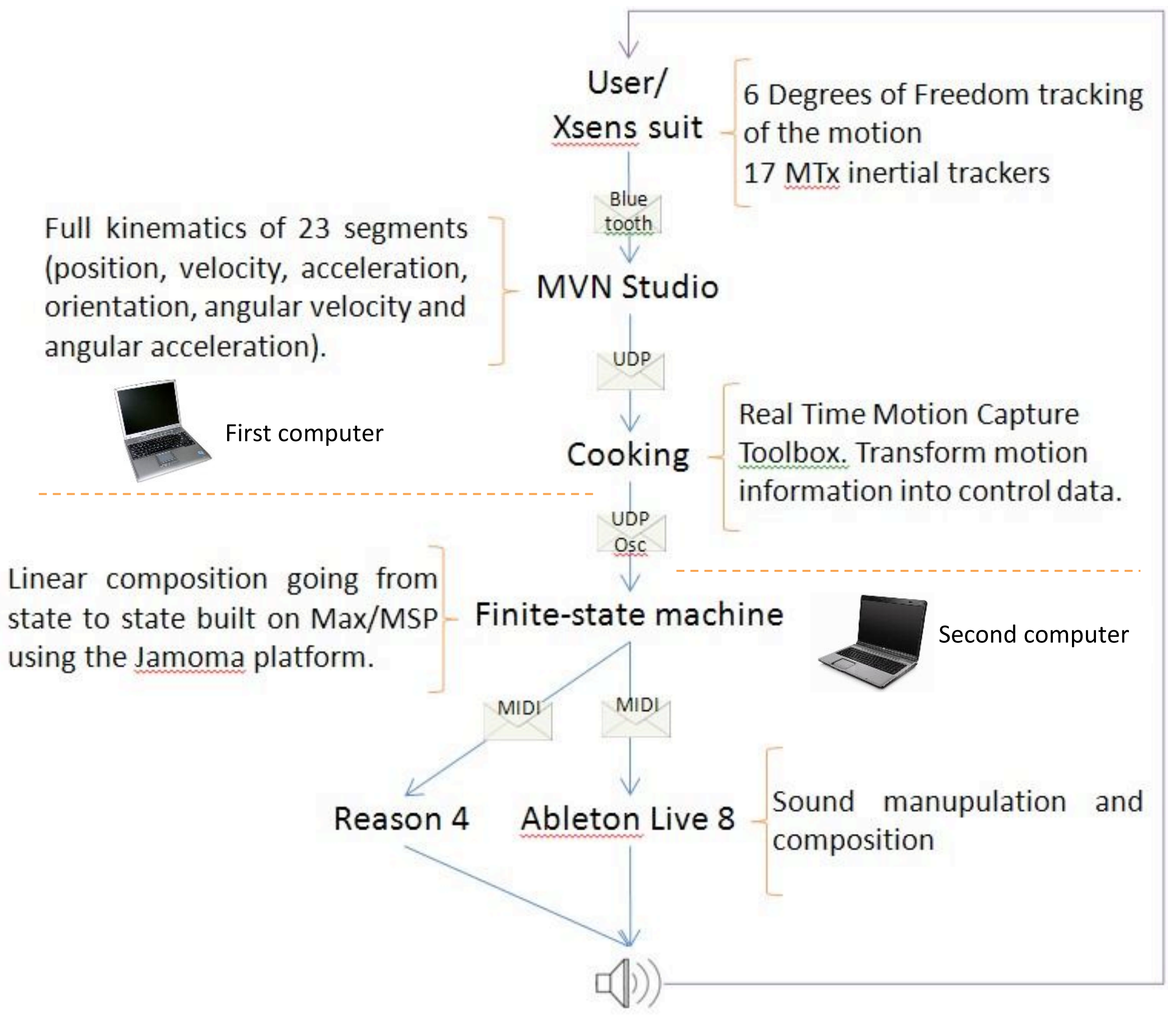

Figure 2 - Xsens data flow

Beyond the MVN software, the rest had to be developed and composed from scratch. The output of the MVN software that came with the Xsens suit needed to be cooked and formatted to fit the OSC protocol used to communicate with the Finite-state machine embedded in Max/MSP. The latter then used the data to control the sound engines; Ableton Live 8 and Reason 4.

2.1.1 Cooking – making sense of numbers

The network stream from the MVN Studio sent information about 23 body segments per frame, a total of 138 floating point numbers at a rate of 120Hz (10). Moreover this there is also combined data such as the angles between adjacent connected body segments. Furthermore, data can be computed over time to get angles over time and other properties such as jerk, acceleration, velocity and quantity of motion. The Max/MSP environment was proving to be too slow for cooking numbers from Xsens, so a dedicated C++ Real Time Motion Capture Toolbox software was developed. This software and the MoCap system were running on one computer and the rest on a second. They communicated through the OS C protocol, using a UDP network stream.

2.1.2 Finite-state Machine – script for a performance

Not unlike the script for a theater performance, this piece was composed using cues that determined:

1. Actions: What could be done (sonically)
2. Transition: What was required to transition to the next cue

This dichotomy, borrowed from the finite-state machine model, was extremely important when it came to designing the interaction (11). This model facilitated the creative flow and permitted quick changes in the program of the performance without the need of re-coding. The software was made on the Max/MSP programming environment and used the Jamoma framework for a more modular approach (12).

The performance was made up of sections that represented themes, and within each section there were various cues. Each cue enabled/disabled actions that communicated with the sound engine. Actions–MIDI messages to be performed at any given moment–harbored a specific mapping for the Xsens MVN data input to sound effects (e.g. left hand height to volume) and/or and automatic triggers (e.g. mute/unmute tracks).

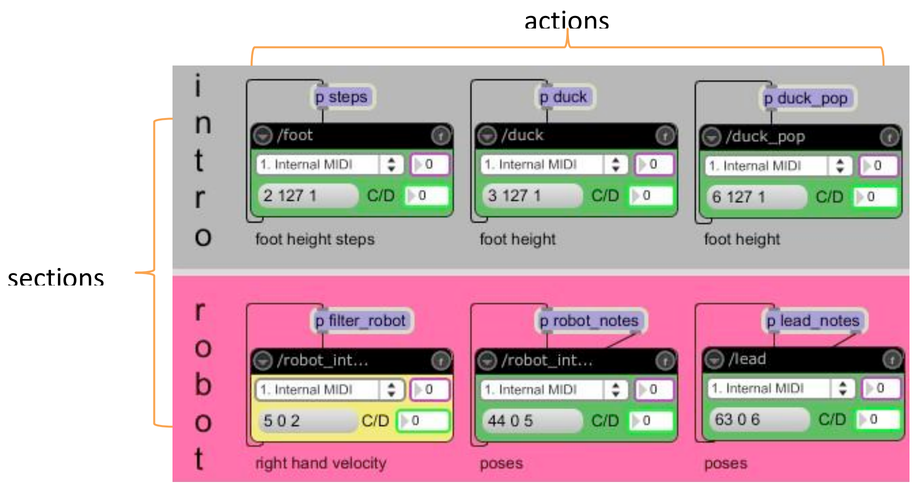

Figure 3 – Six actions

Figure 4 - cueManager in Jamoma

The cueManager was used to quickly create and manage cues while also defining a set of actions for each cue. Cue instructions were saved in a .txt format and could be updated on the fly (12).

Each cue had specific conditions that if met, would transition from the current cue to the next pre-determined cue from the cueManager. These conditions could be changed by manipulating the algorithm inside the transition patch. Cues could be quickly selected from a list that was updated in real time. The structure and planning for the performance was written in this patch.

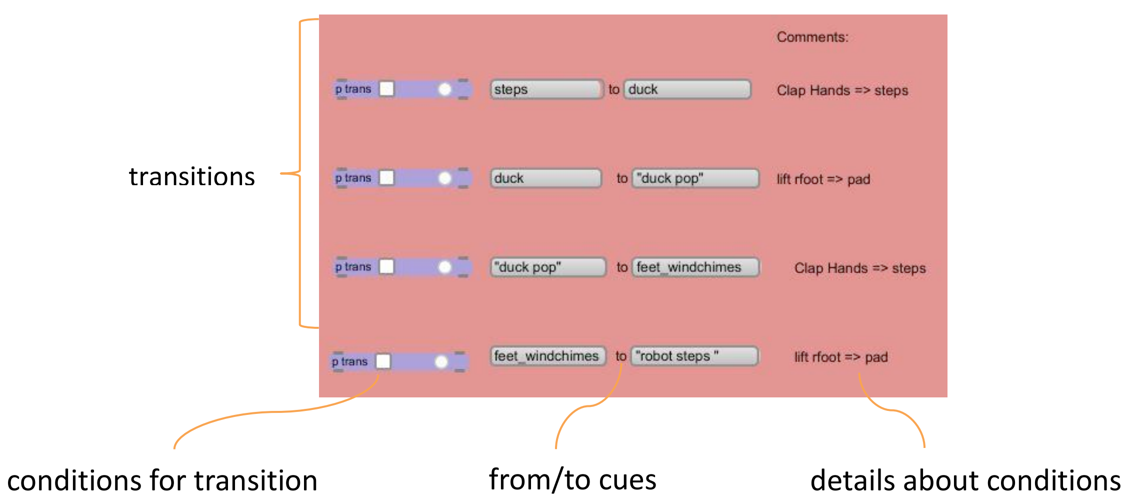

Figure 5 – Transitions between cues

2.1.3 Ableton Live 8 & Reason 4 – designing sound and composing music

Sound for the piece encompassed music compositions and sound effects by me. The audio engines used were Ableton Live 8 and Reason 4 due to their easy MIDI mapping and robust live performance capabilities. These audio software had the task of following instructions (i.e. MIDI messages) from Max. Instructions could be notes, continuous control values (e.g. to alter the filter frequencies) or triggers (e.g. audio clips). These messages were sent through LoopBe30, a software that made it possible to exchange MIDI between programs. MIDI notes were sent to Reason 4 synthesizers and samplers, and Ableton Live 8 to start and stop the sound clips. Control values were used to turn on/off tracks in Ableton Live 8 and manipulate faders. In order to provide a solid and reliable performance, the sonic piece was through-composed, i.e. no cues or sections were repeated. This led to a more organized performance not unlike traditional theater.

During the experiment, values relative to body (e.g. hand height from feet) were found to be more useful in contrast to values relative to geographic markers. Body poses, thresholds, and continuous data were the three types of data used to trigger and manipulate sound.

Body poses were described by the following Xsens parameters:

Left hand height | Left arm angle | Right hand height | Right arm angle

Whenever these relative values matched a body pose description in the Real Time Motion Capture Toolbox, it got sent to Max/MSP. By approximation, the software was always matching the current pose with stored ones. Thus we constantly got stored body pose identifiers that best described the current body pose of the subject. These poses, 8 in total, were used to trigger notes and transition between sections in the song. For example, the user in a T shaped pose (erect posture, arms stretched out to the sides) would send a 6 to Max/MSP which in turn sends a note value to the Reason Thor synthesizer.

Thresholds were sometimes used to transition between cues in the piece. For example, to start the performance, all I had to do was clap my hands, which translated into the distance between both hands being below a threshold of 10cm. In another cue, whenever one foot rose above 20cm, in relation to the other foot, the current cue would transition to the next.

Continuous values manipulated various sound effects in Reason 4. For example, having the frequency of a low pass filter on a synthesizer lead mapped to the left hand height, thus allowing higher frequencies in as my hand raised and vice versa.

2.2 Interactive Dance Club

The system comprised of on-site installations and online services that enabled the user to either directly or indirectly influence the music in the dance club.

On-site installations:

Figure 6 - Wii-Make-Music

Figure 7 - User-driven Vjing

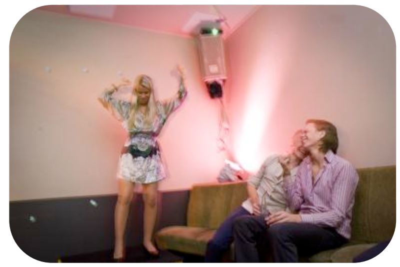

Figure 8 – iDance (13)

Online services:

Figure 9 - Ustream

Figure 10 – Grooveshark

Figure 11 - Twitter

2.2.1 On-site Installations

Wii-make-music consisted of two Wii remotes with colorful ribbons attached that made music when handled. The Wiis were connected by Osculator to a Mac, but the data was analyzed and the sound was produced on a PC. Using the devices' acceleration, one of the Wiis would trigger notes while the other would control a band pass filter applied on chords. To decide what notes and chords to play, audio from the main mixer streamed into a music information retrieval (MIR) patch inside Max/MSP that created a probability table of notes. This MIR patch combined the segment~, zsa.freqpeak and analyzer~ algorithms (14) (15) into a robust all-purpose tool. The resulting audio came out of a localized speaker on one corner of the dance floor.

User-driven Vjing consisted of a camera pointing toward the bar counter—where people danced most often—and a projection on a wall. The camera's video feed was being processed through 6 alternating visual effects in real-time, and then sent to the projector. The camera used was a Unibrain Fire-i without infra-red lenses. Most of the effects used silhouette extraction and frame difference, to be precise, the pixel difference from consecutive frames.

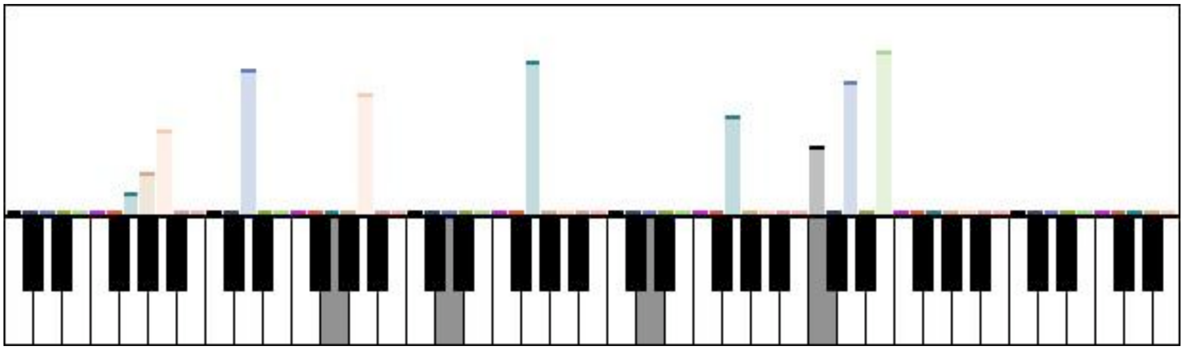

Figure 12 - Probability Table in the MIR patch

iDance was a small one-person podium where anyone could use their movements to trigger notes. This was done using an Optitrack infra-red (IR) camera taped to the roof and positioned to film from the users' hips to a meter above his/her head. This was to avoid interference from others in front of the user, and to be able to catch the arms if stretched above the head. The most robust algorithm for generating notes was quantity of motion, which was calculated by transforming the video input into a binary image and then calculating the frame difference. Therefore whenever the user would do abrupt movements the speaker would play a note. Notes to play were decided using the same probability table in the MIR patch mentioned in wii-make-music. The speaker and the podium were placed on the opposite corner of the dance floor, away from the wii-make-music system. For visual appeal and in order to entice more participation, a tv screen across the room would display the video input. Developed in Max/MSP/Jitter but unused, was an automatic background subtraction system meant for this iDance TV stream. Behind the podium, on the wall, there were 9 reflective markers. Whenever someone was in the podium, they would cover at least one of the markers. If all 9 markers were present for a few seconds, that was a sign that no one was on the podium and the system would take a picture of the background, and later mix it with the user's silhouette.

2.2.2 Online Services

For the benefit of those online, real-time video recording of the dance floor was streaming on ustream.com as well as regular twitter updates. There was on average 20 active viewers at any given time and a total of 174 for the second performance night. The link for the video stream was posted on the event Facebook page as well as the author's Website.

The music playing during the events was based on suggestions from the public. Songs were streamed from youtube.com and grooveshark.com. The latter had a playlist entitled "Gabler" and the credentials to log in and change the play list were available on a facebook.com group, my personal website, and couchsurfing.com (an online community of travelers). It was asked that songs be added, not removed, and that these belong to the "house/dance" genre. The present author acted as a mediator, and streamed songs in real-time from the before mentioned sources. Songs could also be requested on-demand during the night.

3. Results

The main results were obtained in four areas: 1) Communication; 2) algorithms; 3) sound and motion; and 4) dissemination.

Communication

- OSC (Open Sound Control) support for the Xsens MVN Suit system. Making it easier to connect the Xsens suit to different sound applications;
- Latency minimization, to distribute the computational load;
- New Jamoma module that mediates (MIDI) between the audio engines and the cueManager.

Algorithms

- Pose recognition so that the performer can use body poses as triggers for different actions. The algorithm also provided continuous information (proximity to a pose), which was used for continuous controls, e.g. transition between different sounds;
- Real Time Motion Capture Toolbox;
- Finite-state machine framework for a robust and flexible performance plan;
- Music Information Retrieval (MIR) Max/MSP patch suitable for various musical applications.

Sound and body motion

- 15 min composition exploring many of the possible relationships between sound and body motion;
- Tried and true motion-to-sound mappings suitable for dance clubs.

Dissemination

- Over 300 people attended 3 events in Oslo, Norway;
- A video of the first "Dance Jokey" performance on the Xsens website with over 370 views (last count October 2010) (16).
- News coverage of the first Interactive Dance Club held at Gabler (17)
- 15 Website posts related to STSM with more than 1000 views during the mission's two months (18),
- 150 unique viewers on the Ustream portal with a total of 51 hours and 7 minutes of viewer hours.

4. Discussion and Conclusions

The most challenging part for these projects consisted in implementing and developing technologies that could be easily transported and replicated from one performance to the next. These needed to adapt to changes from both the performer's artistic goals (be him using the Xsens suit or the installation) and the venue's space.

The MoCap systems were robust and worked without interruption throughout the events. Most users could hear with precision the correlation between sound and movement. However, uninformed users that didn't know that their movements were triggering sounds would find the sounds coming from the speaker puzzling. Thus, previous knowledge about the performance's goals and expectations can greatly increase the quality of user experience and participation. The performers and the venue managers need to make sure that online and offline publications can provide adequate information for the general public. The same goes for on-site signs and posters advertising and providing insight regarding the nature of the interactive systems.

Dissemination results suggest that interactive events may become an attractive entertainment alternative in dance clubs. Furthermore, it seems that innovative ways of using new technologies are steadily overcoming barriers for adoption of interactive music and pushing the boundaries of participation in dance clubs.

The MIDI protocol unfortunately falls short when handling large amounts of information, which is the case with MoCap systems. So other protocols need to be implemented, a promising protocol being the Open Sound Control (OSC). Up to now I have mostly spent time on technical challenges. The systems have barely scratched the surface of the artistic potential. Beyond exploring more advance motion algorithms and sophisticated sound manipulation, I want to mature its practice. Interestingly, at this stage it is not MoCap that warrants further research but the body-to-sound mapping mechanisms and interfaces; namely the leap from sonic interaction programing to exploration.

Acknowledgments

This publication was supported by the COST Action IC0601. Materials were generously supplied by the FourMs lab, Oslo, under the supervision of Alexander Refsum Jensenius. I want thank Carlos Guedes for support, Kristian Nymoen for developing some of the motion capture algorithms, and Ståle Andreas Van Dorp Skogstad whose drive and vision helped steer the Xsens performance.

Works Cited

1. **O'Reilly, Tim.** What Is Web 2.0. *O'Reilly Network.* [Online] October 2010. http://oreilly.com/web2/archive/what-is-web-20.html.

2. **Barndenburg, Karlheinz.** The Big Question. *Wired.* 2010. June.

3. *Body Movement In Music Information Retrieval.* **Godøy, Rolf Inge and Jensenius, Alexander Refsum.** s.l. : ISMIR, 2009.

4. *Social Active Listening and Making of Expressive Music: The Interactive Piece The Bow is bent and drawn.* **Camurri, A., et al., et al.** s.l. : DIMEA, 2008.

5. *Using IR Optical Marker Based Motion Capture for Exploring Musical Interaction.* **Skogstad, S. , Jensenius, A. R. and Nymoen, K.** Sydney : s.n., 2010. NIME. pp. 407-410.

6. *Wikipedia.* [Online] October 2010. http://en.wikipedia.org/wiki/motion capture.

7. *Toward real-time multimodal processing.* **Camurri, A.** s.l. : AISB, 2004.

8. *Xsens.* [Online] October 2010. http://www.xsens.com/.

9. **Wechsler, Robert.** [Online] Palindrome, October 2010. http://www.palindrome.de/.

10. **Skogstad, Ståle Andreas Van Dorp.** *The Use of Full Body Inertial Sensor Based.* Oslo : s.n., 2010.

11. Wikipedia. [Online] October 2010. http://en.wikipedia.org/wiki/Finite-state_machine.

12. *Jamoma: A modular standard for structuring patches in Max.* **Place, T. and Lossius, T.** New Orleans, USA : International Computer Music Association, 2006. In Proceedings of the International Computer Music Conference 2006.

13. **Nilsen, Igor R.** *Bevegelsesdj-en.* Universitas.no, Oslo : 2010.

14. **Jehan, Tristan.** *analyzer~, segment~.* 2001.

15. **Mal, Mikhail and Jourdan, Emmanuel.** *zsa.freqpeak.* Ircam, 2009.

16. MVN Biomech. *Xsens.* [Online] October 2010. http://www.xsens.com/en/mvn-biomech.

17. kultur. *Universitas.* [Online] September 2010. http://universitas.no/kultur/55159/bevegelsesdj-en/.

18. **Quay, Yago de.** [Online] October 2010. www.yagodequay.com.