Developing the Dance Jockey System for Musical Interaction with the Xsens MVN Suit

Ståle A. Skogstad and Kristian Nymoen
University of Oslo, Department of Informatics
{savskogs,krisny}@ifi.uio.no

Yago de Quay
University of Porto, Faculty of Engineering
yagodequay@gmail.com

Alexander Refsum Jensenius
University of Oslo, Department of Musicology
a.r.jensenius@imv.uio.no

ABSTRACT

In this paper we present the *Dance Jockey System*, a system developed for using a full body inertial motion capture suit (Xsens MVN) in music/dance performances. We present different strategies for extracting relevant postures and actions from the continuous data, and how these postures and actions can be used to control sonic and musical features. The system has been used in several public performances, and we believe it has great potential for further exploration. However, to overcome the current practical and technical challenges when working with the system, it is important to further refine tools and software in order to facilitate making of new performance pieces.

1. INTRODUCTION

The Dance Jockey system is based on the Xsens MVN suit, a commercially available *full body* motion capture system. The suit consists of 17 inertial sensors that are attached to a pre-defined set of points on the human body. Each sensor consists of an accelerometer, a gyroscope, and a magnetometer. The raw data streams from these sensors are combined in the Xsens MVN system to produce an estimation of how the body moves [9].

In previous research we have shown that the Xsens MVN system is well suited for exploring full body musical interaction [9, 10]. The system offers robust motion tracking of the body, which is important in live performance settings. In [9] we presented the Open Sound Control implementation and the technical experience of using the Xsens MVN system. In this paper we will outline in more detail about how we used the Xsens MVN suit to control sonic and musical features in the *Dance Jockey* project (Figure 1).

The motivation for the Dance Jockey project came from our wish of using the full body for musical interaction. As is often commented on, performing with computers allows for many new and exciting sonic possibilities, but many times with a weak or missing connection between the actions of the performer and the output sound [1]. To overcome this problem of missing or unnatural action-sound couplings [6], we are trying to develop pieces in which properties of the output sound match properties of the performed actions. With Xsens MVN motion capture (MoCap) system we are able to measure, with some limitation, the physical properties of our bodies' actions. It should therefore be possible to use this data to create physical relationships between actions and sounds. The challenge, however, is to extract relevant features from the continuous motion capture data stream and turn these features into meaningful sound.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
NIME'12, May 21 – 23, 2012, University of Michigan, Ann Arbor.
Copyright remains with the author(s).

Figure 1: A Dance Jockey performance at Mostra UP in Porto, Portugal. Note the orange sensors on different body parts and the two wireless transmitters on the back of the performer.

The name Dance Jockey is a word play on the well-known term Disc Jockey, or DJ. With this name we wanted to reflect that instead of using discs to perform music, we were using dance or full body motion as the basis for the performance. The name is also a reference to how we may think of the performer more as a DJ/turntablist than a musician: the performer does not play an instrument with direct control of all sonic/musical features, he is more triggering and influencing various types of sonic material through his body.

The developed Dance Jockey System has been used in several public performances over the last years, many of which are documented on our project web page.[1] This paper will mainly focus on the system itself, and we will therefore not present and discuss the performances.

We will start by presenting the main structure of the Dance Jockey System, followed by an overview of different feature extraction methods that have been developed, and how they have been used to control sonic and musical features.

2. THE DANCE JOCKEY SYSTEM

The system on which we have based our Dance Jockey project can be divided into four main parts, as illustrated in Figure 2. Let us briefly look at the concept of sound excitation before presenting the features used to extract control signals.

[1] `http://www.fourms.uio.no/projects/dancejockey/`

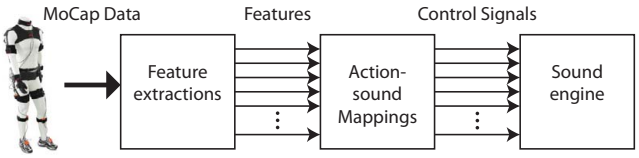

Figure 2: The dataflow of our Dance Jockey system

2.1 Sound Excitation

Most acoustic instruments are controlled with sound-producing actions that can be further broken into *excitation* and *modification* actions [7]. We can further distinguish between two types of excitations and modifications: discrete (e.g. triggering a sound object), or continuous (e.g. bowing a string instrument). This terminology can be seen as similar to what Dobrian identifies as control signals: *triggers* and *continuous streams of discrete data* [3]. These control signals should also be sufficient to control other musical features like tempo, skipping to the next section of the performance, changing synthesizer settings etc. Accordingly, we want to use the Xsens MVN data both for *continuous control* and to extract *trigger signals*.

2.2 Features Used for Extracting Control Signals

The Xsens MVN system outputs data about body motion by expressing body postures sampled at a rate of up to 120Hz. The postures are modeled by 23 body segments interconnected with 22 joints. Each posture sample consist of the *position* and the *orientation* of these segments. In addition, we get each segments' positional and orientational *velocity*, and positional and orientational *acceleration*. (The latter data are of relatively good quality as documented in [9].) All data is given in some *global* coordinate system, e.g. the stage.

There were three main properties we looked for when searching for suitable features from the above data; the features should be (1) robust and usable as consistent control data, (2) usable as visual cues for the audience, and (3) user-friendly for the artist. The features are difficult to evaluate without considering how they are mapped to musical parameters. It is therefore important to include the typical use of the features in the following subsections. We have not tried to make a complete list of all available features; instead, we will present those that we found useful. The features are summarized in Table 1 and several examples are illustrated in Figure 3.

2.2.1 Position data

We could, in theory, use the segments' global positions for both continuous control and extracting triggers by placing virtual positional thresholds on the stage (Figure 3e). But, we did not use the global position directly since the Xsens MVN *horizontal* position data exhibits drift, as documented in [9]. The vertical position, however, is much more consistent and could therefore be used directly as a feature. The latter can also be seen as a global feature since, for example, 1 meter above floor level will stay the same in all parts of the stage (Figure 3a).

The possibility of using global positions for sound spatialization is interesting. However, using global horizontal position for other types of sound excitation is somewhat problematic. We wanted actions in one area of the stage to result in the same output in other areas of the stage. In order to achieve this, we transformed global positions to the local coordinate system of the performer (pelvis). A specified action would then result in the same output in all areas of the stage, regardless of the orientation or position of the performer. This technique is also immune to the Xsens positional drift problem to a large extent. We used this approach when placing virtual "wind chimes" around the performer, who was able to trigger chimes by touching these virtual positions without worrying about standing in the correct position on the stage (Figure 3f).

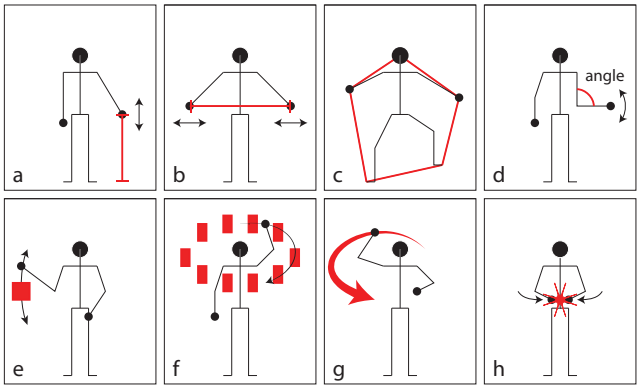

Figure 3: Illustration of some of the different features we have used: (a) vertical height of a hand, (b) distance between hands, (c) spanned distance between main body limbs, (d) elbow angle, (e) virtual trigger area, activated when the hand passes through this area, (f) virtual trigger areas that are always relative to the performer (g) absolute speed of hand, and (h) thresholding acceleration to recognize a hand clap.

2.2.2 Velocity - Continuous Excitation

We found the positional velocity of body limbs, especially the absolute velocity, i.e. the magnitude of velocity in all 3 dimensions, to be especially useful for continuous excitations (follows what Hunt et. al. discovered in [5]). This can also be mapped in an intuitive way with the performer's physical effort: the faster/larger the movement, the louder the sound. A benefit of using absolute velocity is its global nature: it is based on total velocity of the moving limb and is independent of the direction or location of the motion. We used this feature mostly for continuous control, for instance controlling amplitude or filters (Figure 3g).

2.2.3 Acceleration - Triggers

We found *thresholding* acceleration values to be especially suitable for extracting trigger signals, which is also mentioned by Bevilacqua et. al. in [2]. For example, the performer was able to trigger sound samples via abrupt rotations of his hand by thresholding the *rotational* acceleration data. We also used the performer's hip rotations to trigger samples. In this way we were able to synchronize sounds with apparent dance actions.

One of the challenges of using acceleration for extracting triggers is that sudden motion in one part of the body often spread to other parts of the body. As a consequence, it was difficult to isolate different triggers from each other, e.g. separating a kick from a sudden hip movement when only thresholding the segments' acceleration values. We overcame this by specifying extra *conditions* for the different trigger algorithms that needed to be separated. For instance, to be able to safely trigger a hand clap we added the condition that the hands needed to be no more than 20 cm from each other (Figure 3h). In this way we were able

to avoid other abrupt hand movement resulting in "hand clap" triggers. In similar ways we can make appropriate conditions for other trigger algorithms, such that they only trigger by the specified body action. This is one of the benefits of using a full body MoCap system (compared to using single accelerometers).

2.2.4 *Quantity of motion (QoM)*

By summing up the speeds of different body limbs we can compute the performer's total *quantity of motion*. To save computational power, we can add up the speed of only a subset of the main limbs, like head, feet and hands. This gives similar results. We connected this feature to loudness and other effort-related associations in the sound output, and we believe it is an interesting higher-level motion feature. However, the performer found this feature to be difficult to consciously control (low repeatability), and we therefore found it as having only limited use for extracting control signals.

2.2.5 *Relative position between body segments*

The Xsens MVN system outputs data which is mapped to a human body model. We find this model to be quite consistent and stable and therefore an interesting source for extracting control signals. It does not suffer from optical occlusion like infra-red optical marker based motion capture systems or have other major noise sources [9]. We do however experience some limited drift between limbs, but if this drift is taken into account the relations between different body parts can in our experience be quite robust and useful. (This property also applies for subsections 2.2.6 and 2.2.7.)

As a simple example, we used the distance between the performer's hands to reflect a physical space that the performer could manipulate, which again was used to make a physical relationship with the output sound (Figure 3b). Another feature that we used was the spanned distance of the 5 main body extremities: head, hands and feet. We used this distance for continuous excitation and modification, and found it useful to excite sound in a visually dramatic way (Figure 3c).

2.2.6 *Orientations - Joint Angles*

We did not use the segments orientation data directly. Instead, we used them to calculate the angles between different segments to extract joint angles, e.g. elbows and knees (Figure 3d). We believe that joint angles are more useful features than using the global orientation of single body limbs, since they tell more about the body pose. These angles are also relative to the performer's body. We used them to continuously excite or modify sound(s), and thresholded them to extract trigger signals.

2.2.7 *Pose classifier*

We developed a simple recognition algorithm based on an idea that different body poses could control some aspects of the sounds, besides also being valuable visual cues for communicating with the audience. We picked out five key pose features: the two elbow angles, hand distance, and both hand heights. Together these features spanned a pose space in five dimensions. We then stored the corresponding features of a set of 9 poses (the one we wanted to use as "cues" or "control poses"). These poses then had a corresponding point in the pose space. Finally, we implemented a *Nearest Neighbor Classifier* [4] to classify poses to the one of the stored poses that was closest, see Figure 4 for an illustration.

An advantage of this classifier was the high recognition rate, which in practice was 100%. This made it useful for exciting important musical features like notes and chords. However, the performer had problems with timing the pose changes correctly. To overcome this we implemented a system where a metronome was responsible for triggering the pose changes. In this way the performer only needed to be in the right pose at the right time. We also implemented functionality that looked after certain sequences of poses, which we used to extract trigger signals. Additionally, we used the distance, or how close the current posture is to the stored poses, to continuously morph between different sounds or timbres.

For some of the poses the quality of the suit calibration [9] could, to some degree, affect the resulting classification. We used a maximum of 9 different stored poses at one time. Furthermore, the recognition rate would probably decrease if we increased the amount of used poses. However, with a well selected set of pose features, it should be possible to use an extensive set of poses.

Feature	Used to control
Vertical position	Extensively for cont. and cond.
Relative positions	Trigger samples and cont.
Velocity (mag)	For cont., good "effort" relationship
Acceleration	Trig. sounds and state changes
QoM	Difficult for the performer to use
Relative body pos.	For cont. excitation and modification
Joint angles	Mostly for cond., some cont.
Poses	Notes, chords and states triggers

Table 1: Summary of how we used the different extracted features. There are three main uses of features, (1) continuous excitation o modification (cont.), (2) thresholded for use as trigger signals (trig.) and (3) as conditions for other triggers (cond.).

3. CONTROLLING SOUND AND MUSICAL FEATURES

3.1 The sound engine

All the sounds for the performance were generated and manipulated in Ableton Live 8 via MIDI and Open Sound Control (OSC). Ableton Live 8 does not accept OSC messages, so a third-party extension called LiveOSC was used to handle OSC data. However, we experienced considerable latency with the OSC messages, so time-critical events like synth notes, sound clips, and effects manipulation, had to be operated via MIDI.

The performance was organized in *states*, each containing

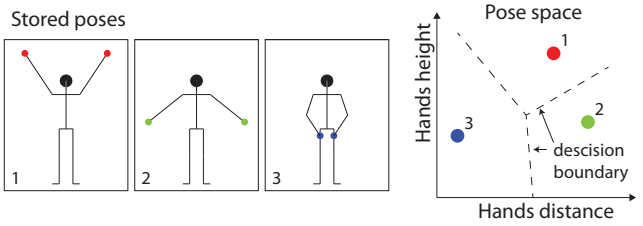

Figure 4: A simplified two-dimensional illustration of the pose classifier. The two pose features *hand height* and *hand distance* spans the pose space (right plot). Every pose will have a corresponding point in the pose space. We classify a pose to the one from the stored set that are closest.

sound effects, synths and other sound generating devices. As the performance progressed, we moved sequentially from one state to the next. A state could have various internal operations that affected Ableton Live 8, such as muting, raising volume, altering tempo, playing a clip, and so on. In the following Section we present how the states were controlled.

3.2 Transition between states

Our initial idea was to make a full-length performance piece in which all aspects of the performance were controlled solely by the Xsens MVN suit. For us, this meant that the performer needed to be, as much as possible, in full control of the whole performance. Therefore we needed to get rid of the invisible control center or the typical "guy behind the laptop"-setting [8].

At the same time we wanted the performance to have some varied content. We soon discovered that it was challenging to design a single instrument, or one synthesizer state, that would be interesting enough to listen to and watch for a whole performance. The performer needed to be able to change between different mappings. Our solution was to implement a so-called *finite-state machine*. This is a mathematical abstraction used to design sequential computer logic, which consists of a finite set of states, transitions between these states, and conditions for when the transitions should occur. To be able to go from one state to another the performer needed to perform predefined transition actions. Hence, the performer starts in one state, and when he/she feels that the part is finished, he/she can trigger the transition to the next state.

4. DISCUSSION

In the following we briefly discuss some of the thoughts we have had during the implementation of the Dance Jockey system.

4.1 Composing Dance Jockey

A challenge with composing and choreographing a performance for the Xsens MVN system was to decide to what degree the performance should be a musical concert controlled by a full body MoCap system, or a sonification of a dance piece [1]. We ended up with something in between. Designing action-sound mappings and making a performance around them turned the whole process into a creative one.

We also had to find a way to balance composition with improvisation. Some parts needed to be specified in detail, while others were left open. Specifically, parts featuring continuous sound excitation were particularly suitable for improvisation, and we found them to be especially important for establishing "expressive" action-sound relationships. The difference between a good and a bad concert was for us mostly determined by whether the performer was able to use these expressive parts to communicate with the audience.

4.2 The gap of execution

The process of composing and investigating action-sound mappings with the Xsens MVN suit takes a lot of time and energy. The suit is fairly quick to put on, but it is not comfortable to wear for several hours. It also involves many tiresome details, like calibration routines and changing batteries. While we were fully capable of performing concerts with the equipment, the time-consuming details and the obtrusiveness of the suit makes it tiresome to practice, compose and be creative.

Efficient tools are essential when attempting to compose and practice performances that employ full body MoCap technology. Through developing own tools and software while working with performance-related and technical aspects of the system, we have decreased the so-called gap of execution, or the gap between an idea - and its realization. Overcoming most of the technical challenges now enables us to focus on the artistic process. In this way our continued work on the Xsens performance will not be strangled by the many burdensome practicalities and obstacles that this technology and setup easily evokes.

4.3 Future research

We have seen a great number of possibilities that the Xsens MVN system offers for musical interaction, and feel that we have only touched the surface of these possibilities. Therefore, in the future we hope to get time and resources to make more thoroughly produced performances. We are currently working with more advanced action-sound mappings using physical models and granular synthesis, in order to build stronger perceptual connections between the MoCap data and sound output.

We also need to base our progression on more formal feedback. Up to now we have based our impressions on the feedback from audience members after concerts. This has not been sufficient to answer the questions we wanted to address, like: "Could you follow the action-sound mappings?" or "Did you enjoy the action-sound couplings or were they too evident/boring?" For that reason, in the future we would like to hand out questionnaires (likert scale, open ended questions, etc.) to get more formal feedback.

5. REFERENCES

[1] C. Bahn, T. Hahn, and D. Trueman. Physicality and feedback: a focus on the body in the performance of electronic music. In *Proc. of ICMC*, 2001.

[2] F. Bevilacqua, J. Ridenou, and D. Cuccia. 3D motion capture data: motion analysis and mapping to music. In *SIMS*, 2002.

[3] C. Dobrian. Aesthetic considerations in the use of 'virtual' music instruments. In *Proc. Workshop on Current Research Directions in Computer Music*, 2001.

[4] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience Publication, 2000.

[5] A. Hunt, M. M. Wanderley, and M. Paradis. The importance of parameter mapping in electronic instrument design. 2002.

[6] A. R. Jensenius. *ACTION - SOUND, Developing Methods and Tools to Study Music-Related Body Movement*. PhD thesis, University of Oslo, 2007.

[7] A. R. Jensenius, M. M. Wanderley, R. I. Godøy, and M. Leman. Musical gestures: concepts and methods in research. In R. I. Godøy and M. Leman, editors, *Musical Gestures: Sound, Movement, and Meaning*, pages 12–35. Routledge, New York, 2010.

[8] M. Kimura. Creative process and performance practice of interactive computer music: a performer's tale. *Org. Sound*, 8:289–296, December 2003.

[9] S. A. Skogstad, K. Nymoen, Y. de Quay, and A. R. Jensenius. OSC Implementation and Evaluation of the Xsens MVN suit. In *Proc of NIME*, Oslo, Norway, 2011.

[10] S. A. Skogstad, K. Nymoen, and M. Hovin. Comparing inertial and optical mocap technologies for synthesis control. In *Proc. SMC*, 2011.