Rhythmic Interaction with a Mobile Device

Vuokko Lantz[1]
Nokia Research Center
P.O. Box 407,
FIN-00045 Nokia Group,
Finland
Vuokko.Lantz@nokia.com

Roderick Murray-Smith[2,3]
[2]Department of Computing Science,
University of Glasgow, Scotland
[3]Hamilton Institute,
Nat. Univ. of Ireland Maynooth,
rod@dcs.gla.ac.uk

ABSTRACT

We describe a rhythmic interaction mechanism for mobile devices. A PocketPC with a three degree of freedom linear acceleration meter is used as the experimental platform for data acquisition. Dynamic Movement Primitives are used to learn the limit cycle behavior associated with the rhythmic gestures. We outline the open technical and user experience challenges in the development of usable rhythmic interfaces.

Author Keywords

rhythmic interaction, mobile devices, dynamic movement primitives, learning, input via inertial sensors

ACM Classification Keywords

H5.m. Miscellaneous, I.3.6 Methodology and Techniques

INTRODUCTION

Our everyday life is filled with rhythmic patterns of behavior, from walking, dancing, swimming, waving, rocking a baby to sleep, cursive handwriting and manual labor, such as sawing or hammering. Traditional working songs were often used to help a group of workers entrain to a comfortable pace of work on repetitive tasks, and ease the strain of labor. As you improve at a sport such as skiing, or rock climbing typically your initially clumsy, jerky moves improve as you find your 'natural rhythm' and benefit from greater efficiency of motion, as well as enjoyment. There has, however, been little work on the introduction of rhythm to user interfaces for computers. This was understandable in desktop scenarios, but with mobile interaction we have both needs and opportunities to introduce rhythmic interaction. Mobile devices are used while people walk – an understanding of rhythmic interaction will allow us to improve design, and work *with* the user's gait, rather than viewing it as a disturbance. Mobile devices have small displays, and are often used in noisy settings, but rhythmic feedback, whether audio, visual or haptic, is a very clear signal which can be used to alter walking speed, or entrain with and control the user's gesturing. An important benefit of rhythmic gestures is that it feels natural to repeat the gesture until the system recognizes the gesture, whereas with discrete gestures, an unrecognized gesture causes the user more frustration. There is a growing background of work on rhythm from psychology, [3, 2, 9], linking it with synchronization theory in physics [4]. Rhythmic models of cursive writing include [8]. Rhythmic interaction between a Phantom force-feedback device and a human, in a simulated dance, is demonstrated in [1].

NordiCHI '04, October 23-27, 2004 Tampere, Finland

DYNAMIC MOVEMENT PRIMITIVES

The modelling approach taken in this paper is to use the Dynamic Movement Primitives (DMP) of Schaal *et al.* [7, 6], which were developed to train robots to imitate human movements. The approach has already been proved in application in robotics, for imitation and skill learning. The advantages suggested in that domain, include spatial and temporal invariance (in terms of shifts or scalings), and its ability to cope well with disturbances, while still performing well in complex tasks. In this framework, gestures performed with a mobile device held in a moving hand can be represented with a second order dynamic equation system as follows:

$$\begin{aligned} \tau\dot{z} &= \alpha_z(\beta_z(g-y)-z)+f \\ \dot{y} &= z \end{aligned} \tag{1}$$

where $y(t)$, $z(t)$ and $\dot{z}(t)$ are the position, velocity and acceleration of the device at time t, respectively. Coefficients α_z and β_z reflect the physiological properties of the hand moving the device. Parameter τ is a time-scale factor which decides how quickly the goal state g (discrete gestures) or limit cycle (rhythmic gestures) $g(t)$ of the movement can be reached. Function $f(t)$ is the control of the movement. Without any control, $f = 0$, and appropriate parameter settings, $\alpha_z/4 \geq \beta_z$, this system will converge from its initial state to the goal state in an exponential, overdamped manner. More interesting movements and gestures can be obtained by having non-zero f. Learning a control function on the basis of an exemplar movement trajectory becomes a very difficult problem if f is an arbitrary nonlinear function. When assuming a DMP model for the control function, the supervised learning problem can be solved with the Receptive Field Weighted Regression (RFWR) algorithm [5]. For rhythmic movements, the DMP models are constructed in the following way:

$$\tau\dot{r} = \alpha_r(A-r) \tag{2}$$

$$\tau\dot{\phi} = 1 \tag{3}$$

$$\mathbf{v} = [r\sin(\phi), r\cos(\phi)]^T \tag{4}$$

$$\Psi_i = e^{-h_i(\mod(\phi,2\pi)-c_i)^2} \quad (5)$$

$$f = \frac{\sum_{i=1}^{N} \Psi_i \mathbf{w}_i^T \mathbf{v}}{\sum_{i=1}^{N} \Psi_i} \quad (6)$$

Here an oscillating system $\mathbf{v}$ excites the movement of the hand. It is defined by variables $r(t)$ and $\phi(t)$ and has an amplitude A and angular velocity $1/\tau$. The (z, y, r) system will be stable when $\alpha_r = \alpha_z/2$. Kernel functions Ψ_i are nonlinear basis functions which are activated around a certain phase c_i of the movement. The widths of their effective phase bands are defined by parameters h_i. Gestures are weighted sums of such basis functions. Gesture specific weights, or linear regression parameters of the model, $\mathbf{w}_i$ and kernel function parameters can be learned from recorded demonstrations with RFWR. The desired control function $f_{desired}$ is then calculated from a measured movement $\ddot{y}_{demo}(t)$:

$$f_{desired} = \tau\ddot{y}_{demo} + \alpha_z(\beta_z(r - y_{demo}) - \dot{y}_{demo}) \quad (7)$$

$$r(t) = A(1 - e^{-\alpha_r t/\tau}) \quad (8)$$

$$\phi(t) = t/\tau \quad (9)$$

when $r(0) = 0$ and $\phi(0) = 0$.

The RFWR algorithm is an incremental function approximation method which assumes that the functions are linear sums of nonlinear kernel functions. It learns the number of kernels, the kernel parameters and the linear regression parameters at the same time. In the beginning, there are no kernels at all. As the algorithm proceeds through the learning samples $(f_{desired}, r, \phi)$ one by one, new kernels are introduced if none of the existing kernels is activated beyond a certain threshold. Also, kernels are pruned to avoid overlapping: if two kernels are activated at the same time over a given threshold, the one with a wider effective phase band will be removed. New kernels are initially centered around the phase of the corresponding learning sample. The effective widths of the kernels are updated with a gradient decent based method whereas the linear regression parameters are learnt with Newton's method. Incremental learning of new gestures is enabled by learning the linear regression parameters independently for each kernel function.

Implementation

The DMP model of rhythmic gestures and the RFWR algorithm [10] used for learning the model parameters were implemented in Matlab. The code[1] was modified 1) to include a rhythmic DMP kernel function (5) with (r, ϕ) as its input, and 2) to have $\mathbf{v}$ as an input of the whole RFWR model. Input variables $r(t)$, $\phi(t)$, and $\mathbf{v}(t)$ were generated with equations (8), (9), (4), respectively, and by setting $A = 1$ and $\alpha_r = 1$. The time scale variable was set to $\tau = T/2\pi$, where T was the period of the rhythmic gesture estimated with an autocorrelation function. The other system parameters were set to $\alpha_z = 2\alpha_r = 2$ and $\beta_z = \alpha_z/4 = 0.5$. Velocity and position of the movement were estimated by integrating the measured acceleration numerically. As one period of a rhythmic gesture should have a zero-mean acceleration (ignoring the irrelevant gravitational component), the acceleration measurements were preprocessed by subtracting the mean in moving time window of width T. Estimated velocities and positions were preprocessed in a similar way. Initial velocities and positions were set to zero. The desired output values of RFWR model were then composed according to equation (7). The three measured acceleration components were treated independently from each other.

[1] http://homepages.inf.ed.ac.uk/svijayak/software/LWPR/index.html

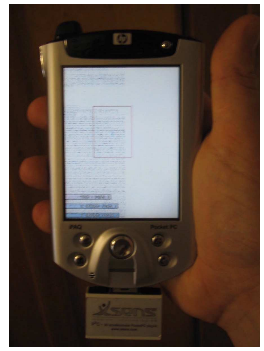

Figure 1. PocketPC with XSens accelerometer (weight=10.35g) attached at the base.

EXPERIMENT

To test whether it was possible to generate and model rhythmic behaviour reliably with a mobile device. We used an HP5550 PocketPC equipped with a 3-axis Xsens P^3C linear accelerometer attached to the serial port, as shown in Figure 1. It samples acceleration readings at 90Hz. In this study we used the PocketPC in a free-standing manner, logging data as gestures were performed. The data were later transferred to a PC for the modelling work. Our exploratory experiment was to log data from a single user performing two sets of 10 different gestures with a PocketPC, and to evaluate the suitability of rhythmic DMP models for representing these actions. All the gestures were performed by one subject while standing, holding the device in her right hand and moving the device in front of her mid-body. The following gestures were recorded, and are visualised in Table 1:

1. Device swung from left to right
2. Device swung back and forth
3. Device moved up and down
4. Anti-clockwise circles drawn with the device
5. Clockwise circles drawn with the device
6. Tapping side of device on left hand, *"ti-ti-ti ..."*
7. Tapping side of device on left hand, *"taa-ti-ti-taa-ti-ti ..."*
8. Anti-clockwise eights (from the top) drawn with device
9. Clockwise eights (from the top) drawn with device
10. Tilting the device from left to right

Usability comments

While a range of gestures might seem appropriate at first sight, there are certain types of rhythmic motions which are easier and more comfortable to perform, and which are more stable when the subject is perturbed. In case of gesture 10,

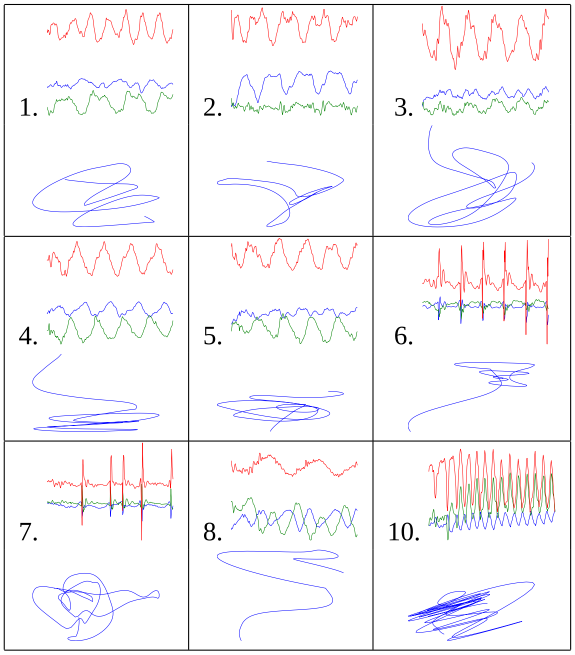

Table 1. Logged gesture data. Upper plots are 3.0 seconds of three-axis acceleration time-series, lower plots are 2-D projections via principal components of 3-D position trajectory, integrated from acceleration data.

it felt difficult to maintain exact beat as it is a small-scale movement – only the forearm is twisted – with a relatively high frequency. In repetition, gesture 10 started to feel uncomfortable and difficult to control. Thus, gesture 10 seems to be of bad design in respect to usability, and it is not well suited for the current implementation of rhythmic DMP modelling because of variability in the rhythm. Gestures 6 and 7 were easy and comfortable to perform, and maintaining the beat did not feel difficult to the subject. However, a closer examination of the measurement reveals that the phase is drifting, causing some difficulty to the modelling process, when fixed τ is used. There were no particular problems in performing the rest of the ten gestures.

Modelling results

Figure 2 illustrates how well the rhythmic DMP approach can model the control term f of the dynamical system (1). The black curve represents the training data, that is the values of $f_{desired}$ calculated with equation (7) and using measured accelerations and numerically integrated velocities and positions. The grey curve corresponds to the control values predicted by the rhythmic DMP model (6), the parameters of which have been estimated using the training data. The normalized mean square error (nMSE) for the prediction was 0.185 and the number of kernel functions was 57. It can be seen from the figure that the rhythmic DMP model can capture the general shape of the control function well but the minor details are lost. The predicted and measured control values are slightly out of phase, as our current implementation assumes the period of a gesture to be constant over time. The measured traces also typically show a fair amount of variation from cycle to cycle, which the DMP model does not replicate. Figure 3 shows the phase plane behaviors of the dynamical system induced by two non-zero controls. The black curve corresponds to the system's response to the measured control $f_{desired}$ whereas the grey curve is the system's response to the predicted control. Note that the DMP not only models the rhythmic behaviour, but also its onset. As would be expected, the discrepancy between the two system behaviors increases in time as the prediction errors of the control are accumulated in the integration of velocity and position. Results for the other components of acceleration and gestures, except gestures 6, 7 and 10, are essentially similar to these shown here for gesture 4 and its first acceleration component. Drifting phase and sharp peaks in acceleration data (very typical to gestures 6 and 7) caused the problems as our implementation of the DMP models assumes the period of the rhythmic movement to be constant over time.

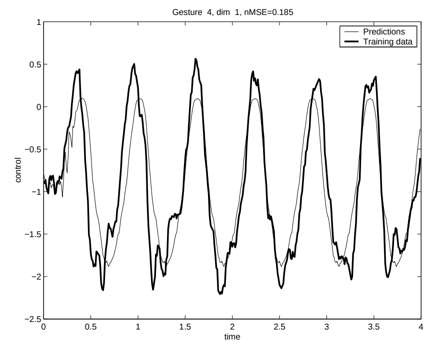

Figure 2. Comparison of the measured and predicted control for gesture 4 (1st acceleration component).

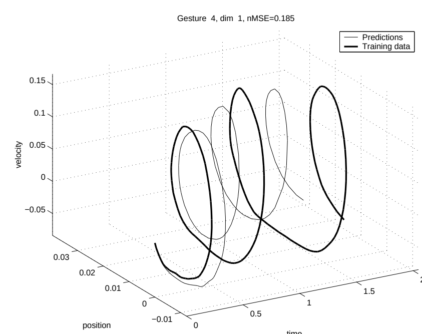

Figure 3. The measured and predicted phase plane (velocity, position, time) behavior for gesture 4,(1st acceleration component).

APPLICATIONS

Gesture recognition

The obvious application of the combination of accelerometers and rhythmic DMP modelling is gestural human-computer interaction. The kernel functions, or movement primitives, are learned from a database containing recordings of different kinds of gestures and thus they are gesture-independent. The linear regression parameters of the DMP model are learned separately for each gesture, using recordings of one type of gesture and are therefore gesture-dependent. Further gestures can be learned easily by including further submodels, and adjusting the linear regression parameters according to recorded gesture exemplars and introducing new kernel functions if necessary. Such an incremental learning will not degrade the recognition system's performance on the previously learned gestures. The gesture-dependent model parameters can be used as features in a gesture recognition task.

Model-driven feedback mechanisms

The gesture-dependent model parameters are also well suited to natural feedback generation, especially audio or vibration feedback. As in [11], for any classifier, different gestures can be sonified by linking each movement primitive to a different audio source and adjusting the intensity of the source proportionally with the similarity of the observed movement. This provides feedback about the system's interpretation of user intention, conditioned on the data seen so far. With DMP models, we can also link the parameters of the second-order dynamic equations of discrete or rhythmic DMPs to physically appropriate audio feedback, i.e., pick audio components relating to the mass, damping and spring terms for the discrete case, and a beat and amplitude for the rhythmic case.

Providing a sonification or vibrational feedback for gestures can serve several purposes. Firstly, the device can be used simply as an instrument, just for producing interesting sound effects or tactile patterns. Secondly, the feedback conveys the recognition engine's interpretation (and confidence) of the gesture being performed which can help the user actually perform the gesture more accurately. The problems we observed related to the variability in beat in gesture performance is an area where rhythmic audio or vibrotactile feedback could be invaluable in entraining with and stabilising user performance, and therefore improving recognition rates. Thirdly, the feedback can add a modality to the interaction between the device and the user making it more efficient, less dependent on the visual modality, and more engaging or realistic, in, for example, a gaming application.

CONCLUSIONS AND OUTLOOK

Our experimental system for acquisition and modelling of rhythmic gesture data with a PocketPC represents a starting point for rhythmic gesture research. We sampled a range of 10 test rhythmic gestures, and built models to represent these gestures using the Dynamic Movement Primitive approach. These models can be used to provide a spatiotemporal gesture classification, and can form the basis of new ways of generating vibrotactile and audio feedback, during classification. This initial investigation highlighted a number of challenges in this area.

Technical challenges

Currently the individual components of acceleration are modelled independently – modelling the interaction among the components will improve the model quality. The speed of the rhythm the gestures are performed with should be viewed as a time-varying variable, as it tends to drift in time both between users and within gestures. In order to make gesture-based interaction truly fluent, a way to segment gestures performed in an uninterrupted flow needs to be introduced. The finite state machine approach to combining movement primitives in robotics suggests close parallels between this approach and speech-recognition algorithms. The typical recognition accuracy and delay in response as the system recognizes examples from a useful range of gestures needs to be tested in a realistic and complete study.

User experience challenges

Careful design of the gestures is of paramount importance. Gestures should be easy for the human motor system to perform under a range of conditions, easy to remember, and easy to associate with their meaning or the functionality they arouse. In addition, they should be distinct enough not to be confused with each other by the user or recognition engine. Feedback generation applications lead to the problem of selecting audio or vibration sources which are perceptually distinctive, descriptive in respect to the parameters or gestures they represent, and produce pleasing combinations.

Acknowledgments

Both authors are grateful for the support of the EC TMR grant HPRN-CT-1999-00107. RM-S is grateful for support from EPSRC grant GR/R98105/01. SFI grant 00/PI.1/C067 and BRG project *Continuous Gestural Interaction with Mobile devices*.

REFERENCES

1. Gentry, S., and Murray-Smith, R. Haptic dancing: human performance at haptic decoding with a vocabulary. *IEEE International conference on Systems Man and Cybernetics*. IEEE (2003).
2. Kelso, J. A. S. *Dynamic Patterns: The Self-Organisation of Brain and Behavior*. MIT Press, 1995.
3. Kugler, P. N., and Turvey, M. T. *Information, Natural law, and the self-assembly of rhythmic movement*. Lawrence Erlbaum Associates, 1987.
4. Pikovsky, A., Rosenblum, M., and Kurths, J. *Synchronization: A universal concept in nonlinear sciences*. Cambridge University Press, 2001.
5. Schaal, S., and Atkeson, C. G. Receptive field weighted regression. Tech. Rep. TR-H-209, ATR Human Information Processing Laboratories, 1997.
6. Schaal, S., Ijspeert, A., and Billard, A. Computational approaches to motor learning by imitation. *Phil. Trans. Royal Society of London: Series B, Biological Sciences 358* (2003), 537–547.
7. Schaal, S., Peters, J., Nakanishi, J., and Ijspeert, A. Learning Movement Primitives. *International Symposium on Robotics Research (ISRR2003)*. Springer (2004).
8. Singer, Y., and Tishby, N. Dynamical encoding of cursive handwriting. *Biological Cybernetics 71* (1994), 3.
9. Thelen, E., and Smith, L. B. *A Dynamic Systems Approach to the Development of Cognition and Action*. MIT Press, 1994.
10. Vijaykumar, S., and Schaal, S. Locally Weighted Projection Regression: An $O(n)$ Algorithm for Incremental Real Time Learning in High Dimensional Space. *17th Inter. Conf. on Machine Learning (ICML2000), Stanford, California*. Morgan Kaufmann (2000), 1079–1086.
11. Williamson, J., and Murray-Smith, R. Granular synthesis for display of time-varying probability densities,. *International Workshop on Interactive Sonification (Human Interaction with Auditory Displays)*. Bielefeld University, Germany (2004), A. Hunt and T. Hermann, Eds.