# Motion Modeling for Expressive Interaction

## A Design Proposal using Bayesian Adaptive Systems

Baptiste Caramiaux
Department of Computing
Goldsmiths, University of London
New Cross London SE14 6NW, United Kingdom
b.caramiaux@gold.ac.uk

## ABSTRACT

While human-human or human-object interactions involve very rich, complex and nuanced gestures, gestures as they are captured for human-computer interaction remain relatively simplistic. Our approach is to consider the study of variation of motion input as a way of understanding expression and expressivity in human-computer interaction and in order to propose computational solutions for capturing and using these expressive variations. The paper reports an attempt at drawing the lines of design guidelines for modeling systems adapting to motion variations. We propose to illustrate them through two case studies: the first model is used to estimate temporal and geometrical motion variations while the second is used to track variations of motion dynamics. These case studies are illustrated in two applications.

## Categories and Subject Descriptors

H.5.2 [**User Interface**]: Input devices and strategies; H.5.2 [**User Interface**]: Interaction Styles; I.2.6 [**Artificial Intelligence**]: Learning—*induction*

## General Terms

Algorithms, Design

## Keywords

Motion, Expressivity, Interaction Design, Machine Learning, Adaptive Systems, Bayesian inference, Particle Filtering, Creative Applications

## 1. INTRODUCTION

Body movements and gestures are a powerful medium for non-verbal interaction. As such, they are increasingly being exploited in human-machine interaction for workplace, leisure, and creative interfaces. While human-human or human-object interactions involve very rich, complex and nuanced gestures, gestures as they are captured for human-computer interaction remain relatively simplistic (shapes, postures, simple movement primitives) on consumer devices such as touch screens, depth camera video controllers, and smartphone rotation sensors. There is a leap between body movements as used by humans in expression and as used by systems in interaction. On the other hand, in machine mediated gesture analysis, movement variability is often discarded in the name of consistency and generalisability of human-machine interaction. Our approach is the development of computational systems that are designed to capture variation of gesture as a way of understanding expression and expressivity in human-computer interaction and a way to embed it in new interactive systems.

To that extent, we would like to first introduce what we mean by motion expressivity. Motion expressivity is the notion of how a body movement is performed. For instance, in human-human communication, it is important to differentiate between the information content (*what* is communicated) and the expressive information (*how* it is communicated) [12]. In human-computer interaction, variation in gesture performance can exist across different users of an interactive system, or within a single user in multiple iterations recreating the same gesture primitive. The ways in which a gesture recognition system can be robust against, or sensitive to, these variations depends on the task at hand and the classification/adaptation algorithm used. We propose the term motion expressivity to describe meaningful variation in the execution of a gesture: *how a motion is performed?*. Motion expressivity is thus understood as a relative notion: meaningful variations of motion according to a reference.

The range of variations in the characteristics of a given gesture define its dimensions of expressivity. Multidimensional spaces for expressivity and expression of emotion have been previously studied in fields such as experimental psychology then applied in computer graphics animation, computer mediated communication and performing arts [16, 4, 12, 10].

On the other hand, there are very few examples of prior works using motion expressivity, defined as its meaningful variations, in an interactive context. The task is challenging. What are the variations? How to represent them? These are the first challenges that a designer has to face in order to build the interaction based on the motion's expressive content. Second, once defined and represented, variations are dynamic and, by definition, always changing. A second set of challenges is the development of robust and fast


*MOCO'14,* June 16-17 2014, Paris, France.

http://dx.doi.org/10.1145/2617995.2618009.

methods for estimating and tracking those variations. Finally, variation's definition, representation and tracking are context-dependent. In this paper, our context of application embraces creative and interactive applications that engage the user in an exploration of digital media such as visuals and sounds.

Thence, this paper is an attempt to extract design guidelines for creating computational models that adapt to variations and that allow for the use of these variations in creative interactive settings.

The paper is organized as follows. In the next section we will report related works on computational adaptive systems for motion recognition and tracking. Then we will present our design guidelines that drive the modeling of the proposed adaptive systems (Section 3). Two models for adaptation will then be presented, illustrating the design guidelines, in Section 4. The first model adapts to temporal and geometric gesture variations while the second adapts to physical regimes governed by a physical dynamical model. Both are using particle filtering for tracking variations. These adaptive systems will be illustrated in two applications (Section 5). And, finally, we will conlude in Section 6 and propose future directions.

# 2. RELATED WORK

In this section we review prior works on the design of computational systems that adapt to gestural input in an interaction context. We aim at refining typical applications where adaptation to gestural inputs is used, and consider how adaptation is handled and what are the typical models involved.

In motion computation, adaptive systems have been widely used for user-dependent gesture recognition. An adaptation process allows for specializing a set of free parameters on-the-fly in response to the user input and other constraints of the model. Wilson et al. [17] proposed a Parametric version of the well-known Hidden Markov Models (HMM). A global parameter is used in order to take into account spatial variations of the input gestures. Later Licsar et al. [11] developed a vision-based hand gesture recognition where users interact with a projected image by hand gestures, realizing an augmented reality tool in a multi-user environment. Finally, Gillian et al. [9] developed an adaptive gesture recognizer for semiotic musical gestures (communicative gestures exchanged by musicians while performing) based on a Naive Bayesian Classifier. These methods were used in applications where the goal was to build a robust user-dependent gesture recognition system that could be deployed in various environments. The interaction paradigm relies on an *iconic* control of the digital content, meaning that the focus is put on *what* is communicating. When considering creative applications such as performance and digital arts, this paradigm must be extended to account for the dynamic nature of motion and include *how* the motion is performed.

In the context of gestures related to music, temporal structure of complex gestures has then to be modeled, often using machine learning techniques [6]. Wilson et al. [18] reports *Watch and Learn*, a vision-based system able to learn gestures online for interactive control, based on HMM. The authors evaluate the system in a conducting scenario. The user is listening to a beat and starts to follow it with her hand, mimicking a conductor's movement. As a consequence, the system learns and is able to follow changes in speed of the user's beat and to apply this change on a music excerpt. Bevilacqua et al. [3] propose a method, also based on HMM, that allows for realtime temporal alignment of a gesture onto a template. The model has been first developed for synchronization of the gesture onto a musical score. Subsequently, it has been used as continuous parameter for expressive musical interaction [2]. An extension proposed by Françoise et al. [8] takes into account switches between gestures drawing upon a structure of hierarchical HMM.

In the research field related to dance and technology, prior works also tried to define adaptive methods for capturing the qualities of body movements. Swaminathan et al. [14] propose a dynamic bayesian network for recognizing qualities and ambiguity between qualities. Visell et al. [15] propose a particle filtering based method for inferring non-linear motion dynamics. More recently, Fdili Alaoui et al. [1] propose the use of the motion dynamic, independent of the trajectory, as a characterization of movement qualities.

# 3. DESIGN AND MODELING

The previous section presented several computational models, adaptive, mostly based on machine learning, that have been used for motion-based interaction. These solutions drew upon constraints from their application in interactive context. As we mentioned previously, our context involves creative interactive applications that engage the user in an exploration of digital media such as visuals and sounds. In this section we report the design characteristics that stem from our applicative context and we propose a class of models that embodies these characteristics.

## 3.1 Design characteristics

In an attempt to develop design guidelines for computational models used in creative interactive settings, we gather here characteristics for adaptive models that have been extracted from previous model designs. These design characteristics address three points: *Time is a feature*; *Making sense of uncertainty*; *Allowing (almost) the arbitrary.*

- **Time is a feature**. Motion variations characterize *how* body movements are performed temporally. They are fundamentally dynamic. Variations can occur at one specific instant (*local* variation) or over a longer period of time (*global* variation). Time is a feature to take into account in the modeling and time dependency occurs at different levels. Previous works widely used HMM that took into account the motion temporal structure. Similarly, our models will be temporal models.
- **Making sense of uncertainty: variation is not variability**. User's motion is captured through sensors that always involve noise. The noise can be of different nature: due to the sensor itself, the transmission, the digital quantization, environmental condition, etc. On the other hand the noise can also come from the user, e.g. performing a gesture poorly. Here uncertainty comes from variability and a probabilistic approach could handle this uncertainty, either via learning methods or heuristics. On the other hand, a gesture can also be consciously performed "wrongly". In this case uncertainty comes from variations. In an attempt to handle variations and variability, our models will be probabilistic. Note that previous models

such as HMM are also probabilistic models, unfortunately such model often assimilate variations to variability. Making sense of uncertainty involves considering variations not as noise.

- **Allowing (almost) the arbitrary**. User's input motion can be captured via various types of sensors leading to various types of representation. Motion representation often evolves in a multidimensional space. On the other hand, motion variations are also represented in a multidimensional space, often of different dimensions. More importantly, the variations are the designer's choice and can be (almost) arbitrary. It is critical here to design models that can relate these two spaces in a simple and efficient way. Our models are state-space models. More precisely a state-space model allows for defining a set of features representing the variations as state space.

## 3.2 Class of models

Formally, one class of models that can address the previous design characteristics can be written as a general dynamical, discrete time, state-space system:

$$\begin{cases} \mathbf{x}_k = f_T(\mathbf{x}_{k-1}, \mathbf{v}_{k-1}) \\ \mathbf{z}_k = f_O(\mathbf{x}_k, \mathbf{w}_k; \mathbf{m}) \end{cases} \tag{1}$$

At discrete time $k$, the elements at play in the system can be explained as follows:

- $\mathbf{x}_k$ is a vector representing the *system state*. Concretely, state element, denoted $\mathbf{x}_k(i)$, is a feature of variation (e.g. speed);
- $f_{\mathrm{T}}$ is a (possibly non linear) function that governs the evolution of the system state (the dynamic), depending on $\mathbf{x}_{k-1}$ and an independent and identically distributed (i.i.d.) process noise sequence $\mathbf{v}_{k-1}$. Concretely, the dynamics govern the speed and accuracy of convergence of the variation estimation towards the actual motion variations;
- $f_{\mathrm{O}}$ is a (possibly non-linear) function that generates the *observations* $\mathbf{z}_k$, depending on the system state $\mathbf{x}_k$, an i.i.d. measurement noise sequence $\mathbf{w}_k$ and a model $\mathbf{m}$. The model $\mathbf{m}$ is the link between the variations $\mathbf{x}_k$ to the actual motion observations $\mathbf{z}_k$ as we will see below. Concretely, we will use a distance function between the predicted value $\hat{\mathbf{z}}_k$ computed from the state estimation $\mathbf{x}_k, \mathbf{w}_k, \mathbf{m}$ and the actual observation $\mathbf{z}_k$ that will give us how accurate is the estimation of variations.

From this modeling, it is clear that the challenge in realtime adaptation, drawing upon realtime estimation of gesture variations, is to infer at each time $k$ the state value $\mathbf{x}_k$. This can be done thanks to *inference* methods.

The methodology is as follows: based on the motion capture system and the designer's choices, define the features of variation $\mathbf{x}_k$; then define the model $\mathbf{m}$ that would link the motion capture data to the features of variation; finally define the inference method that could estimate the variations $\mathbf{x}_k$ at each time step.

# 4. CASE STUDIES

## 4.1 Model 1: temporal, geometric variations

Our first model is designed in order to take into account temporal gesture variations (slower–faster) and geometrical variations (smaller–bigger, tilt) according to a reference. The model relies on position-based motion capture to derive the notions of speed, scaling and orientation. A first phase consists in recording a template gesture that will be taken as a reference. Four variations are considered: the time progression of the performed gesture in the template (index of the temporal alignment), the relative speed, the relative size and orientation. Figure 1 illustrates the variations addressed by the model in the case of a 2-dimensional shape gesture.

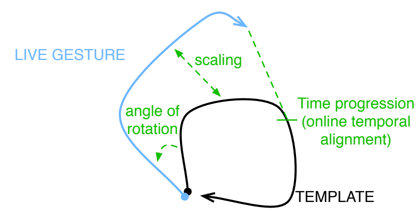


**Figure 1: Illustration of the variations tracked by Model 1.**

The model relies on Equation (1), where we defined states, dynamics, observations.

**States**. $\mathbf{x}_k$ is a vector where elements are the variations: $\mathbf{x}_k(1)$ is the time progression value; $\mathbf{x}_k(2)$ is the relative speed value; $\mathbf{x}_k(3)$ is the scaling coefficient; $\mathbf{x}_k(4)$ is the angle of rotation.

**Dynamics**. The transition function $f_T$ between a state at time $k$ and the state at time $k+1$ is linear: state at time $k$ equals the state at time $k-1$ plus a Gaussian noise centered at 0, see Figure 2. This means that when receiving a new gesture sample, the algorithm propagates the gesture variation values in a neighborhood whose size is governed by the variance of the gaussian. The variance is thus a parameter that can be tuned according to the application scenario.

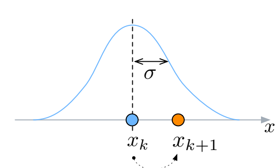


**Figure 2: Transition function is linear, relying on a Gaussian noise: $x_{k+1} = x_k + \mathcal{N}(0, \sigma)$.**

**Observation**. The observation function is assumed to be a Gaussian probability density function centered at the predicted observation $\hat{\mathbf{z}}_k$ with variance $\sigma$. The predicted observation $\hat{\mathbf{z}}_k$ is computed from the template's sample taken at index $\mathbf{x}_k(1)$, scaled by $\mathbf{x}_k(3)$ and rotated by an angle of $\mathbf{x}_k(4)$. The observed value $\mathbf{z}_k$ is then used to compute $p(\mathbf{z}_k|\mathbf{x}_k)$ that returns a value between 0 and 1 where 1 is reached when both the prediction and the observation are equal. It is used as a likelihood function to test the accuracy of the estimation[1].

[1] At the time of the writing, an article detailing the algorithm and its evaluation is in review: "*Adaptive gesture recognition*

### 4.2 Model 2: dynamic variations

Our second model is designed to take into account variations in dynamics of the gesture. Dynamics are understood as physical regimes of a dynamical physical model. The physical model considered is a second order linear dynamical system, also known as oscillatory system, given by:

$$\frac{d^2\mathbf{x}}{dt^2} + a\frac{d\mathbf{x}}{dt} + b\mathbf{x} = c \qquad (2)$$

If a movement can be modeled as such, the variations envisaged are the physical regimes of the system: oscillatory, oscillatory and damped, totally damped. Regimes are determined by the values of the coefficients $(a, b, c)$. Figure 3 illustrates the variations tracked.

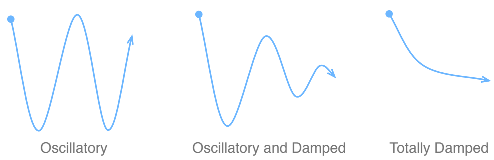


**Figure 3: Illustration of the variations from a 2nd order linear system tracked by Model 2.**

**States**. Here the state space comprises coefficients of the second order linear dynamical system as defined before: $\mathbf{x}_k(1) = a$, $\mathbf{x}_k(2) = b$, and $\mathbf{x}_k(3) = c$.

**Dynamics**. Dynamics between states at successive time is governed by the same transition function as for the model 1, i.e. a linear transition with Gaussian noise.

**Observation**. The prediction is based on how well the dynamical model fits the observation. Concretely, the observation is used to verify the equation (2) considering the estimated coefficients:

$$\frac{d^2\mathbf{z}_k}{dt^2} + \mathbf{x}_k(1)\frac{d\mathbf{z}_k}{dt} + \mathbf{x}_k(2)\mathbf{z}_k - \mathbf{x}_k(3) = 0$$

If the model fits well the incoming observation, the equation is satisfied.

### 4.3 Adaptation using particle filtering

Adaptation is the potential of a system to estimate in realtime given parameters in order to fit the model to the given input under given constraints. Back to our models, it means that the values $\mathbf{x}_k(i)$ must be estimated in order to ensure the best prediction value according to the observation. The adaptation process is a tracking problem.

For both models that we introduced above, we use a sampling method to perform the realtime tracking of the features of variation, namely particle filtering [13]. The idea behind sampling methods is to sample possible state values, i.e. possible variations values (according to a given probability distribution) and compute weights associated to each one of the values, leading to more probable values than others (and giving rise to a new distribution). Figure 4 schematizes the process: sampled values (circles) are weighted (blue color), a new estimate is computed from the most probable values (blue line), while another less probable possibility is discarded (dashed blue line). A particle is denoted $\mathbf{x}_k^i$, its weight $w_k^i$ and the estimated value at time $k$ is computed as: $\hat{\mathbf{x}}_k = \sum_{i=1}^{N_s} w_k^i \mathbf{x}_k^i$.

*with variation estimation for interactive systems*"; however an open-source c++ library (*Gesture Variation Follower*) is available online: `https://github.com/bcaramiaux/gvf`

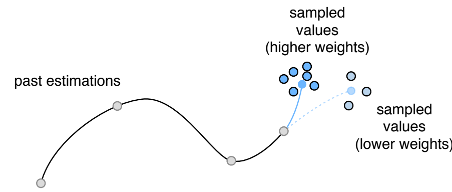


**Figure 4: Particle filtering: sampled values (circles) are weighted (blue color), a new estimate is computed from the most probable values (blue line), while another less probable possibility is discarded (dashed blue line).**

According to the model considered, $\hat{\mathbf{x}}_k$ has a different interpretation (either phase, speed, scale, orientation or coefficients of the physical model). Nonetheless, the vector value $\hat{\mathbf{x}}_k$ gives the value at a given time of the features of variations of the incoming motion. Particle filtering has advantages and drawbacks.

Among advantages, the method is **incremental** (the inference can be performed online and in realtime), handles **non-gaussianity** (the probability distribution estimated by the set of particles is non-gaussian which means, for instance, that it can model ambiguity), handles **non-linearity** (the model can be arbitrarily complex)

Among drawbacks, the method is **incremental** (it might make the algorithm converge slowly), can have a high **computational cost** (an accurate estimation is enhanced by a high number of particles which means a higher computation cost), has several **degrees of freedom** (several parameters have to be set manually, or via complex and computationally expensive training techniques).

## 5. APPLICATIONS

We now illustrate the two models within two distinct applications. The first application scenario uses gesture geometric variations to control continuously visual effects on an image. The second application scenario uses gesture physical regimes to control a virtual violin parameters.

### 5.1 Application 1: gesture geometric variation for the control of visual effects

**Scenario**. In this scenario, a user takes a picture with a digital device such as mobile phone, and wants to apply visual effects to it. A solution[2] is to navigate within menus and to select the effect to be applied. Intensity of the effect can eventually be modulated by manipulating a graphical command (e.g. slider). In our scenario, the user is able to use gestural input to select and control a visual effect in realtime. More precisely, the interaction design draws upon gesture recognition to select an effect and different interpretations of the user's gesture (i.e. gesture variations) to continuously manipulate the visual effect's intensity while the gesture is being performed. We call it: *gesture only* interaction. Such scenario is aimed to be embedded in modern mobile devices, constraining the input gestures as a 2-dimensional shapes performed on the device itself.

[2]Such as in Apple PhotoBooth application or Instagram

**Prototype**. In the implemented prototype, users' gestures are captured on the touchscreen of an iPad and sent using the OpenSoundControl[3] (OSC) protocol to the host computer. Then we used the Max/MSP Jitter environment to emulate the image processing software. Figure 5 illustrates the application set-up. User's gesture is analyzed in realtime by the model: recognition allows for selecting the visual effect and both variations, time progression within the gesture and scale, are tracked. Time progression is mapped onto one effect's continuous parameter (e.g. deformation) while the estimated scaling is mapped onto another parameter (e.g. saturation). This allows the user to select an effect and control its two continuous parameters simultaneously by performing variations of one gesture[4].

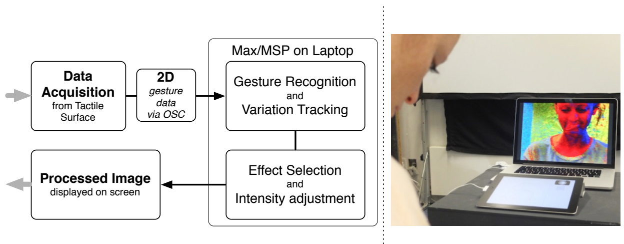


**Figure 5: Application 1 setting. An iPad is used to capture 2-dimensional gestures that are sent to a computer performing gesture variation tracking and image processing.**

**Observations**. In a previous article [5] we showed how users experienced this new interaction based on the gesture only interaction paradigm. In summary, such interaction has been rated more hedonic than the usual menu/slider interaction, namely more attractive and more expressive. However it has been found less deliberate. Here we would want to bring other aspects based on our observations during the experiment and our own use of the application.

A first observation was that the users did not attach so much importance to the accuracy of the variation estimation. We believe that it is partially due to the task: users were asked to explore and not to complete a precise task. Also, we believe that it is linked to the perception of the visual feedback: users could not quantitatively and precisely assess their own movement variation (how much the movement is deviating in size or speed) through the feedback. This means that we can relax the accuracy constraint in the estimation, eventually to the benefit of an other feature. This other feature could be the speed of convergence towards the good variation estimation, as users where slowing their movement to ensure a synchronicity between their intention and the feedback. Both estimation accuracy and speed of convergence can actually be controlled in the model by the variance in the transition function (see Section 4).

Finally, a very interesting aspect resides in the gesture itself. The gesture suffers from radical transformation as observed in Figure 6. The gesture performed live (right on the Figure) is indeed visually very different from the template (left on the Figure). It seems that the users were locally aware of the gesture and its variations while not planning the gesture globally. As a result, the exploration space offered by the interaction method is widened as clearly illustrated by the Figure 6.

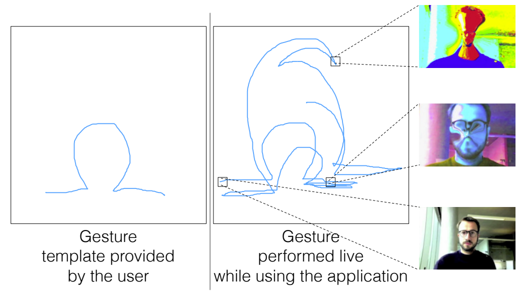


**Figure 6: Transformation of a gesture during the interaction. On the left is the template gesture provided by the user before using the application. On the right is the live gesture while using the application. Snapshots of the feedback during the performance are displayed on the right.**

## 5.2 Application 2: virtual violin

**Scenario**. In this scenario, a user is able to play a virtual violin (simulated on a computer) through the physical behavior of her arm. The virtual violin is governed by a physical model and the changes in motion behaviors will continuously drive changes in the violin's sonic behavior. In the scenario, the user must go beyond control. To that extent, the scenario involves more inherent motion captured from muscle activity, precisely the mechanical response to muscle activity.

**Prototype**. The prototype uses a mechanomyogram sensor that captures the mechanical response of muscle activation (called Xth-sense[5] [7]). The sensor consists of an arm band containing an electret condenser microphone where acoustic perturbations from muscle contraction are digitized as audio. The audio channel is sent to a computer and it is analyzed. The analysis uses a linear second order dynamical system as described previously. This dynamical model is fitted on the audio channel and the coefficients of the equation are estimated[6]. Note that the model is linear while muscle activity is non-linear but in this prototype we approximate with a simpler model where the coefficients are easily interpretable. The sound synthesis is performed using the violin simulator in Modalys[7] within Max/MSP. Figure 7 reports the application data flow and a snapshot.

**Observations**. Estimation accuracy has been found important here since the estimated coefficients are used to control a sound physical model where subtle variations in the control parameters can lead to critical sonic differences. However the speed of convergence of the estimation was found to be less critical since the user does not seem to expect an instantaneous reaction from the system, as for the previous application, thanks to the physical model that can be perceived as having inertia or hysteresis.

[3]http://opensoundcontrol.org/

[4]A video illustrating the application is available online `https://www.youtube.com/watch?v=Bqg_Zg5ompc`

[5]http://res.marcodonnarumma.com/projects/xth-sense/

[6]A video illustrating the application is available online `https://www.youtube.com/watch?v=-1z6tjOwQlO`

[7]http://forumnet.ircam.fr/product/modalys/

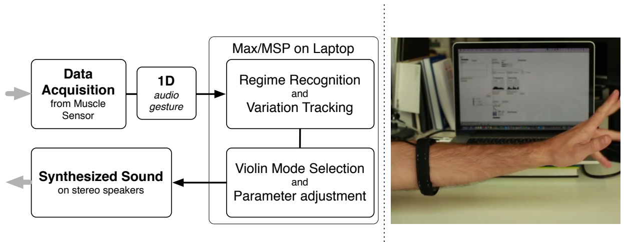


**Figure 7: Illustration of the second scenario: the arm band muscle sensor captures muscle mechanical activity that is analyzed in order to extract physical regimes, subsequently mapped to a violin synthesizer.**

Second, in this model, no training is required since all our hypothesis (the dynamical system itself) is defined beforehand and not learned from users' gestural inputs. Consequently, the model is rigid in the sense that only behaviors allowed by the dynamical systems can be spotted. On the other hand, the model can be directly used without requiring a preliminary phase that would not be understood by the user.

Finally a very interesting aspect here resides in the input: the motion. While the previous model considered gestures with clear starting and ending points, this model focus on motion. Motion here draws upon physics of movement and tries to exploit these physics into the analysis.

# 6. CONCLUSION AND PERSPECTIVES

In this paper we argued for the importance of considering expressive information in motion-based human-computer interaction. This expressive information is understood as *how* the gesture is performed. More precisely, we were interested in the meaningful variations between performances of the same gesture. We propose three design characteristics and a class of computational adaptive models that embodies these characteristics. Hence, we derived two case studies: the first allows for tracking temporal and geometric variations while the second tracks dynamic variations. Both rely on a particle filtering inference and have been implemented in real-world applications. From these applications, we extracted observations on accuracy and its trade-off with speed of convergence. In addition we reported considerations on the motion inputs. The first application involves gestures that are transformed offering a wider gesture design space for interaction. The second application involves motion where no starting and ending points are specified, enhancing exploration.

Future works will focus on the evaluation of the first model in a musical application (piano pedagogy and performance). The second model will be further developed in order to explore and evaluate the use of motion dynamics in interaction. Utlimately, design guidelines will be nourished and refined by these works.

# 7. REFERENCES

[1] S. F. Alaoui, B. Caramiaux, M. Serrano, and F. Bevilacqua. Movement qualities as interaction modality. In *Proceedings of DIS'12*, pages 761–769. ACM, 2012.

[2] F. Bevilacqua, N. Schnell, N. Rasamimanana, B. Zamborlin, and F. Guédy. Online gesture analysis and control of audio processing. In *Musical Robots and Interactive Multimodal Systems*, pages 127–142. Springer, 2011.

[3] F. Bevilacqua, B. Zamborlin, A. Sypniewski, N. Schnell, F. Guédy, and N. Rasamimanana. Continuous realtime gesture following and recognition. In *LNCS*, pages 73–84. Springer, 2010.

[4] A. Camurri, B. Mazzarino, M. Ricchetti, R. Timmers, and G. Volpe. Multimodal analysis of expressive gesture in music and dance performances. In *LNCS*, pages 20–39. Springer, 2004.

[5] B. Caramiaux, F. Bevilacqua, and A. Tanaka. Beyond recognition: using gesture variation for continuous interaction. In *CHI'13, extended abstracts*, pages 2109–2118. ACM, 2013.

[6] B. Caramiaux and A. Tanaka. Machine learning of musical gestures. *Proc. of NIME 2013*, 2013.

[7] M. Donnarumma. Xth sense: researching muscle sounds for an experimental paradigm of musical performance. In *Proceedings of the LAC*, 2011.

[8] J. Françoise, B. Caramiaux, F. Bevilacqua, et al. A hierarchical approach for the design of gesture-to-sound mappings. In *Proceedings of SMC'12*, pages 233–240, 2012.

[9] N. Gillian, R. B. Knapp, and S. O'Modhrain. An adaptive classification algorithm for semiotic musical gestures. In *Proceedings of SMC'11*, 2011.

[10] A. Kleinsmith and N. Bianchi-Berthouze. Affective body expression perception and recognition: a survey. *IEEE Transactions on Affective Computing*, 4(1):15–33, 2013.

[11] A. Licsár and T. Szirányi. User-adaptive hand gesture recognition system with interactive training. *Image and Vision Computing*, 23(12):1102–1114, 2005.

[12] C. Pelachaud. Studies on gesture expressivity for a virtual agent. *Speech Communication*, 51(7):630–639, 2009.

[13] M. Sanjeev Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Trans. on Signal Processing*, 50(2):174–188, 2002.

[14] D. Swaminathan, H. Thornburg, J. Mumford, S. Rajko, J. James, T. Ingalls, E. Campana, G. Qian, P. Sampath, and B. Peng. A dynamic bayesian approach to computational laban shape quality analysis. *Advances in HCI*, 2009:2, 2009.

[15] Y. Visell and J. Cooperstock. Enabling gestural interaction by means of tracking dynamical systems models and assistive feedback. In *IEEE International Conference on SMC*, pages 3373–3378. IEEE, 2007.

[16] H. G. Wallbott. Bodily expression of emotion. *European journal of social psychology*, 28(6):879–896, 1998.

[17] A. D. Wilson and A. F. Bobick. Parametric hidden markov models for gesture recognition. *IEEE Trans. on PAMI*, 21(9):884–900, 1999.

[18] A. D. Wilson and A. F. Bobick. Realtime online adaptive gesture recognition. In *Proceedings of ICPR'00*, volume 1, pages 270–275. IEEE, 2000.