15th Triennial World Congress, Barcelona, Spain

PRIMITIVES FOR HUMAN MOTION: A DYNAMICAL APPROACH

Domitilla Del Vecchio [*] **Richard M. Murray** [*] **Pietro Perona** [*,**]

[*] *Division of Engineering and Applied Science, California Institute of Technology, Pasadena, CA 91125*
[**] *Università di Padova, Italy*

Abstract: Using tools from dynamical systems theory and systems identification theory we develop the study of primitives for human motion which we refer to as *movemes*. We introduce basic definitions of *dynamical independence* of linear time-invariant dynamical systems (LTI) and *segmentability* of signals and we develop classification and segmentation algorithms for two dimensional motions. We test our ideas on data sampled from four human subjects who were engaged in a simple real-life activity including two movemes. Our experiments show that we are able to distinguish between the two movemes and recognize them even when they take place in an activity containing more than one moveme.

Keywords: classification, convex optimization, identification, computer experiments, data acquisition

1. INTRODUCTION

Building systems that can detect and recognize human actions and activities is an important goal of modern engineering. Applications range from human-machine interfaces, to security to entertainment. The first fundamental problem in achieving this goal is one of representation. Our point of view is that human activity should be decomposed into its building blocks which belong to an "alphabet" of elementary actions that the machine knows. We refer to these primitives of motion as *movemes*. This word first came up in the work by (Bregler and Malik, 1997). Their approach does not include an input and therefore is only applicable to periodic or stereotypical motions, such as walking or running where the motion is always the same. (Goncalves *et al.*, 1998) also proposed to divide human motion into elementary trajectories called movemes. They dealt with the problem in a phenomenological and non-causal way: each moveme was parameterized by goal and style parameters. We attempt here to define movemes in terms of causal dynamical systems; this way a moveme could be parameterized by a small set of dynamical parameters and by an input which drives the overall dynamics. Our aim is to build an "alphabet of movemes" which one can compose to represent and describe human motion similar to the way phonemes are used in speech. Two more problems we address are the ones of segmentation and classification: can a continuous trajectory of the human body be decomposed automatically into its component movemes?

We validate our ideas by analyzing the mouse trajectories generated by computer users as they "point-and-click" (we call this the reach moveme) and trace straight lines (we call this the draw moveme).

2. AXIOMATIC PERSPECTIVE ON MOVEMES

This section is concerned with the theoretical approach to the study of movemes: we give a few basic definitions and set up the requisite mathematical framework. Let $M = M(\theta)$ denote a LTI systems class parameterized by $\theta \in \mathbb{R}^p$ and let $\mathcal{U}$ denote a class of inputs. Let $y(t) = Y(M(\theta)|_{u,x_0})(t)$ denote the output of $M(\theta)$ once parameter $\theta \in \mathbb{R}^p$, input $u \in \mathcal{U}$ and initial condition $x_0 \in \mathbb{R}^n$ have been chosen. To semplify notation we will write $y(t) = Y(M(\theta)|_{u,x_0})$.

Definition 1. Let $M^R(\theta) = \{M(\theta)|\theta \in \mathcal{C}^R\}$ and $M^D(\theta) = \{M(\theta)|\theta \in \mathcal{C}^D\}$ denote two subsets of models in the class M with $\mathcal{C}^j \subset \mathbb{R}^p$, for $j = R, D$. The two sets $M^R(\theta)$ and $M^D(\theta)$ are said to be *dynamically independent* if:

(i) the class M and the class of inputs $\mathcal{U}$ are such that $Y(M(\theta_1)|_{u_1,x_0}) = Y(M(\theta_2)|_{u_2,x_0})$ iff $(\theta_1, u_1) = (\theta_2, u_2)$ for $u_1 \in \mathcal{U}$ and $u_2 \in \mathcal{U}$;
(ii) the sets $\mathcal{C}^R$ and $\mathcal{C}^D$ are non-empty, bounded and linearly separable.

A set $\mathcal{M} = \{M^1, ..., M^l\}$, where $M^i(\theta) = \{M(\theta)|\theta \in \mathcal{C}^i\}$ is said to be a set of *mutually dynamically independent* model sets if all the pairs $\{M^i, M^j\}$ are dynamically independent for $i \neq j \in [1, l]$.

The linear separability requirement for the sets $\mathcal{C}^R$ and $\mathcal{C}^D$ can be relaxed, in a more general framework, just to separability.
Each of the elements of the set $\mathcal{M}$ of mutually dynamically independent model sets is called a *moveme*. Let $M^j \in \mathcal{M}$ be a moveme. We let $y^j(t) = Y(M^j(\theta)|_{u,x_0}) = Y(M(\theta)|_{u,x_0})$ denote the *moveme output* for M^j once the parameters $\theta \in \mathcal{C}^j$, input $u \in \mathcal{U}$ and initial conditions $x_0 \in \mathbb{R}^n$ are determined. Given a signal $y(t)$, $t \in [t_0, T]$, let $s_1(t)$ and $s_2(t)$ be the two signals defined as

$$\begin{aligned} s_1(t) &= y(t),\ t \in [t_0, \tau] \\ s_2(t) &= y(t),\ t \in [\tau, T]\ , \end{aligned} \tag{1}$$

where $\tau \in (t_0, T]$. We let $(s_1(t), s_2(t))_\tau$ denote the *segmentation* of $y(t)$ at time τ.

Definition 2. A signal $y(t)$ is said to be *segmentable* if there exists $\tau^* \in (t_0, T)$ such that the segmentation at time τ^*, $(s_1(t), s_2(t))_{\tau^*}$, satisfies

$$\begin{aligned} s_1(t) &= Y(M(\theta_1)|_{u_1,x_{t_0}}),\ t \in [t_0, \tau^*] \\ s_2(t) &= Y(M(\theta_2)|_{u_2,x_{\tau^*}}),\ t \in [\tau^*, T] \end{aligned} \tag{2}$$

for some u_1, u_2, $, x_{t_0}$, x_{τ^*}, θ_1, θ_2 with $(\theta_1, u_1) \neq (\theta_2, u_2)$. The couple $(s_1(t), s_2(t))_{\tau^*}$ defined here is referred to as the *actual segmentation*.

Proposition 3. A moveme output $y^i(t) = Y(M^i(\theta^*)|_{u^*,x_{t_0}})$, $t \in [t_0, T]$, is not segmentable.
Proof. Let $(s_1(t), s_2(t))_\tau$ be the segmentation of $y^i(t)$ for any $\tau \in (t_0, T)$. Suppose for $t \in [t_0, T]$ $s_1(t) = Y(M(\theta_1)|_{u_1,x_{t_0}})$ and $s_2(t) = Y(M(\theta_2)|_{u_2,x_\tau})$, $t \in [\tau, T]$. Also $s_1(t) = Y(M^i(\theta^*)|_{u^*,x_{t_0}})$, $t \in [t_0, \tau]$ and $s_2(t) = Y(M^i(\theta^*)|_{u^*,x_\tau})$,$t \in [\tau, t_0]$. Therefore $Y(M(\theta_1)|_{u_1,x_0})$= $Y(M(\theta^*)|_{u^*,x_0})$, $Y(M(\theta_2)|_{u_2,x_\tau}) = Y(M(\theta^*)|_{u^*,x_\tau})$ which by (i) of Definition 1 imply $(\theta_1, u_1) = (\theta^*, u^*)$, $(\theta_2, u_2) = (\theta^*, u^*)$ which in turn imply $(\theta_1, u_1) = (\theta_2, u_2)$ that contradicts Definition 2. □

Proposition 4. If $y(t)$, $t \in [t_0, T]$, is segmentable, then the actual segmentation is unique.
Proof. Let $(s_1(t), s_2(t))_{\tau^*}$ as defined in (2) be the actual segmentation of $y(t)$. Suppose there is a $\tau < \tau^*$ such that $(\tilde{s}_1(t), \tilde{s}_2(t))_\tau$ is an actual segmentation, then since $\tau < \tau^*$ we have $\tilde{s}_1(t) = Y(M(\theta_1)|_{u_1,x_{t_0}})$, $t \in [t_0, \tau]$, and $\tilde{s}_2^a(t) = Y(M(\theta_1)|_{u_1,x_\tau})$ for $t \in [\tau, \tau^*]$ while $\tilde{s}_2^b(t) = Y(M(\theta_2)|_{u_2,x_{\tau^*}})$ for $t \in [\tau^*, T]$ which means by Definition 2 that $\tilde{s}_2(t)$ is segmentable. Therefore $(\tilde{s}_1(t), \tilde{s}_2(t))_\tau$ is not an actual segmentation according to Definition 2 since $\tilde{s}_2(t)$ is not a moveme output. The same argument holds for $\tau > \tau^*$. □

More generally a signal $y(t)$, $t \in [t_0, T]$, is said to be m-segmentable if there exists a sequence $t_0 < t_1, ... < t_m = T$, $m > 1$ such that $y(t)$, $t \in [t_{j-1}, t_{j+1}]$, $j = 1, ..., m-1$, is segmentable according to Definition 2 with $\tau^* = t_j$.

For the sake of simplicity, we restrict the choice of the model class M to second order linear systems described by

$$\ddot{y}(t) = \theta^T \varphi(t)\ , \tag{3}$$

where $\theta \in \mathbb{R}^3$ and $\varphi^T(t) = (-\dot{y}(t), -y(t), u(t))$, with unit step input $u(t) = 1(t) \in \mathbb{R}$. Given any signal $\tilde{y}(t)$, $t \in [t_0, T]$, we can determine the best representative of $\tilde{y}(t)$ in the class M by minimizing $\int_{t_0}^{T} (\ddot{\tilde{y}}(t) - \theta^T \tilde{\varphi}(t))^2 dt$ with respect to θ, so that

$$\hat{\theta} = \text{arg min} \int_{t_0}^{T} (\ddot{\tilde{y}}(t) - \theta^T \tilde{\varphi}(t))^2 dt \tag{4}$$

to get

$$\ddot{\hat{y}}(t) = \hat{\theta}^T \hat{\varphi}(t)\ , \qquad \hat{\varphi}(t_0) = \tilde{\varphi}(t_0)\ , \tag{5}$$

where $\hat{\varphi}(t) = (-\dot{\hat{y}}(t), -\hat{y}(t), 1(t))^T$ and $\tilde{\varphi}(t) = (-\dot{\tilde{y}}(t), -\tilde{y}(t), 1(t))^T$. We verify that the class (3) satisfies property (i) of Definition 1 by using the following lemmas.

Lemma 5. For any C^2 time signal $y(t)$ given by

$$\begin{cases} \dot{x} = \begin{pmatrix} 0 & 1 \\ -\theta_2 & -\theta_1 \end{pmatrix} x + \begin{pmatrix} 0 \\ \theta_3 \end{pmatrix} + \begin{pmatrix} 0 \\ d(t) \end{pmatrix} \\ y = (1, 0)x \end{cases} \tag{6}$$

with $d(t)$ white noise or $d(t) = 0$, let λ_1, λ_2 denote the eigenvalues of the matrix in system (6), $E = \{(v_1, v_2)^T \in \mathbb{R}^2 : v_2 = \lambda(v_1 - \frac{\theta_3}{\theta_2}),\ \lambda = \lambda_1,\ \lambda_2\}$ and $\varphi(t)^T = (-\dot{y}(t), -y(t), 1(t))$. If $x(t_0) \notin E$, then the matrix $\int_{t_0}^{t_1} \varphi(t)\varphi(t)^T dt$ with $t_1 > t_0$ is nonsingular.
Proof (sketch). Suppose the assumptions hold and suppose $\int_{t_0}^{t_1} \varphi(t)\varphi(t)^T dt$ is singular. This implies that $\dot{y} = -ay + b$ for some a and b. This equation together with system (6) imply,with $d(t) = 0$, that $x(t_0) \in E$ and with $d(t) \neq 0$ that $d(t)$ satisfies a first order differential equation, leading to contradiction. □

We will assume in what follows that assumptions of Lemma 5 are satisfied.

Lemma 6. Given the two dynamical systems $\ddot{y}_1(t) = \theta_1^T \varphi_1(t)$ and $\ddot{y}_2(t) = \theta_2^T \varphi_2(t)$, $t \in [t_0, T]$, assume $\varphi_1(t_0) = \varphi_2(t_0)$, then $y_1(t) = y_2(t)\ \forall t \in [t_0, T]$ iff $\theta_1 = \theta_2$.
Proof (sketch). ($\Leftarrow$) if $\theta_1 = \theta_2$ and $\varphi_1(t_0) = \varphi_2(t_0)$, by the uniqueness of solutions we have $y_1(t) = y_2(t)$. ($\Rightarrow$) we prove that if $\theta_1 \neq \theta_2$, then $y_1(t) \neq y_2(t)$. Suppose instead that $y_1(t) = y_2(t)\ \forall t \in [t_0, T]$, show this leads to contradiction of Lemma 5. □

We will focus on the case $\mathcal{M} = \{M^R, M^D\}$. Recalling Definition 1 let $\theta_{j,k}$ denote an element of $\mathcal{C}^j$, then we define the centers of the two sets $\mathcal{C}^R$ and $\mathcal{C}^D$ to be:

$$c_R = \frac{1}{|\mathcal{C}^R|} \sum_{k=1}^{|\mathcal{C}^R|} \theta_{R,k}$$

$$c_D = \frac{1}{|\mathcal{C}^D|} \sum_{k=1}^{|\mathcal{C}^D|} \theta_{D,k} \ ,$$

where $|\mathcal{C}^j|$ is the cardinality of the set $\mathcal{C}^j$. From here on we assume that $\mathcal{C}^R$ and $\mathcal{C}^D$ are two balls in $\mathbb{R}^3$ centered in c_R and c_D with radii r_R and r_D respectively, i.e.,

$$\mathcal{C}^R = B_{r_R}(c_R) \ , \qquad \mathcal{C}^D = B_{r_D}(c_D) \ . \quad (7)$$

In this section we have proposed a definition for a moveme, and on the basis of such a definition we showed the main properties that hold for moveme outputs. We have defined the particular model class M chosen, the set $\mathcal{M}$ of movemes and the sets $\mathcal{C}^R$ and $\mathcal{C}^D$ which parameterize the movemes.

3. SEGMENTATION PROBLEM

Given any signal $y(t)$, $t \in [t_0, T]$ which can be either non-segmentable or segmentable, we would like to consider the problem of finding its actual segmentation $(s_1(t), s_2(t))_{\tau^*}$, in which $\tau^* = T$ for the non-segmentable case. We start by looking at the simplest case in which $y(t)$ is generated by a nominal system and then we extend the result to the case of a perturbed system. Given any signal $y(t)$ let $(s_1(t), s_2(t))_\tau$ be the current segmentation at time $\tau \in (t_0, T]$. Then we define the approximation error, e_a, as

$$e_a(\tau) = \frac{1}{\tau - t_0} \int_{t_0}^{\tau} (s_1(t) - \hat{s}_1(t))^2 + \frac{1}{T - \tau} \int_{\tau}^{T} (s_2(t) - \hat{s}_2(t))^2 \ , \ \tau \in (t_0, T) \quad (8)$$

and $e_a(T) = \frac{1}{T - t_0} \int_{t_0}^{T} (s_1(t) - \hat{s}_1(t))^2$ where $\hat{s}_1(t)$ and $\hat{s}_2(t)$ are the best representatives in the class M of $s_1(t)$ and $s_2(t)$ according to (4) and (5). Similarly, we define the parametric error, e_p, as

$$e_p(\tau) = \|\hat{\theta}_1 - c_j\| + \|\hat{\theta}_2 - c_i\| \ , \ \tau \in (t_0, T) \quad (9)$$

and $e_p(T) = \|\hat{\theta}_1 - c_j\|$, where $j = R$ if $\hat{\theta}_1 \in \mathcal{C}^R$ or $j = D$ if $\hat{\theta}_1 \in \mathcal{C}^D$, and analogously for i.

3.1 *Nominal case*

Consider the segmentation problem, with $i \in \{R, D\}$ and $j \in \{R, D\}$, for the nominal system

$$\ddot{y}^0(t) = \begin{cases} c_i^T \varphi^0(t) & t \in [t_0, \tau^*] \\ c_j^T \varphi^0(t) & t \in [\tau^*, T] \end{cases}$$
$$c_i^T \varphi^0(\tau^*) = c_j^T \varphi^0(\tau^*) \ . \quad (10)$$

Letting $(s_1^0(t), s_2^0(t))_\tau$ be the segmentation at time τ of $y_0(t)$, the output of system (10), we show that the quantities $e_a^0(\tau)$ and $e_p^0(\tau)$, computed as in (8) and (9) satisfy the following lemmas.

Lemma 7. Let $e_a^0(\tau)$ and $e_p^0(\tau)$ be defined as in (8) and (9) for system (10). Then $e_a^0(\tau) = 0$ iff $\tau = \tau^*$ and $e_p^0(\tau) = 0$ iff $\tau = \tau^*$.
Proof. See (DelVecchio *et al.*, 2001).

Lemma 8. Let $e_a^0(\tau)$ and $e_p^0(\tau)$ be defined as in (8) and (9) for system (10), then $\frac{de_a^0(\tau)}{d\tau}|_{\tau=\tau^*} = 0$ and $\frac{de_p^0(\tau)}{d\tau}|_{\tau=\tau^*} = 0$; moreover there exists an interval I^0, $\tau^* \in I^0$, in which $e_a^0(\tau)$ and $e_p^0(\tau)$ are C^1 functions.
Proof. See (DelVecchio *et al.*, 2001).

The functions $e_a^0(\tau)$ and $e_p^0(\tau)$ are C^1 in I^0, both their derivatives are zero at $\tau = \tau^*$ and at such a point they have their global minimizer. It follows that they are locally convex and therefore they satisfy at $\tau = \tau^*$ first and second order necessary conditions for a minimizer. The problem of finding the actual segmentation point is then a locally convex minimization problem.

3.2 *Perturbed case*

We want to solve the segmentation problem for a signal $y(t)$, $t \in [t_0, T]$, which has been generated by a perturbed version of (10), namely by

$$\ddot{y}(t) = \begin{cases} (c_i + \delta v_1)^T \varphi(t) + d(t) & t \in [t_0, \tau^*] \\ (c_j + \delta v_2)^T \varphi(t) + d(t) & t \in [\tau^*, T] \end{cases} \quad (11)$$

where $\delta \in [0, \bar{\delta}]$, v_1 and v_2 are unit vectors, $d(t) \in [-\bar{d}, \bar{d}]\ \forall t \in [t_0, T]$ is a realization of white noise. With structure (11) we are not guaranteed anymore that $e_a(\tau)$ and $e_p(\tau)$ have a minimizer at $\tau = \tau^*$ and the obtained estimates $\hat{\theta}_1$ and $\hat{\theta}_2$ are moved away from c_i and c_j by the presence of disturbance $d(t)$ and parameter uncertainty δ. In order to let $\hat{\theta}_1 \in \mathcal{C}^i$ and $\hat{\theta}_2 \in \mathcal{C}^j$ lie in the sets $B_\delta(c_i)$ and $B_\delta(c_j)$ in which $c_i + \delta v_1$ and $c_j + \delta v_2$ lie, we can either constrain $\hat{\theta}_1$ and $\hat{\theta}_2$ to lie into balls of radii δ around the centers

c_i and c_j, or minimize $e_p(\tau)$ while also minimizing $e_a(\tau)$. Since we do not know *a priori* what i and j are, we choose the second option and reformulate the problem of segmentation as an unconstrained optimization problem. To minimize two competitive quantities we minimize $e_a(\tau)e_p(\tau)$.

Lemma 9. Let $f^0(y)$ and $g^0(y)$, $y \in [y_0, y_M]$ be C^1 non negative functions which admit their global minimum at y^* with $f^0(y^*) = g^0(y^*) = 0$. Let I denote the smallest of the convexity intervals of $f^0(y)$ and $g^0(y)$ around y^*. Let $f(y)$ and $g(y)$ be perturbed versions such that:

$$\begin{array}{c} f^0(y) - \varepsilon \leq f(y) \leq f^0(y) + \varepsilon \\ g^0(y) - \Delta \leq g(y) \leq g^0(y) + \Delta \end{array} \tag{12}$$

for all $y \in I^0 \subset I$ and $y^* \in I^0$. Then the minimizer $\bar{y} \in I^0$ of $f(y)g(y)$ is such that

$$(\bar{y} - y^*)^2 < \frac{(\varepsilon b + \Delta a) + \sqrt{(\varepsilon b + \Delta a)^2 + 8\varepsilon\Delta\bar{b}\bar{a}}}{2\bar{b}\bar{a}} \tag{13}$$

for suitable positive constants a, b, $\bar{a}$, $\bar{b}$.
Proof. See (DelVecchio *et al.*, 2001).

By proceeding with the perturbation analysis we quantify how $\hat{\theta}_1$ and $\hat{\theta}_2$ vary with respect to $\hat{\theta}_1^0$ and $\hat{\theta}_2^0$, and how the signals $s_1(t)$, $s_2(t)$ and their estimates $\hat{s}_1(t)$, $\hat{s}_2(t)$ vary with respect to the nominal signals $s_1^0(t)$, $s_2^0(t)$, $\hat{s}_1^0(t)$, $\hat{s}_2^0(t)$. Then we compute $e_a(\tau)$ and $e_p(\tau)$ to finally get that, for any $\tau \in I^0$ with $\tau^* \in I^0$ and I^0 given in Lemma 8, there exist functions $\bar{\delta}(r_D, r_R)$ and $\bar{d}(r_D, r_R)$ such that, if $\bar{\delta} < \bar{\delta}(r_D, r_R)$ and $\bar{d} < \bar{d}(r_D, r_R)$ we have $e_p^0(\tau) - \Delta \leq e_p(\tau) \leq e_p^0(\tau) + \Delta$, with $\Delta = k_{\Delta,1}(\bar{d} + \bar{d}^2 + \bar{d}^4) + k_{\Delta,2}(\bar{\delta} + \bar{\delta}^2 + \bar{\delta}^3)$, $k_{\Delta,1}$ and $k_{\Delta,2}$ suitable positive constants, r_R and r_D given in (7). As far as $e_a(\tau)$ is concerned we obtain a similar result, that is $e_a^0(\tau) - \varepsilon \leq e_a(\tau) \leq e_a^0(\tau) + \varepsilon$, with $\varepsilon = k_{\varepsilon,1}(\bar{d} + \bar{d}^2 + \bar{d}^4 + \bar{d}^8) + k_{\varepsilon,2}(\bar{\delta} + \bar{\delta}^2 + \bar{\delta}^3 + \bar{\delta}^4 + \bar{\delta}^6)$, for $k_{\varepsilon,1}$ and $k_{\varepsilon,2}$ appropriate positive constants. We can then combine all these results to give the following theorem.

Theorem 10. Given system (11), there exist constants $\bar{\delta}(r_R, r_D)$ and $\bar{d}(r_R, r_D)$ such that if $\bar{\delta} < \bar{\delta}(r_R, r_D)$ and $\bar{d} < \bar{d}(r_R, r_D)$ then the solution of the segmentation problem for $y(t)$ found by minimizing over $\tau \in I^0$ the product $e_a(\tau)e_p(\tau)$ is a real $\bar{\tau}$ which satisfies

$$(\bar{\tau} - \tau^*)^2 \leq \frac{(\varepsilon b + \Delta a) + \sqrt{(\varepsilon b + \Delta a)^2 + 8\varepsilon\Delta\bar{b}\bar{a}}}{2\bar{b}\bar{a}} \tag{14}$$

with $\Delta = k_{\Delta,1}(\bar{d} + \bar{d}^2 + \bar{d}^4) + k_{\Delta,2}(\bar{\delta} + \bar{\delta}^2 + \bar{\delta}^3)$ and $\varepsilon = k_{\varepsilon,1}(\bar{d} + \bar{d}^2 + \bar{d}^4 + \bar{d}^8) + k_{\varepsilon,2}(\bar{\delta} + \bar{\delta}^2 + \bar{\delta}^3 + \bar{\delta}^4 + \bar{\delta}^6)$, a, b, $\bar{a}$, $\bar{b}$ positive constants, r_D and r_R defined in (7).

For the complete perturbation analysis see (DelVecchio *et al.*, 2001).

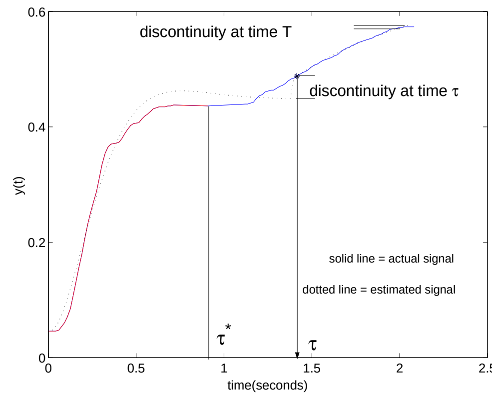

Fig. 1. **Discontinuity terms**

4. THE SEGMENTATION ALGORITHM

The actual segmentation algorithm, implemented in MATLAB 6.0 and then run on real data, minimizes the function $E(\tau) = e_p(\tau)e_a(\tau)(\alpha + e_d(\tau))$ in which we introduced the additional term $e_d(\tau)$ defined as $e_d(\tau) = |\hat{s}_1(\tau) - \hat{s}_2(\tau)| + |\hat{s}_2(T) - y(T)|$. This term accounts for the discontinuity of the estimate at $t = \tau$ and the discontinuity at $t = T$; both terms are shown in figure 1. The constant α is arbitrary positive. The general structure of the minimization algorithm is

- specify a guess of solution τ_0
- while $\frac{dE(\tau)}{d\tau} < 0$
  $\tau_{k+1} = \tau_k + \Delta\tau_k$
- stop

Typically one would choose $\Delta\tau_k = -\frac{dE(\tau)}{d\tau}|_{\tau_k}\eta_k$, with η_k chosen according to the backtracking technique for example (see (Nash and Sofer, 1996)). It is verifiable in a few steps that $\frac{dE(\tau)}{d\tau} < 0$ is equivalent to $\frac{e_p'(\tau)}{e_p(\tau)+\beta} + \frac{e_a'(\tau)}{e_a(\tau)+\gamma} + \frac{e_d'(\tau)}{\alpha+e_d(\tau)} < 0$, where β and γ have been introduced to avoid having zero denominators and they are positive arbitrary constants. This form gives evidence that in the minimization process we are looking at each step for a global percentage decrease of the functions $e_a(\tau)$, $e_p(\tau)$ and $e_d(\tau)$. Since in practice we have a sampled version of the functions $e_a(\tau)$, $e_p(\tau)$ and $e_d(\tau)$, we let $\tau = nT_s$ where $n \in \mathbb{N}$ and T_s is the sampling time, so that by replacing derivatives by finite differences it can be shown that the structure of the minimization algorithm transforms to

- let n^o be the initial guess of the minimizer
- for n=1,2,3,... check if n^o is not the optimum, i.e. if

$$\frac{e_p(n) - e_p(n^o)}{\frac{e_p(n)+e_p(n^o)}{2} + \beta} + \frac{e_a(n) - e_a(n^o)}{\frac{e_a(n)+e_a(n^o)}{2} + \gamma} + \frac{e_d(n) - e_d(n^o)}{\alpha + \frac{e_d(n)+e_d(n^o)}{2}} < 0 \tag{15}$$

  update the new minimizer n^o to n
- set $\bar{n}$ to n^o.

Definition 2 clearly establishes that the output generated by system (10) is segmentable. The same definition does not apply to the output of system (11). Then let $y(t)$, $t \in [t_0, T]$, be the output of system (11), we establish that $y(t)$ is segmentable if

$$e_p(T) > 0.5 e_p(\bar{\tau}) \;, \tag{16}$$

where $\bar{\tau} = T_s \bar{n}$ is the minimizer found with the process described above. Since we have also a sampled version of the signal $y(t)$ instead of representation (3), we use a discrete time LTI representation, see (Ljung, 1999),

$$\begin{aligned} &y(t) = \theta^T \varphi(t), \\ &\varphi(t)^T = (-y(t-1), -y(t-2), 1(t-1)) \;, \end{aligned} \tag{17}$$

so that we avoid measuring the acceleration and we just measure the output $y(t)$ itself.

5. EXPERIMENTS

We describe here the basic experiments. A more detailed description of the experimental setup and procedure is given in (DelVecchio *et al.*, 2001).

5.1 *Dataset*

We carried out our experiments on trajectories captured on four human subjects. Two simple video games were implemented in Matlab for this purpose on a commercial PC running Windows NT. The screen of the PC measured 1600×1200 pixels^2 and the working window was 800×600 pixels^2. The position of the mouse cursor was tracked from Matlab using the function "get(gca,'Current Point')" which sampled the data at approximatively 100Hz if the mouse in the working area was moving. In the first "point-and-click" game a "sequence" was initiated when three boxes about 20×20 pixel appeared at random positions in the working window. The user, starting from a base location, had to point and click inside each of the boxes and then click inside a box indicating the base to terminate the sequence. In the second "point-and-draw" game a "sequence" was initiated when a line with a marked endpoint appeared at random positions and inclinations in the working window. The user starting again from a base location had to point the marked endpoint of the line and then trace a new overlapping line, then click inside a box indicating the base to terminate the sequence. This second game had another option in which two connected lines with random inclinations and positions were appearing instead of one. The users were allowed to practice for approximatively 3 sequences for each game so to carry out each task in a natural way. In total about 70 sequences for each task were captured for each of the four subjects. The average length of a point-and-click sequence was 157 points and for a point-and-draw sequence was 182 points.

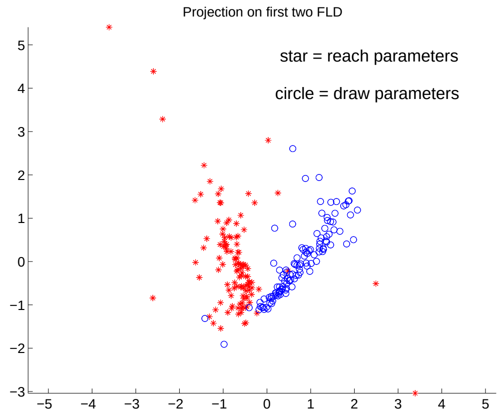

Fig. 2. **Reach and draw parameters for the x dynamics**

5.2 *Classification Problem*

For each reach sample and for each draw sample we used (17) to model the dynamics on y and x axis and (4) (with the mentioned modifications for the discrete case) to estimate the reach and draw parameters $\theta_R = ((\theta_R^x)^T, (\theta_R^y)^T))^T$ and $\theta_D = ((\theta_D^x)^T, (\theta_D^y)^T))^T$, where the superscript denotes the axis (y or x) whose dynamics was considered for the parameter estimation. The first problem we deal with is the one of correctly classifying a new reach or draw sample as reach or draw based on the dynamical parameters estimated for a training set of reach and draw samples. Then given a training set of reach/draw parameters, which we call $\tilde{\Theta}^R = \{\theta_{R,k}\}$ and $\tilde{\Theta}^D = \{\theta_{D,k}\}$ respectively, we train a classifier to distinguish between the two sets of parameters. We use the Fisher classifier (see (Bishop, 1995)) for x and y parameters separately which projects the parameters along the first two Fisher linear discriminants; in other words we find two linear transformations which transform x and y parameters in two 2D basis with respect to which the data are maximally separable. Then the training sets are mapped into two sets which we call $\tilde{C}^R$ and $\tilde{C}^D$. The distribution of x reach and draw parameters for a typical user is shown in Fig. 2 and it is clear that there are two distinct subsets. Then we train a linear neural network with signum activating function to classify the reach and draw sets both for y and x directions. We chose the data of different subjects for training and testing since in general it is likely that we have to classify the actions of people who never participated in previous experiments. Letting the above defined sets denote the x and y training sets, we obtained a training error of $6/116$ (5.17%) and test error of $5/76$ (6.5%) for the x parameters, while a training error of $4/116$ (3.4%) and a test error of $4/76$ (5.26%) for the y parameters suggesting minimal or no overfitting and excellent cross-subject generalization
Excluding from the sets $\tilde{C}^R$ and $\tilde{C}^D$ the parameters that were misclassified in this process we obtain the sets $\mathcal{C}^R$ and $\mathcal{C}^D$ which are linearly separable and then satisfy Definition 1. These sets parameterize two sets of LTI dynamical systems $M^R \subset M$ and $M^D \subset M$

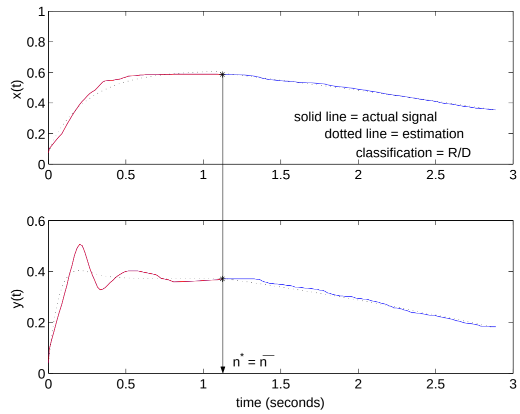

Fig. 3. **Segmentation result in a R/D case**

which according to Definition 1 are dynamical independent sets of models. Then we have found two *movemes*, the reach and draw movemes, whose output is the synthetic reproduction of the time sequence of a reach action and a draw action.

6. SEGMENTATION ALGORITHM RESULTS

Table 1: Confusion Matrix

	Predicted					
Actual	R	D	R/D	R/R	D/D	D/R
R	94	2	4	11	0	4
D	0	71	1	0	5	0
R/D	3	0	99	0	0	0
R/R	22	0	13	75	0	5
D/D	0	18	0	0	71	1
D/R	1	2	0	1	5	91

We ran the segmentation algorithm with check (15) and segmentability check (16) on the data. The classification algorithm is a subroutine of the segmentation algorithm and it is necessary to compute the proper parametric errors (9); the outputs of the segmentation algorithm are the estimated segmentation point $\bar{n}$ and the classification of the movemes found. Therefore the answer of the algorithm is correct if it has provided not only the right segmentation point, but also the right classification. In the code the possibility of recognizing when the hand is not moving has been included: the periods in which nothing happens can be identified as a pause in the resulting segmentation when it produces smaller values of the cost E. In the segmentation process we compute the quantities $e_a^x(n)$, $e_a^y(n)$, $e_p^x(n)$, $e_p^y(n)$ separately for x and y channels and then $e_a(n) = e_a^x(n)+e_a^y(n)$ and $e_p(n) = e_p^x(n)+e_p^y(n)$. The resulting errors (mis-segmentation or correct segmentation but wrong classification) are reported in the confusion matrix. The best results are obtained for R/D and D/R sequences for which in Fig. 3 we report an example, while the worst are for R/R and D/D sequences. The reason is that some of the R/R or D/D sequences really looked like just one R or D movement: in some cases the D/D sequence was performed with two lines that were almost aligned and the R/R sequence was performed reaching points which were very close to each others. In these cases the algorithm improperly classify the results.

7. CONCLUSIONS

We have proposed a dynamical formulation of movemes. We restricted our attention to two dimensional movements and showed that there exist two movemes that have dynamical characteristics which are sufficient to distinguish between them. The experiments also showed that the clusters in parameter space are not subject dependent. The segmentation algorithm was tested on about 600 samples of composed and simple actions and it gave approximatively 90% accuracy. Our analysis of second order LTI systems can also be extended to more complex dynamical systems. In the future we plan to acquire more data and look for other primitives of motion to add to the alphabet; we will also consider the case in which the action to be segmented is segmentable in more than two parts and generalize to the case of motion in 3D.

ACKNOWLEDGEMENTS

Funded in part by the NSF Engineering Research Center for Neuromorphic Systems Engineering (CNSE) at Caltech (NSF9402726), and by an NSF National Young Investigator Award to PP (NSF9457618).
We thank the people who participated in the experiments.

8. REFERENCES

Bishop, C.M. (1995). *Neural Networks for Pattern Recognition*. Clarendon. Oxford.

Bregler, C. and J. Malik (1997). Learning and recognizing human dynamics in video sequences. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition*. Puerto Rico. pp. 568–674.

DelVecchio, D., R.M. Murray and P. Perona (2001). Primitives for Human Motion: a Dynamical Approach. Technical Report CDS 01-009. California Institute of Technology, Pasadena CA 91125 URL: `http://www.cds.caltech.edu/~murray/papers/2001i_dmp01-cds.html`.

Goncalves, L., E. Di Bernardo and P. Perona (1998). Reach out and touch space (motion learning). In: *Proc. of the Third International Conference on Automatic Face and Gesture Recognition*. Nara, Japan. pp. 234–239.

Ljung, L. (1999). *System Identification*. Prentice Hall. New Jersey.

Nash, S.G. and A. Sofer (1996). *Linear and Nonlinear Programming*. McGraw-Hill. New York.