HAVE 2005 – IEEE International Workshop on
Haptic Audio Visual Environments and their Applications
Ottawa, Ontario, Canada, 1-2 October 2005

Participant Identification in Haptic Systems Using Hidden Markov Models

Yednek Asfaw, Mauricio Orozco, Shervin Shirmohammadi, Abdulmotaleb El Saddik, Andy Adler
School of Information Technology and Engineering
University of Ottawa, Ottawa, Canada
{morozco,shervin,abed}@discover.uottawa.ca, {adler,yasfaw}@site.uottawa.ca

Abstract – *Biometric systems allow identification of individuals based on behavioral or physiological characteristics. In this paper we explore biometric applications in access control of haptic systems. These systems produce information on human computer interaction behavior of a specific participant and could potentially be unique. This paper proposes a novel design based on Hidden Markov Models (HMM). Architecture is developed where each participant has an HMM model. Results are promising in that they show three out of four users identified correctly from their respective models based on the Match Score (MS) values.*

Keywords *– Biometrics, Haptic Systems, Measurement of Haptic Interaction, Hidden Markov Models*

I. INTRODUCTION

Haptic technology has seen a wide variety of applications such as modeling and animation, geophysical analysis, dentistry training, virtual museums, assembly planning, surgical simulation, and remote control of scientific instrumentation. In certain cases, the haptic system is security sensitive, such as haptic systems for military, medical and industrial applications. To control access, such sensitive systems typically have some type of login requirement, which may be based on a password, token, or perhaps a physical biometric. However, login authentication, at best, can only offer assurance that the correct person is present at the start of the session; it cannot detect if an intruder subsequently takes over the haptic controls (physically or electronically).

In this paper, we continue exploring an avenue of research proposed recently [6] to address this problem by allowing continuous authentication of participants in a haptic system. This continuous authentication is based on the characteristic pattern with which participants perform their work. We investigate techniques to automatically characterize and differentiate participants based on these data. Previous work has proposed techniques to examine haptics from a biometric aspect and has shown that simple frequency and time domain algorithms do not allow low error rates [6]. In this work, we explore the use of a more sophisticated Hidden Markov Model approach for authentication of users in such systems.

The haptic software application was developed in a combination of Python script code and VRML-based scene graph module using the PHANToM haptic interface [14]. The 3D environment was defined by using VRML-node-fields approach, while Python provided the procedural process to handle certain events and output the data to a file. The haptic stimuli are provided by accessing the Reachin API [13] which handles the complex calculations for the touch simulation and the synchronization with graphic rendering.

Data were collected by asking volunteers to complete a relatively simple maze task. The haptic maze is built on an elastic membrane surface with sticky walls and an elastic floor where the user is asked to navigate the stylus through a maze. A screenshot is shown in figure 1. Such a task allows many different behavioral attributes of the user to be measured, such as reaction time to release from a sticky wall, the route, velocity, and the pressure applied to the floor. The user must move the stylus from the entry to the exit arrow without crossing walls.

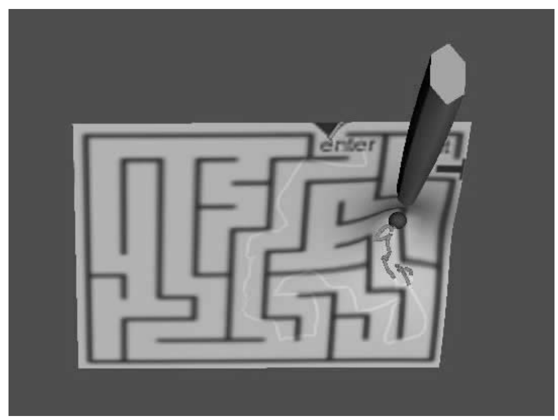

Figure 1: Screenshot of a user navigating the maze. The user navigates the maze starting at the top arrow with “enter”. The line shows the path taken by the user.

II. METHOD

A. Hidden Markov Model

Hidden Markov modeling is a powerful statistical learning technique with widespread application in pattern recognition tasks, such as speech recognition. HMMs have also been applied successfully to other language related tasks, including part-of-speech tagging, named entity recognition and text segmentation [4]. An important motivation for the use of HMMs is their strong statistical foundations, which provide a sound theoretical basis for the constructed models

[3]. On the other hand, one concern with the use of HMMs is the large amount of training data required to acquire good estimates of the model parameters [3].

HMM has a set of states, Q, an output alphabet, O, transition probabilities, A, output probabilities, B, and initial state probabilities, π. The current state is not observable. Instead, each state produces an output with a certain probability (B). Usually the states, Q, and outputs, O, are understood, so an HMM is said to be a triple, (A, B, π) [4].

B. Haptic Maze Data

During each user interaction with the haptic system, the following stylus parameters were acquired with a sample period of 15 ms: torque, force, and angle (see Figure 2). Considering that each output parameter shows difference from user to user, it is possible to use this data to do hidden Markov modeling. For this study, data from four participants were acquired; each person performed the exact same maze task 10 times, one trial immediately after the other. Participants were given the opportunity to practice the maze before the trials were actually recorded. Since there is only one correct path through the maze and the ability to solve the maze was not being judged, it was important to ensure participants knew how to correctly solve the maze in advance.

Data were divided into training and test sets. For each user, the HMM is calculated based on a training dataset containing 6 data sets of force and torque output parameters. Subsequently, the models were tested on a separate test database of 4 data sets per user of the 6 output parameters.

C. Haptic Maze Model and HMM

At a task level classification, solving a maze has 4 different states: Inactive, Stylus placement, Solve maze, and stylus Removal. The general state diagram connectivity is outlined in Figure 3. The general requirement is that the state begins and ends with an Inactive state where there is no movement of the stylus. The solve maze state is the part where the user navigates through the maze.

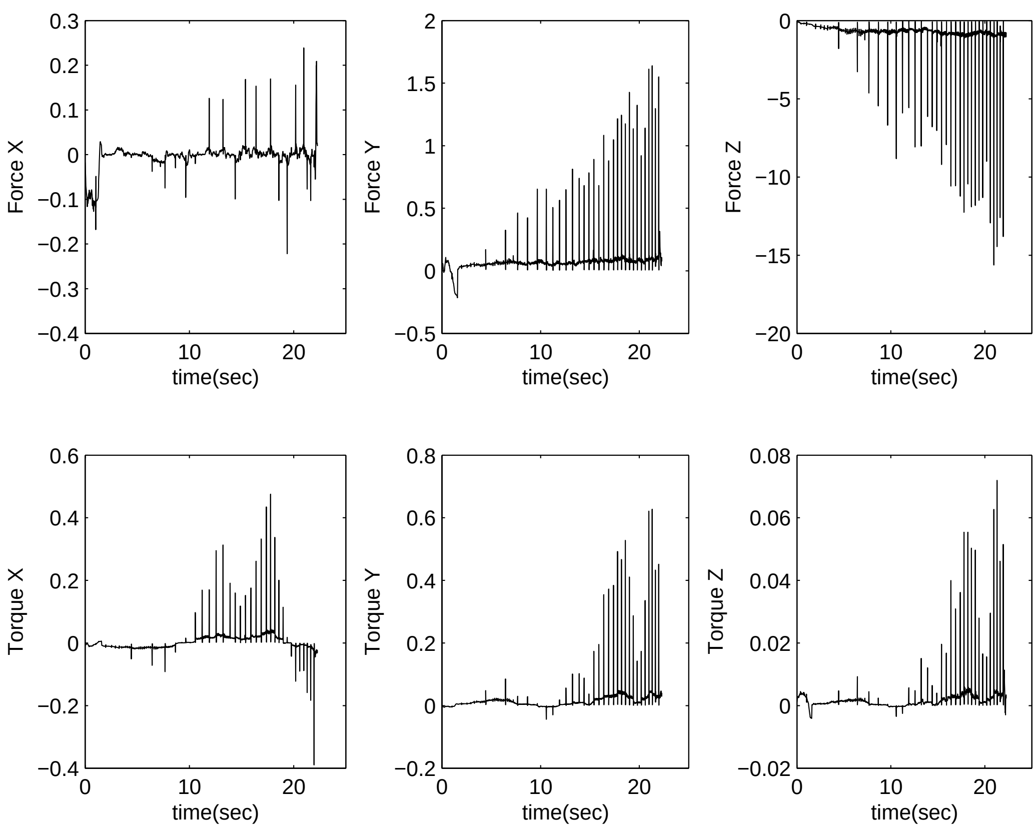

Figure 2: Raw Force and Torque output for representative participant (trail #6 of 10). Time axis is time from entering maze solve state. Force and torque units are arbitrary but the relative magnitude is constant for each axis. The z axis force is primarily negative, because pressure is applied vertically to a horizontal maze. Forces and torques increase dramatically toward maze completion. This pattern is consistent between users, but we currently cannot explain its origin.

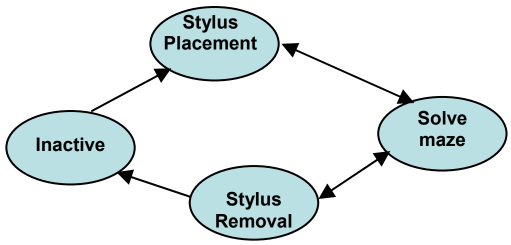

Figure 3: Task level state machine. Inactive state has no stylus movement. Stylus Placement is when the stylus is in position on the haptic maze. Solve maze is when the user navigates through the maze. Stylus Removal state when the stylus is removed from the haptic maze.

To be able to incorporate more details for participant identification, the Solve Maze state is sub-divided into more states at the maze level based on identifying strokes boundaries which are defined as sudden changes in stylus direction (Figure 4). Stroke boundaries were selected as part of the algorithm design, and identified by a human user. M=4 strokes were chosen (Figure 5). The structure for this subdivision is a left-to-right transition with no state skips as shown in Figure 4.

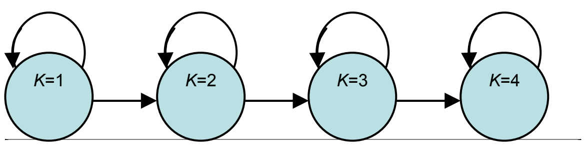

Figure 4: State sub-structure for solve maze state

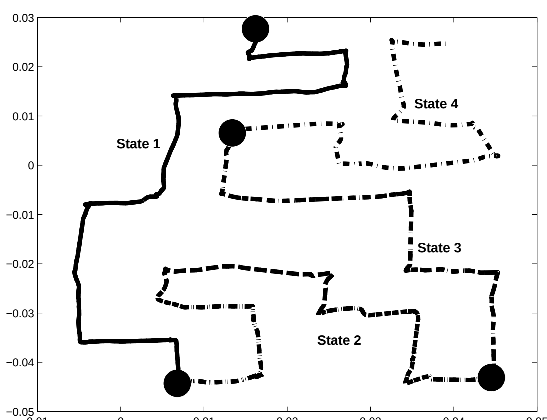

Figure 5: four stroke locations represented with four dots and the corresponding states (M=4)

The corresponding outputs for both task level and segment level states are: Stylus torque, $T(x,y,z)$, and Stylus force, $P(x,y,z)$, as a function of position (x,y,z). Figure 2 shows an example of the torque and force data for a sample subject.

The user-maze interaction is thus encoded as a time sequence of six parameters. Recorded data were quantized and normalized. Each maze is uniformly divided into $N=M*k$ time segments, where N is the length of the HMM output symbols sequence. The output parameter of a maze, is denoted by $\Phi(k)$, k=1,2,…,N:

$$\Phi(\mathrm{k}) = \frac{\mathrm{Sum(Output\ Parameter\ of\ segment\ k)}}{\mathrm{length\ of\ segment\ k}} \tag{1}$$

The normalized output parameter is then quantized into 10 levels as:

$$\Phi^*(\mathrm{k})=\mathrm{Q}[\Phi(\mathrm{k})], \quad \mathrm{k} = 1,2, \ldots, N \tag{2}$$

We assume each of the 10 levels (chosen heuristically) is represented a unique symbol (1,10,20,…100), giving 11 symbols. Thus, the maze is described by a sequence of N symbols $\Phi^*(\mathrm{k})$ of length N. A representative sequence of $\Phi^*(\mathrm{k})$ is shown in figure 6.

D. Participant Identification

Based on these states and outputs, it is possible calculate HMM parameters for each participant. The Baum-Welch algorithm is used to estimate the HMM, $\lambda = (A, B, \pi)$, as described in [2]. It generates a new estimate $\lambda_1 = (A_1, B_1, \pi_1)$ such that:

$$\Pi_i\ \mathrm{P}(\lambda_1 |\ \mathrm{O(n)}) \geq \Pi_i\ \mathrm{P}(\lambda |\ \mathrm{O(n)}) \tag{3}$$

The estimate is optimized via the EM algorithm [8] using the entire training dataset for each participant. The estimate, λ_1, is then taken as the HMM model for each participant. Each model is then tested against a test dataset to determine the probability of the dataset, $P(O|\ \lambda_1)$, belonging to a specific model via the Backward-Forward algorithm [8]. This algorithm calculates the probability of observing the partial sequence $o_1,\ldots,o_t$ and resulting in state i at time t:

$$\alpha_i(t)=\mathrm{P}(\mathrm{O}_1=\mathrm{o}_1, \ldots ,\mathrm{O}_t=\mathrm{o}_t,\mathrm{Q}_t=i|\lambda) \tag{4}$$

$P(O \mid \lambda)$ is the posterior probability of output O given the system model is λ. It is calculated as a recursive sum of $\alpha_i(t)$:

$$\mathrm{P(O} \mid \lambda) = \sum_{j=1,\ldots,N} \alpha_M(j) \tag{5}$$

$P(O \mid \lambda)$ is usually presented as the log likelihood LL= $\log(P(O|\ \lambda))$. For this application, a good match score is a negative value of LL close to zero

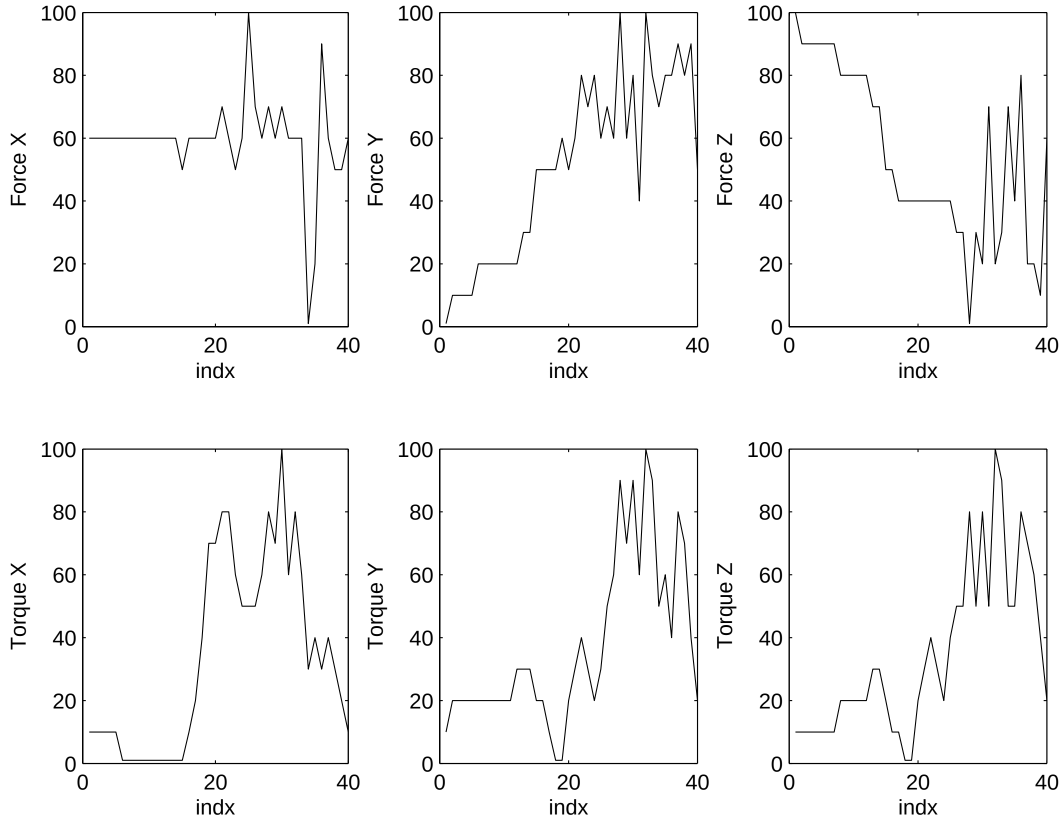

Figure 6: Normalized and quantized symbol sequence (M=4, k=10), corresponding to the raw force and torque data of figure 2.

Two different approaches were taken in this work to estimate the HMM: Single parameter HMM and Multiple parameter HMM. In a single parameter HMM, a model is created for each output parameter. Thus, each user will have six separate models corresponding to each force and torque parameter. In multiple parameter HMM, a model is created for each user based on all six output parameters. Thus, each user has a single HMM determined from 6 data sets each with 6 output parameters. The algorithm calculates a match score (*MS*) which increases with the likelihood that the model matches the data. *MS* is calculated as *sum*(*LL*) for all parameters.

III. RESULTS

For each user a six single parameter and one multiple parameter HMM was calculated based on the training data set. Subsequently, each calculated HMM is tested against test data for all users and the *MS* calculated. Thus each HMM is tested against test data for the same user and all other users. It is anticipated that *MS* be more positive for a user tested against herself than any other user.

A. Single Parameter HMM

An HMM based on torque in the Y direction is shown in figure 7. The *MS* is the sum of log likelihood values of torque in the Y for the test data set of a specific user when applied to a particular HMM. We can see that HMM of user 1, user 2 and user 3 have a high log likelihood value for their corresponding user data. However, HMM of user 4 is only within the top 2.

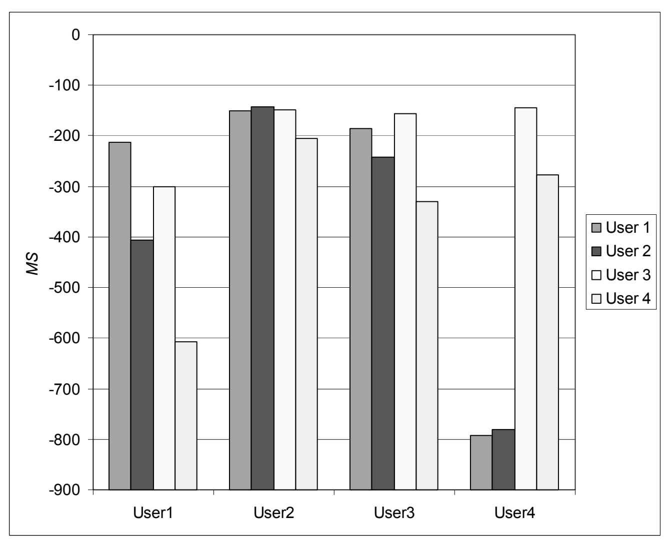

Figure 7. HMM based on torque in the Y direction: All users correspond to their HMM except user 4.

Table 1. Rank 1 and Rank 2 identification rates for all users. HMM was determined for single parameters and the percentage of identification was determined for the different ranks.

	User 1	User 2	User 3	User 4
Rank 2	100%	66%	66%	33.3%
Rank 1	50%	50%	50%	16.67%

The rank 1 and rank 2 identification rates are shown in Table 1. User 1 has a 100% identification rate for rank 2 and 50% identification rate for rank 1. Both user 2 and user 3 had identification rate of 66% for rank 2 and 50% for rank 1. The worst performer was HMM of user 4, where the test data was only identified 16.67% for rank 1 and 33.3% for rank 2.

B. Multiple Parameter HMM

In multiple parameter HMM, a model is created for each user based on all six output parameters. The match score (*MS*) is the sum of log likelihood values of the six parameter sequences of all test data of the user (when applied to a particular HMM). Figure 8 shows the *MS* results. Values for users 1, 2 and 3 correspond to their identity, and our expected results. However, results for user 4 are opposite to expectations, with the lowest *MS* for his corresponding user data.

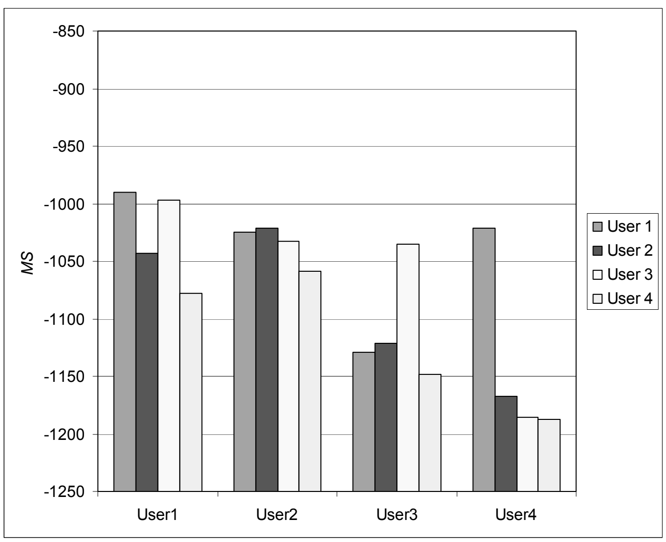

Figure 8. HMM based on all six parameters. This figure shows the sum of the log likelihood of all testing set per user.

IV. DISCUSSION AND CONCLUSION

In this paper, we develop and test a hidden Markov model (*HMM*) model of for participant identification in haptic systems. Participants solved a haptic maze while force and torque data were acquired. Two different HMMs were calculated based on training data. Each HMM for each user was tested against test data for all users, from which identification rates were calculated.

However, there are a number of issues to be explored further. Even though the result for at the "maze solve" state is promising, the general procedure during the data collection should be carried out in ways it would allow a task level segmentation of data. Incorporating the task level information should allow for better participant identification than simply working with output parameters based on how the user navigates the maze. Another issue, is the necessity to carry out pre-screening of data to remove unusual user data. Instances where the user may have gone off course or did not complete the maze in one motion currently need to be discarded.

This study considered a model with M=4 HMM states; this value was chosen to preserve some unique directional information. The results could again be improved by including more states with only the "useful" output parameters. Varying the number of segments per state to observe the impact resulted in a low *MS* with an inconsistent identification rate for M<4.

It was observed that some output parameters (such as Y direction torque) show better identification rates than others. Parameters which do not contribute to identification should be removed from the model. The single parameter HMM can be used to screening each output parameter. The multi-parameter HMM used all parameters without pre-selection. Still, the result was promising; three out of four users were on average identified to their models. Considering that this identification scheme is a continuous and live, it is possible to carry out identification on a data set within a certain time frame.

Results shown include the training effect. The training data was based on the most latter 6 datasets (out of 10 from each user) and the test set was constructed from the initial 4 datasets. This suggests that, all results shown in the project have a significant amount of training effect, and may improve if only trained data were used for testing.

In conclusion, this paper develops and evaluates algorithm for participant identification in haptic system. The algorithm was based on an HMM model using N=4 states using 6 training datasets and 6 likelihood measurements. Results were tested for a haptic maze, and showed a rank 2 identification rate of 100% for one user, 66.66% for two users, and 33.33% for the other user. While these results are mixed in terms of performance, these results suggest that participant identification is possible using HMM algorithms; however, further study of the model design, including output parameters, number of states, sequence accuracy, and quantization level needs to be investigated.

REFERENCE

[1] Rabiner, L. “A tutorial on hidden Markov models and selected application in speech recognition.” Proceedings of the IEEE, 1989.

[2] Vlontzos J. A. and Kung S. Y., “Hidden Markov Models for Character Recognition”, IEEE Transaction on Image processing,4(1), 1992. pp 539-543

[3] Yang L., Widjaja B. K., Prasad R., “Application of Hidden Markov Models for Signature Verification”. Pattern Recognition, 28(2). 1995. pp.161-170

[4] Rabiner L. R., Juang B.H., “An introduction to Hidden Markov Models”, IEEE ASSP Magazine, Jan 1986. pp. 4-16

[5] National Institute of Standards and Technology. [online] http://www.nist.gov

[6] Orozco M et al, ”Automatic Identification of participants in Haptic systems”, IEEE IMTC, 2005, pp. 888-892

[7] Kevin M., “Hidden Markov Model (HMM) Toolbox for Matlab”, http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html

[8] Bilmes J. A., “A Gentle Tutorial of the EM algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models”, International Computer Science Institute, Berkeley CA.

[9] Hundtofte et al, “Building a Task Language for Segmentation and Recognition of User input to Cooperative Manipulative Systems”, IEEE Virtual Reality Conference, 2002, pp. 225-230

[10] J.L. Wayman, "Fundamentals of Biometric Authentication Technologies", Proc. Card Tech/Secure Tech, 1999. http://www.engr.sjsu.edu/biometrics/nbtccw.pdf

[11] Michael C N, “Learning and Validation of Human Control Strategies”, Phd thesis, The Robotics Institute Carnegie Mellon University, 1998

[12] Bunke H and Caelli T. Hidden Markov Models Applications in Computer Vision.World Scientific. Series in Machine Perception and Artificial Intelligence-Vol. 45. 2001. ISBN: 981-02-4564-5

[13] Reaching Technologies http://reachin.se/

[14] Massie, T., and K. Salisbury, “The PHANToM Haptic Interface: A Device for Probing Virtual Objects”, Proc. ASME Symposium on Haptic Interfaces for Virtual Environments and Teleoperator Systems, Chicago, IL, U.S.A., November 1994, DSC-Vol. 1, pp. 295-301.