ASME International

The American Society of Mechanical Engineers
Three Park Avenue
New York, NY 10016-5990

Reprinted From
DSC-Vol. 69-2, Proceedings of the ASME Dynamic Systems and Control Division
Editor: Satish S. Nair
Book No. H1218B – 2000

A REAL-TIME STATIC POSTURE CLASSIFICATION SYSTEM

Lynne A. Slivovsky
Hong Z. Tan
Haptic Interface Research Laboratory
School of Electrical and Computer Engineering
Purdue University
1285 Electrical Engineering Building
West Lafayette, Indiana 47907-1285
E-mail: {lynnes,hongtan}@ecn.purdue.edu

ABSTRACT

As computing becomes more ubiquitous, there is a need for distributed intelligent human-computer interfaces that can perceive and interpret a user's actions through sensors that see, hear and feel. A perceptually intelligent interface enables a more natural interaction between a user and a machine in the sense that the user can look at, talk to or touch an object instead of using a machine language. The goal of the present work on a *Sensing Chair* is to enable a computer to track, in real time, the sitting postures of a user through contact sensors that act like a layer of artificial skin. This is accomplished with surface-mounted pressure distribution sensors placed on the backrest and the seatpan of an office chair. Given the similarity between a pressure distribution map from the contact sensors and a greyscale image, computer vision and pattern recognition algorithms, such as Principal Components Analysis, are applied to the problem of classifying steady-state sitting postures. A real-time multi-user sitting posture classification system has been implemented in our laboratory. The system is trained on pressure distribution data from subjects with varying anthropometrics, and performs at an overall accuracy of 96%. Future work will focus on the modeling of transient postures when a user moves from one steady-state posture to the next. A robust, real-time sitting posture tracking system can lead to many exciting applications such as automatic control of airbag deployment forces, ergonomics of furniture design, and biometric authentication for computer security.

1 INTRODUCTION

This work is motivated by the desire to make computers intelligent. Today's computers can perform astronomical computations, yet they do not have the capability to do anything more than what we tell them to do. We enter strings of commands and the computer executes them. A computer is aware of its surroundings only if the necessary information has been entered by a user through the low bandwidth devices of mouse and keyboard. Intelligence would allow a computer to interpret and anticipate the needs of a user while interacting with his or her surroundings. This can be achieved by giving a computer human-like senses such as sight, hearing, and touch. This can be done by employing digital cameras, microphones, and pressure sensors as the eyes, ears, and skin of the computer, along with algorithms that can interpret sensory input from these devices and from it, predict the user's wishes.

We can extend the notion of intelligent machines to include typically non-computational aspects of our surroundings, such as a room, a chair, a desk, and clothing. The range of items is endless. Systems are being developed that can locate, recognize, and track people; interpret gestural commands; understand natural language; and use all of this information to predict and assist the needs of a user. These include the smart room [16], the intelligent room [2, 23], the intelligent classroom [5], an intelligent house [12], clothing [16, 11], and chairs [20, 7, 4].

The majority of these systems use audio and visual cues to interpret their surroundings. The use of haptic, or touch-based, interfaces, specifically those that use pressure-sensing devices, are driving a new and exciting area of machine intelligence research. Current work in the use of pressure-sensing devices to measure sitting pressure distri-

Figure 1. THE SENSING CHAIR.

butions is focused on relating these distributions to seat comfort [9, 19, 21, 22, 1, 3, 8, 18]. Most of these systems do not perform real-time analysis, nor do they use pressure as an input to drive another application.

The goal of the Sensing Chair Project is to enable a computer to track, in real time, the sitting postures of a user through the use of surface-mounted contact sensors. A picture of the Sensing Chair is shown in Figure 1. It is an office chair (Aeron chair by Herman Miller) with surface-mounted pressure sensors (Tekscan Inc.). The problem of posture interpretation via the analysis of contact pressure can be divided into two subcategories: static posture classification and dynamic posture tracking. Static posture classification deals with the recognition of a posture while the user is in a steady-state posture (e.g., sitting upright). A real-time single-user static posture classification system was presented in [20]. The work presented here is an extension of that work into a multi-user system. Dynamic posture tracking tackles the problem of interpreting posture from sitting pressure distributions as a user moves from one static posture to another. Issues involved in dynamic posture tracking will be discussed in the future work section of this paper. The realization of a robust, real-time tracking system will lead to many exciting applications such as automatic control of airbag deployment forces, ergonomics of furniture design, and biometric authentication for computer security.

The remainder of this paper is as follows. Section 2 describes the system configuration used in the Sensing Chair Project and the pressure data that is obtained from the hardware. Section 3 presents the algorithm used for static posture classification. The results of the system are given in Section 4. The paper ends in Section 5 with a conclusion and discussion of future work.

2 SYSTEM CONFIGURATION

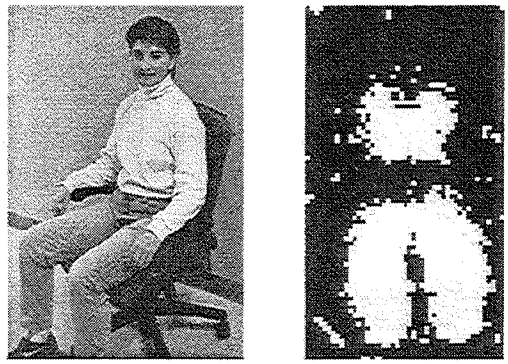
The sensing system used in the Sensing Chair Project is the Body Pressure Measurement System (BPMS) (Tekscan, Inc, South Boston,MA). It consists of two identical surface-mounted pressure-sensitive transducer sheets, their interface electronics, a PC interface board, and an API library for real-time data capture.

Each sensor sheet contains a flexible printed circuit array of 42x48 pressure sensing elements (sensels). The sensels are uniformly spaced 10 mm apart. The overall effective sensing area is 41 x 47 cm. For the Sensing Chair Project, the two sensor sheets have been mounted on the seat back and seatpan of a Herman Miller Aeron chair (Figure 1). Each sensel acts as a variable resistor. Its resistance is determined by the normal force being applied to its location. When unloaded, its resistance is high. As an applied force is increased, its resistance decreases. This resistance is converted to an 8-bit digital value. The interface electronics and PC interface board can capture the two pressure distribution maps at rates up to 127 Hz.

The sensors can be calibrated in two steps. The first step, equilibration, makes sure that equal pressures applied to two sensing elements will result in equal sensor readings. The second step, calibration, enables conversion from raw digital sensor readings to pressure values in units such as PSI or mmHg. We did not use either calibration step for three reasons. First of all, like any resistive sensor, our sensors suffer from nonlinearity, hysteresis, thermal sensitivity, drift, and sensor durability, just to name a few. Our experience indicates that the amount of noise that can be reduced by equilibration to make the effort worthwhile is relatively small. Secondly, the algorithm we have developed utilizes only the relative pressure distribution information. This makes it unnecessary to convert raw sensor data to proper pressure units. Finally, the sensor sheets are bent when a person is seated in the chair. Thus it is almost impossible to calibrate the sensor sheets with the exact conditions under which they are used. Qualitatively, the noise present in the unloaded sensors affixed to the chair is located mostly around the sensor corners [Figure 3(b)]. They don't change much once the seat is occupied (thus do not affect our algorithm).

The pressure maps can be visualized as a greyscale image. Figure 2(a) shows a person sitting in the Sensing Chair seated in an upright posture. Figure 2(b) shows the corresponding pressure map (with the seatback pressure map shown stacked above the seatpan pressure map). The seatback and seatpan data captured at the same time are always combined and treated as one image. Lighter areas in the image correspond to higher pressures, and darker areas correspond to lower pressures.

The image shown in Figure 2(b) is subject to noise due



(a) (b)

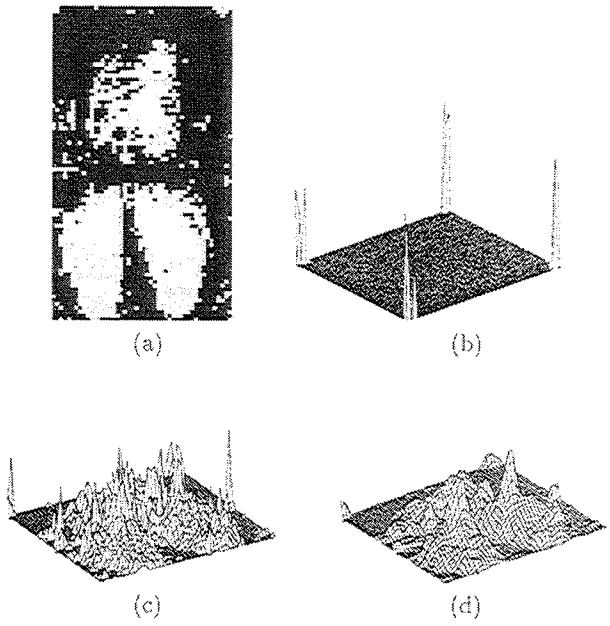
Figure 2. (a) A PERSON SEATED IN AN UPRIGHT POSTURE IN THE SENSING CHAIR. (b) AN EXAMPLE OF A RAW SITTING PRESSURE DISTRIBUTION MAP, FROM A PERSON SEATED UPRIGHT IN THE SENSING CHAIR, DISPLAYED AS AN 8-BIT GREYSCALE IMAGE. THE TOP HALF OF THE IMAGE SHOWS THE PRESSURE DISTRIBUTION ON THE BACK OF THE CHAIR, AND THE BOTTOM HALF SHOWS THAT OF THE SEATPAN. THE TOP, BOTTOM, LEFT, AND RIGHT SIDES OF THE IMAGE CORRESPOND TO THE SHOULDER AREA, KNEE AREA, RIGHT SIDE, AND LEFT SIDE OF THE PERSON, RESPECTIVELY.

to two sources: inherent *Sensor Noise*, and *Sensor Sheet Deformation*. Sensor noise can be seen as the local abrupt changes in greyscale values (e.g., the spikes in Figure 3(c)). Sensor sheet deformation introduces *pressure artifacts* into the sitting pressure distribution map that are the results of the sensors bending around and conforming to the chair. The pressure sensors in the Body Pressure Measurement System were designed to be placed on firm flat surfaces. The Aeron chair is contoured to fit the human body. To affix the pressure sensors to the chair, their corners and edges have been wrapped around the edges of the chair. This causes pressure artifacts to appear in the sitting pressure distribution maps (e.g., see the small pressure areas in the upper-left and upper-right corners of Figure 3(a)).

The raw sitting pressure distribution maps undergo preprocessing to remove the Sensor Noise in the maps. The raw sitting pressure distribution map is smoothed by convolving it with the following smoothing kernel:

$$\frac{1}{7}\begin{bmatrix} 0.5 & 1.0 & 0.5 \\ 1.0 & 1.0 & 1.0 \\ 0.5 & 1.0 & 0.5 \end{bmatrix} \quad (1)$$

Figure 3(c) shows the raw sitting pressure distribution map for a sample of posture Upright, and Figure 3(d) shows the result of applying the smoothing operator. These pressure maps are displayed as 3-D height maps, where the



(a) (b) (c) (d)

Figure 3. (a) 2-D VIEW OF A RAW SITTING PRESSURE DISTRIBUTION MAP FOR A PERSON SEATED UPRIGHT. (b) THE SENSOR NOISE PRESENT IN THE UNLOADED SENSORS SHOWN AS A 3-D HEIGHT MAP. (c) THE SITTING PRESSURE DISTRIBUTION WHEN LOADED WITH A PERSON SEATED IN AN UPRIGHT POSITION. (d) RESULT OF APPLYING THE SMOOTHING OPERATOR TO THE PRESSURE MAP SHOWN IN (c).

height of the surface above the X-Y plane indicates the pressure value at each sensel on the pressure sheets. Smoothing greatly reduces the noise spikes visible in the raw sitting pressure distribution map. It also increases the overall accuracy of our static posture classification system by 7%. The removal of pressure artifacts due to sensor sheet deformation turned out to be unnecessary.

After the sitting pressure distribution maps are smoothed, they are then normalized. The pressure values on the seatpan are usually much higher than those of the seat back. Therefore, the values in the seat back and seatpan are normalized independently.

3 STATIC POSTURE CLASSIFICATION ALGORITHM

Given the similarity between a pressure distribution map from the contact sensors and a greyscale image, it is speculated that computer vision and pattern recognition algorithms used for object recognition and classification might be applicable to the problem of static posture classification. We will therefore briefly review work on appearance-based object recognition methods in computer vision, and then

describe how we adapted this technique for posture classification.

In an appearanced-based system, objects are represented by a set of global parameters that is dependent solely on the object's visual appearance. Training of the system involves acquiring a set of training images of the objects and deriving a parameter vector from each of these images. Recognition of a test object is a matter of finding the closest match between the parameter vector of the test image and the parameter vectors of the training images.

A common approach to reducing the size of the parameter vector is through the use of Principal Components Analysis (PCA) [6]. In this method, the training images are projected onto an orthogonal space. Each image is then represented by a vector of coefficients of the principal components in this space. Recognition is then performed in this lower-dimensional space. This method has been successfully applied to the recognition of faces through the use of a set of principal face images, termed *eigenfaces*, [24, 15]. In posture classification, we can view different steady-state postures (e.g., sitting upright and slouching) as being analogous to different faces (e.g., Mike, Ted, Julie) and set up our system to classify posture instead of the person.

3.1 Static Posture Database

We have collected a small database containing Static Sitting Pressure Distribution Maps. The Static Posture Database provides the necessary training data for the development of our static posture classification system system, as well as data needed for the evaluation of the classification system. The set of postures we have selected to be in the database includes *Sitting Upright, Leaning Forward, Leaning Left, Leaning Right, Right Leg Crossed, Left Leg Crossed, Leaning Left with Right Leg Crossed, Leaning Right with Left Leg Crossed, Leaning Back, and Slouching.* The selection of a posture set is application-dependent. We have selected the above set as a starting point because it includes postures commonly found in office environments [10].

A total of 30 subjects (15 females and 15 males) participated in data collection. Five samples for each of the ten postures were collected from each subject. A total of 150 samples were collected for each posture. Subjects were selected on the basis of their overall size. The goal was to obtain subjects with a wide distribution of anthropometric measurements (e.g., weight and height).

3.2 Posture Space Training

Training one of the ten posture spaces starts with converting each of the training sitting pressure distribution maps for that posture into a vector. This is done by raster scanning each sample into a 4032x1 vector. The eigenposture space for each posture k ($k = 1...N$ with $N = 10$), is

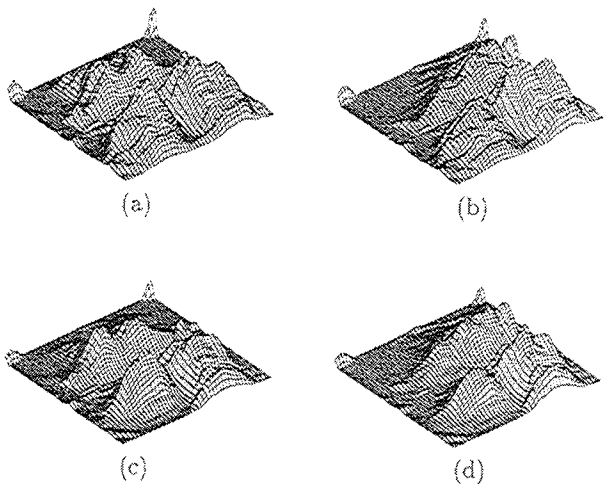


Figure 4. MEAN SITTING PRESSURE DISTRIBUTION MAPS FOR POSTURES (a) SITTING UPRIGHT, (b) LEANING LEFT, (c) RIGHT LEG CROSSED, AND (d) LEANING LEFT WITH RIGHT LEG CROSSED.

computed as follows.

Let $P_k = \{X_{1,1}, ...X_{1,S}, X_{2,1}, ...X_{2,S}, ..., X_{H,1}, ...X_{H,S}\}$ be the set of training samples for posture k, where $H = 30$ is the number of subjects. $S = 5$ is the number of samples per posture per subject. There are $HS = 150$ training samples of each posture in the Static Posture Database. For simplicity, the training set P_k is rewritten as $P_k = \{X_1, X_2, ..., X_M\}$ where $M = 150$. The mean sitting pressure distribution map vector for posture k is calculated as:

$$\bar{X}_k = \frac{1}{M} \sum_{i=1}^{M} X_i \tag{2}$$

The mean sitting pressure distribution map vectors for postures sitting upright, leaning left, right leg crosses and leaning left with right leg crossed are shown in Figure 4. The vectors have been converted back to arrays for display purposes.

Zero-mean training data is obtained by subtracting the mean from each of the training samples:

$$\phi_i = X_i - \bar{X}_k \tag{3}$$

The zero-mean training samples are stacked to form a large matrix Φ, of size 4032x150, whose covariance matrix C is:

$$C = \frac{1}{M}\sum_{i=1}^{M}\phi_i\phi_i' = \frac{1}{M}\Phi\Phi' \quad (4)$$

where $\Phi = [\phi_1, \phi_2, ...\phi_M]$ is of size 4032xM.

The eigenvectors and eigenvalues of C determine the eigen-decomposition of the training samples. Seeing that C is such a large matrix, computation of its eigenvectors is computationally expensive. Since $M = 150 < 4032$, no more than M eigenvalues are non-zero. The eigenvectors of our large 4032x4032 matrix $C = \frac{1}{M}\Phi\Phi'$ and those of the MxM matrix $C' = \frac{1}{M}\Phi'\Phi$ matrix are related in the following manner [13] [24]:

Let vectors v_i of size Mx1 be the eigenvectors of $C' = \frac{1}{M}\Phi'\Phi$. Then,

$$\frac{1}{M}\Phi\Phi' v_i = \mu_i v_i \quad (5)$$

where μ_i are eigenvalues. Premultiplying by Φ yields:

$$\frac{1}{M}\Phi\Phi'(\Phi v_i) = \mu_i(\Phi v_i) \quad (6)$$

So, the eigenvalues and eigenvectors of C are μ_i and Φv_i, respectively.

The eigen-decomposition of the training samples consists of finding the M eigenvectors v_i of $C' = \frac{1}{M}\Phi'\Phi$ and obtaining the corresponding eigenvectors of C, e_i of size 4032x1, as follows:

$$e_i = \Phi v_i = \sum_{j=1}^{M} v_{ij}\phi_i \text{ for } i = 1, 2, ...M \quad (7)$$

where v_{ij} is the j-th component of v_i. The e_i $(i = 1, 2, ...M)$ are the axes for the eigenposture space; i.e., the reduced-dimensional space for representing the original pressure distribution vectors. They are called the eigen pressure maps.

The procedure for obtaining the eigen pressure maps described above is repeated for each of the ten postures.

3.3 Static Posture Classification

Classification of a test sitting pressure distribution map is performed as follows. Figure 5 shows a flowchart for the following procedure. First, the test map undergoes the same preprocessing steps the training data has undergone (smoothing and normalization). The following steps are repeated for each posture space k. The posture mean $\bar{X}_k$ of the training samples for posture k is subtracted from the test map X_t.

$$\phi_t = X_t - \bar{X}_k \quad (8)$$

The mean-adjusted test sample ϕ_t is projected onto the first D eigenvectors of eigenposture space k to obtain a D-dimensional weight vector $W_k = [w_{k1}...w_{kD}]^T$, where the ith element of W_k is the projection of ϕ_t onto the ith eigenvector of posture space k. The test map is then reconstructed as follows:

$$\hat{\phi}_t = \sum_{i=1}^{D} w_{ki}e_{ki} \quad (9)$$

where w_{ki} is the ith element of W_k and e_{ki} is the ith eigenvector of posture space k.

The distance from posture space $d_k^2 = ||\phi_t - \hat{\phi}_t||^2$ (i.e., the Distance From Feature Space - DFFS [24]) is used as a distance measure between the test map ϕ_t and posture space k. The posture space yielding the smallest d_k^2 in the reconstruction is taken as the posture label for classification. If $\min_k d_k^2 \geq threshold$ then the sitting pressure distribution map is labeled as an *unknown posture*. The threshold is dependent on the value of D, and is determined empirically.

4 RESULTS

The static posture classification system was evaluated in three ways. First, execution time as a function of number of eigenvectors used was measured. This is an important parameter for any real-time application. The results are shown in Table 1. It was observed that when using 20 eigenvectors in classification, there is a noticeable delay between the time of moving to a new posture and that of the display of the classification result. As the number of eigenvectors is decreased, say to ten, this delay is no longer noticeable.

Second, extra pressure-map samples (from the same people who contributed to the Static Posture Database) that did not get used in eigen decomposition were used to test the accuracy of the posture classification system, again, as a function of number of eigenvectors used. There were a total of 200 extra samples, 20 for each of the ten postures. The results in Figure 6 shows an increase in accuracy as the number of eigenvectors used in classification

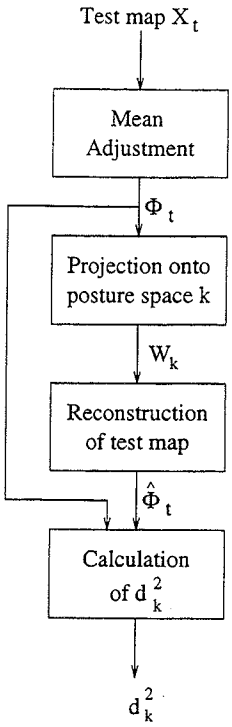


Figure 5. FLOWCHART OF THE PROCEDURE TO RECONSTRUCT A TEST MAP X_t IN POSTURESPACE k.

Number of Eigenvectors	Average Classification Time (ms)
5	62.07
10	107.81
15	168.13
20	241.0

Table 1. AVERAGE CLASSIFICATION TIME.

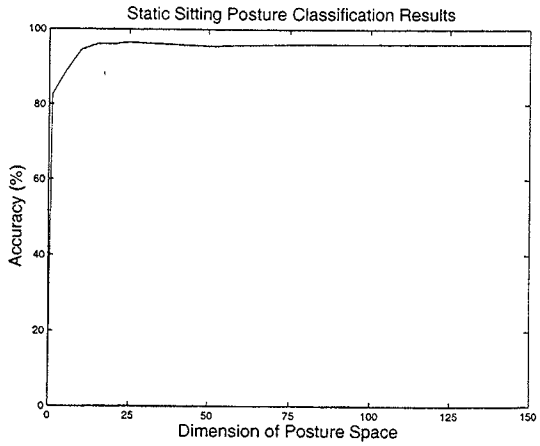


Figure 6. ACCURACY VS. THE NUMBER OF EIGENVECTORS USED FOR CLASSIFICATION. THE ACCURACY INCREASES AS THE DIMENSION D, THE NUMBER OF EIGENPOSTURE MAPS USED IN THE RECONSTRUCTION, IS INCREASED.

Posture Class	Accuracy (%)
Sitting Upright	97.4832
Leaning Forward	95.6376
Leaning Left	95.1678
Leaning Right	98.2215
Right Leg Crossed	95.0671
Left Leg Crossed	94.7651
Leaning Left with Right Leg Crossed	93.5235
Leaning Right with Left Leg Crossed	96.5436
Leaning Back	90.3020
Slouching	99.7987

Table 2. CLASSIFICATION ACCURACY FOR EACH POSTURE CLASS AVERAGED OVER DIFFERENT NUMBERS OF EIGENVECTORS USED.

is increased. Even when only 15 eigenvectors are used, the overall accuracy of the system is 96% correct. Table 2 lists the accuracy as computed for each posture averaged over different numbers of eigenvectors used. These values range from 90.3% for posture *leaning back* to 99.8% for posture *slouching*.

Third, real-time evaluation was conducted using four subjects who contributed pressure-distribution samples to the training data ("old" subjects), and four others who did not ("new" subjects). In general, the system correctly classified sitting postures for all eight subjects. Even using as few as 15 eigenvectors in classification, the system still seemed to be able to correctly classify static postures. The fact that our system performed equally well with the four "new" subjects and with the four "old" ones indicates that it can be used as a multi-user system as long as the user's anthropometry is represented in the Static Posture Database.

In view of the fact that when the number of eigenposture maps used in static posture classification increased beyond 15, (1) the average classification time increased beyond 200 ms, (2) the overall classification accuracy plateaued, and (3) there was no noticeable enhancement in real-time performance, it was decided that only the eigenposture maps corresponding to the largest 15 eigenvalues need to be used in the static posture classification system.

5 DISCUSSION

We have extended our previous work on a single-user static posture classification system to a multi-user system by training eigenposture spaces with pressure-distribution maps from a group of individuals whose anthropometrics (in terms of weight, height, age, and *paddedness* around the thighs) cover a wide range of values. An overall accuracy of 96% is achieved with a reasonably small number of eigenposture maps (15) used in classification. The system runs in near real-time on a Pentium PC. Testing with new users demonstrates the potential for a user-independent system as long as the training data reflect the anthropometrics of the general population. It is interesting to point out that classification accuracy varies with posture, with an impressive 99.8% for the posture of *slouching*. Given that this posture deprives the user of proper support in the lumbar area and could potentially lead to ailments such as lower-back pain, we envision a system that can identify almost 100% of the occurrences of this sitting posture and discourage the user from slouching for extended periods of time.

The accuracy of our static posture classification system can potentially be improved by examining the first few eigenposture spaces that produce the smallest DFFS values. For example, a small difference between two lowest DFFS values could indicate that a test map could be classified as either of the two postures. Additional information could either verify or reject the posture corresponding to the minimum DFFS value. Possible candidates for additional information may include features such as center of force, average force, and spread of the pressure peaks. One approach we are taking is to model the pressure-distribution map as a surface represented by a mixture-of-lognormal function. A lognormal distribution is preferred in order to accommodate the fact that equal pressure contours in the sitting pressure distribution maps are skewed. The distribution of the means and covariances of the lognormal densities for each of the postures will be learned from the training samples in the Static Posture Database and used for sitting posture classification in a Bayesian framework.

Our current work focuses on the development of a real-time dynamic posture classification system. Here we are specifically looking at the problem of modeling transient postures between a pair of static postures. Possible approaches to take are the manifold approach used in computer vision for estimating the pose of an object [14] or Hidden Markov modeling that is widely used in speech recognition [17]. The ultimate goal of our work on the *Sensing Chair* is to develop a chair interface that can be part of a multimodal human-computer interface system in an intelligent environment.

ACKNOWLEDGMENT

The authors would like to thank Joanne Lax for her help with the editing of an earlier version of this manuscript. The thoughtful comments from the three anonymous reviewers are greatly appreciated.

REFERENCES

[1] Richard I. Barnett and Frederick E. Shelton. Measurement of support surface efficacy: Pressure. *Advances In Wound Care*, pages 21–29, Nov/Dec 1997.

[2] Michael H. Coen. The future of human-computer interaction, or how I learned to stop worrying and love my intelligent room. *IEEE Intelligent Systems*, pages 8–10, March/April 1999.

[3] David Cohen. An objective measure of seat comfort. *Aviation, Space, and Environmental Medicine*, 69(4):410–414, April 1998.

[4] Peter Coy. Ready for its maiden flight: a new airline seat. *Business Week*, page 199, November 4 1996.

[5] David Franklin. The intelligent classroom. *IEEE Intelligent Systems*, pages 2–5, September/October 1999.

[6] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, New York, second edition, 1990.

[7] Lisa Guernsey. Magic carpet chairs of virtual reality. *The New York Times*, page G13, May 6 1999.

[8] Eric C. Hughes, Wenqi Shen, and Alicia Vertiz. The effects of regional compliance and instantaneous stiffness on seat back comfort. *Society of Automotive Engineers*, (980658), 1998.

[9] Wesley R. Iversen. Tactile sensing, 1990s style. *Assembly*, pages 23–26, Feb-Mar 1993.

[10] Rani Lueder and Kageyu Noro, editors. *Hard Facts About Soft Machines*. Taylor & Francis, London, 1994.

[11] Steve Mann. Humanistic intelligence: Wearcomp as a new framework for intelligent signal processing. *Proceedings of the IEEE*, 86(11):2123–2125, November 1998.

[12] Michael C. Mozer. An intelligent environment must be adaptive. *IEEE Intelligent Systems*, pages 11–13, March/April 1999.

[13] Hiroyasu Murakami and B. V. K. Vijaya Kumar. Efficient calculation of primary images from a set of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-4(5):511–515, 1982.

[14] H. Murase and S. K. Nayar. Visual learning and recognition of 3D objects from appearance. *International Journal of Computer Vision*, 14(1):5–24, 1995.

[15] Alex Pentland, Baback Moghaddam, and Thad Starner. View-based and modular eigenspaces for face recognition. In *IEEE International Conference on Computer Vision and Pattern Recognition*, 1994.

[16] Alex P. Pentland. Smart rooms. *Scientific American*, 274(4):68–76, April 1996.

[17] Lawrence R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2):257–285, 1989.

[18] Mary Roach. Hot seat. *Discover*, pages 74–77, March 1998.

[19] Joseph A. Sember III. The biomechanical relationship of seat design to the human anatomy. In Rani Lueder and Kageyu, editors, *Hard facts about soft machines: the ergonomics of seating.* Taylor & Francis, London, 1994.

[20] Hong Z. Tan. A Sensing Chair. In *Proceedings of the American Society of Mechanical Engineers: Dynamic Systems and Control*, Nashville, TN, November 15 1999.

[21] Kuntal Thakurta, Daniel Koester, Neil Bush, and Susan Bachle. Evaluating short and long term seating comfort. *Society of Automotive Engineers*, (950144), 1995.

[22] Kuntal Thakurta and Cindy Linder. Seat design gets comfortable. *Automotive Body Interior and Safety Systems, IBEC*, pages 7–11, 1995.

[23] Mark C. Torrance. Advances in human-computer interaction: The intelligent room. In *CHI '95 Research Symposium*, Denver, CO, May 6-7 1995.

[24] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.