

Template-based Recognition of Static Sitting Postures

Manli Zhu¹, Aleix M. Martínez¹ and Hong Z. Tan²

¹Dept. of Electrical Engineering
The Ohio State University

²School of Electrical and Computer Engineering
Purdue University

zhu.105@osu.edu, aleix@ee.eng.ohio-state, hongtan@purdue.edu

Abstract

In this paper we introduce a generalization of Fisher-Rao's discriminant analysis and its application in a human-computer interaction scenario: a sensing chair. Our algorithm shows to be able to successfully estimate the underlying distributions of the pressure maps data of the sensing chair. Other linear discriminant techniques, such as LDA, had been found to be inadequate for the job; typically yielding inferior results than PCA. We compare our approach to several template-based approaches and show that the new discriminant function is comparable to the best approach classifier. This is important because generally each application tends to prefer a different algorithm. Fortunately, our new algorithm is usually the top one (or comparable to the top one). In this paper we will however restrict our study to the classification of sitting postures.

1 Introduction

One way to improve current human-computer interaction systems is by using multimodal interfaces. Two important challenges in the area of multimodal interfaces are: i) to identify new potential modalities that can be effectively used to improve our communication with computers, and ii) to design classification algorithms that can be successfully used in several of these modalities. The sensing chair [13, 14] has recently shown to have potential toward the first goal. Unfortunately, the second point remains unsolved. The sitting posture data captured with the sensing chair has been found to be extremely challenging for most of the pattern recognition algorithms defined in the literature. To our knowledge, a simple Euclidean distance in the reduced space of the Principal Components of the covariance matrix of the pressure map is currently the

best and most robust system designed to date [14] – where results are around 85% for known users and below 75% for unknown users.

We have recently experimented with other classification algorithms such as K Nearest Neighbor (K -NN) and Linear Discriminant Analysis (LDA), only to find out that the results obtained with Principal Components Analysis (PCA) are almost always superior (see *Experimental Results* for details). Unfortunately, this is not of particular note. Martinez and Kak [10] showed that PCA can be seen as a superior technique under several conditions.

It is thus desirable to design a single classifier that can adapt to the current problem so as to perform as the best existing technique. Note that different applications are bias toward distinct approaches. We are working toward the design of problem-invariant algorithms (or close to).

In this paper, we present a first step toward this goal. Our algorithm is based on the ideas of Sliced Inverse Regression (SIR) [8] and Regularized Discriminant Analysis (RDA) [4].

2 The Sensing Chair

2.1 System Overview

Our Sensing Chair is equipped with a commercially available pressure distribution sensor called the Body Pressure Measurement System manufactured by Tekscan Inc. (South Boston, MA). This system was selected for (1) its high resolution (10 mm inter-element distance), and (2) the flexibility of the sensor sheets (0.10 mm in thickness) so they can conform to the shape of a chair. Two sensor sheets (placed inside plastic protective casings, as shown in Fig. 1) are surface mounted on the seatpan and the backrest of the chair.



Figure 1. The sensing chair.

Each sensor sheet has an array of 42-by-48 pressure sensing elements. Each sensing element outputs an 8-bit digital value proportional to the local pressure. The raw digital data are used directly as a measure of sitting pressure. A potential problem of this system is sensor displacement while an occupant moves in the chair. This problem is minimized by fastening the corners of the protective casings to the chair. To further minimize this problem, only the area that corresponds to the pressure created by the person sitting on the chair is considered (cropped out).

2.2 Data Normalization

Before each of the seat prints can be compared to the others, they need to be normalized with regard to translation and size. For simplicity, we have restricted our preliminary studies to the seatpan of the pressure map only.

The pressure maps in the seatpan are not translation invariant. People tend to sit differently: Some people prefer the center of a chair, while others tend to sit closer to one side. Similarly, while most people prefer using the backrest of the chair for support, others prefer not to. This means that the pressure points (sensors) activated are in different locations of the seatpan, and therefore the system is not translation invariant. This problem can be solved by cropping the area of the seatpan that corresponds to the contact area between the subject and the seatpan only – leaving out the rest of the zero-pressure pixels. Specifically, we project the 2D image onto each of the two dimensions, x and y axes, as shown in Fig. 2(a) and 2(b), respectively. The projection along the x-axis produces two overlapping Gaussians for the two legs. The left threshold is given by the left $u\%$ cut of the Gaussian on the left. The right threshold is given by the right $u\%$

cut of the Gaussian on the right. Fig. 2(a) shows an example with $u = 2$. The projection along the y-axis corresponds to a Gaussian distribution. We select an upper and lower threshold that omits $u\%$ of the distribution (usually, 2). Fig. 2(b) shows an example with an upper and a lower threshold of 2% . We can now crop the original seatpan pressure image with borders set by the upper, lower, leftmost and rightmost limits obtained above (see vertical lines in Fig. 2). After cropping, all images represent the rectangle that contained most of the contact area between a person and the seatpan.

The areas cropped above is now invariant to translation, but not to scale. Note that the size of the cropped area depends on the size of the person sitting on the chair, and on her/his weight. To solve this problem, we can resize all (cropped) images to a standard (normalized) size. We decided to resize all images to the average size of the windows obtained by the procedure described in the previous paragraph. Our experimentation with different sizes shows that similar results are obtained if we resize the images to other sizes (e.g., the max or min size). Alternatively, one could use algorithms to enhance the resolution of the original low-quality images [11].

2.3 Data collection

The database used in this paper is an extension of the one defined in [14]. The new database consists of 10 different postures: (1) seated upright, (2) leaning forward, (3) right leg crossed, (4) left leg crossed, (5) leaning back, (6) leaning left, (7) leaning left with right leg crossed, (8) leaning right, (9) leaning right with left leg crossed, and (10) slouching. Fifty participants (25 males and 25 females) sat in each of these postures five times. Between data collection of different samples, each participant was asked to stand up, move away from the chair, and then come back and sit down again. The data were then normalized as described above.

3 Our Classifier

Our classifier is based on the Sliced Inverse Regression (SIR) algorithm of Li [8]. In this section, we first introduce SIR, and then describe our extension for class and subclass classification.

3.1 Slice Inverse Regression

The general goal in regression analysis is to determine the conditional distribution of \mathbf{Z} given $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$, where $\mathbf{x}_i \in \mathbb{R}^p$. If the distribution $\mathbf{Z}|\mathbf{X}$

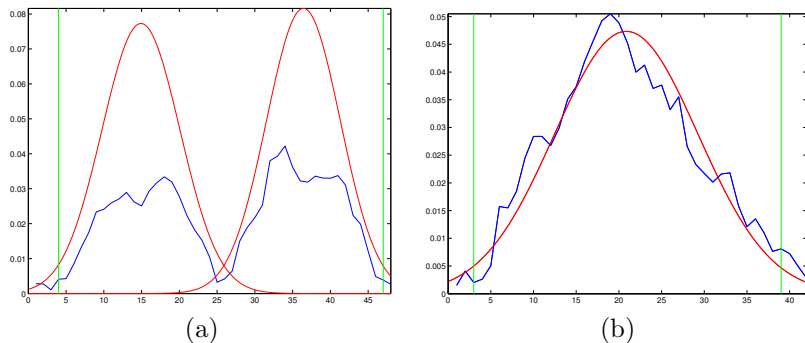


Figure 2. Normalization with respect to translation (for posture “seated upright”). (a) Projection onto the axis and the associated best-fitting Gaussian functions. (b) Projection onto the y axis and the associated best-fitting Gaussian function.

were given, we would know the class value for each of the values of \mathbf{X} . In practice, $\mathbf{Z}|\mathbf{X}$ is not known and needs to be estimated from training data. This computation is, however, problematic and important assumptions are generally necessary to make it tractable [1]. SIR reverses the roles of \mathbf{Z} and \mathbf{X} for the following reason. If we assume that \mathbf{Z} is of low dimensions, then it is easier to compute the underlying distribution of \mathbf{Z} than that of \mathbf{X} (which is of much higher dimensionality). Since SIR is a supervised classification method, we can assume that the training data has been sorted by classes so that all the samples corresponding to the first class go first, followed by the ones of the second class, etc. Mathematically, $\mathbf{Y} = \{\mathbf{x}_{1,1}, \dots, \mathbf{x}_{1,t}, \dots, \mathbf{x}_{C,t}\}$, where $\mathbf{x}_{i,j}$ is the j^{th} sample of class i , $n = C \cdot t$, and C is the number of classes. To estimate the underlying distribution of \mathbf{Z} , we divide \mathbf{Y} in H equal slices and then compute the covariance of the slice means ($H \leq C$). An estimate of that distribution is given by:

$$\Sigma_n = \sum_{h=1}^H \frac{n_h}{n} (\mu_h - \mu)(\mu_h - \mu)^T \quad (1)$$

where μ_h is the mean feature vector of slice h , μ is the global mean, and n_h is the number of samples in slice h . The SIR directions (features) are then given by the following generalized eigenvalue decomposition of Σ_n with respect to Σ_x :

$$\Sigma_n \mathbf{E} = \Sigma_x \mathbf{E} \Lambda \quad (2)$$

where \mathbf{E} is the eigenvectors matrix, $\mathbf{e}_1, \dots, \mathbf{e}_{H-1}$, and Λ is a diagonal matrix containing the corresponding eigenvalues, $\lambda_1, \dots, \lambda_{H-1}$. The eigenvectors associated with the largest eigenvalues are selected as the SIR features; $q \leq H - 1$.

3.2 SIR and LDA

3.2.1 Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) [3, 12] searches for those vectors of the underlying space of the data that best (linearly) discriminate among classes. More formally, given a number of independent features of the data, LDA creates a linear combination of those that yield the largest mean differences between the desired classes. Mathematically, for all the samples of all classes we define two measures. The first is called the within-class scatter matrix, and is given by

$$S_w = \sum_{i=1}^C \sum_{j=1}^{n_j} (\mathbf{x}_{i,j} - \mu_j)(\mathbf{x}_{i,j} - \mu_j)^T$$

where $\mathbf{x}_{i,j}$ is the i^{th} sample of class j , μ_j is the mean of class j , C is the number of classes, and n_j the number of samples in class j . The second measure is the between-class scatter matrix:

$$S_b = \sum_{j=1}^C (\mu_j - \mu)(\mu_j - \mu)^T$$

where μ represents the mean of all classes. The goal of LDA is to maximize the between-class measure while minimizing the within-class measure, which is given by the vectors that maximize the following function:

$$\max_{\mathbf{e}} \frac{\mathbf{e}^T S_b \mathbf{e}}{\mathbf{e}^T S_w \mathbf{e}}. \quad (3)$$

3.2.2 The special case of $H = C$

It is quite clear that SIR is a supervised technique similar to LDA [7]. In particular, when the number of slices, H , in SIR equals the number of classes, C , then

Σ_n is equivalent to the between-class scatter matrix [2]. To show this, we need to find the eigenvectors of Eqn. 3, which are given by:

$$S_B \mathbf{e}_i = \gamma_i S_W \mathbf{e}_i$$

Since $\Sigma_n = S_B$, the equation above can be rewritten as:

$$\Sigma_n \mathbf{e}_i = \gamma_i S_W \mathbf{e}_i$$

We now add $\gamma_i \Sigma_n \mathbf{e}_i$ on both sides:

$$(1 + \gamma_i) \Sigma_n \mathbf{e}_i = \gamma_i (\Sigma_n + S_W) \mathbf{e}_i$$

Since $\Sigma_X = S_W + S_B$ [5], we can finally rearrange the above equation to yield:

$$\Sigma_n \mathbf{e}_i = \frac{\gamma_i}{1 + \gamma_i} \Sigma_X \mathbf{e}_i \quad (4)$$

which is the same as Eqn. 2 with

$$\lambda_i = \frac{\gamma_i}{1 + \gamma_i} \quad (5)$$

This brings the following [9]:

Proposition 1: When $H = C$, the SIR dimensions (features) are the same as those obtained by LDA except for a possible difference in scaling.

3.2.3 Subclass classification

We have shown that when $H = C$, SIR is equivalent to LDA (except for a scaling factor given by Eqn. 5). The question now is what happens to SIR when H is made larger? Note that we will not consider smaller values of H , because in such a case we would only be able to distinguish between less than C classes. We obviously need to be able to discriminate between, at least, C classes.

Before we continue, we will impose an additional ordering constraint on \mathbf{Y} . The reasons for this will become apparent shortly.

The classical way to order the data is by proximity as given by the L_2 norm. Clustering is a common way to achieve this [5]. In this paper we describe a simple clustering method which has shown to be adequate for our data.

We assume that the vectors of each of the classes, $\mathbf{x}_{i,1}, \dots, \mathbf{x}_{i,t}$, have been sorted as follows. $\mathbf{x}_{i,1}$ and $\mathbf{x}_{i,t}$ are the two most distant feature vectors of class i (i.e., the Euclidean distance between these two vectors is the largest: $\text{argmax}_{j,k} \|\mathbf{x}_{i,j} - \mathbf{x}_{i,k}\|$). $\mathbf{x}_{i,2}$ is the closest feature vector to $\mathbf{x}_{i,1}$; and $\mathbf{x}_{i,t-1}$ is the closest to $\mathbf{x}_{i,t}$. In general, $\mathbf{x}_{i,j}$ is the $j - 1^{\text{th}}$ feature vector closest to

$\mathbf{x}_{i,1}$, and $\mathbf{x}_{i,t-j}$ is the $j - 1^{\text{th}}$ closest to $\mathbf{x}_{i,t}$. Fig. 3 shows the ordering of eight vectors as given by this procedure.

The clustering procedure described above is useful when H is larger than C . For example, if $H = 2C$, then each class is divided into two groups (i.e., two subclasses). This is suitable for those cases where *i*) the underlying distribution of each of the classes is not Gaussian, but can be represented as a combination of two Gaussians (see Fig. 3(c)), or *ii*) the classes are not linearly separable, but the subclasses are (see Fig. 3(d)). However, to allow such a subdivision of classes, the sample vectors in each group need to be ordered. This is the reason why we first required a clustering (ordering) step within each class.

3.2.4 The optimal value of H

To make the above approach useful for machine learning, we still need to define a way to calculate the most convenient value for H . This can be achieved by means of the leave-one-out strategy [4, 5, 6]. The technique is to remove one of the training samples from X , compute SIR using the remaining samples (for a given value of H), and then compute the recognition rate of the sample that was omitted; i.e. $p_{i,H} = 0$ if recognition fails, and $p_{i,H} = 1$ otherwise; where $p_{i,H}$ represents the recognition rate when the i^{th} sample was left out. Since, for n samples, there are exactly n possible ways to leave one sample out, we need to repeat this procedure n times. The final recognition rate is given by:

$$r_H = 1/n \sum_{i=1}^n p_{i,H}. \quad (6)$$

The optimal value of H is then given by the H ($H = j \cdot C$, $j = \{1, 2, \dots, (n/C) - 1\}$) that maximizes Eqn. 6, $H_{\text{optimal}} = \text{argmax}_H r_H$.

4 Experimental Results

We divided our data (see Section 2.3) into two groups, one for training and one for testing. The training set contains three samples of each posture of each person (i.e., a total of 1500 samples). The testing set consists of the other two samples of each posture for each of the individuals (i.e., a total of 1000 images).

In Fig. 4(a) we show the recognition rates obtained with PCA, LDA and our SIR-based approach for each of the ten postures of our database. Fig. 4(b) compares the average recognition rate of PCA, LDA and SIR to those obtained with a simple nearest neighbor classifier (NN) and a K -NN (with $K = \{3, 5, 7\}$) classifier.

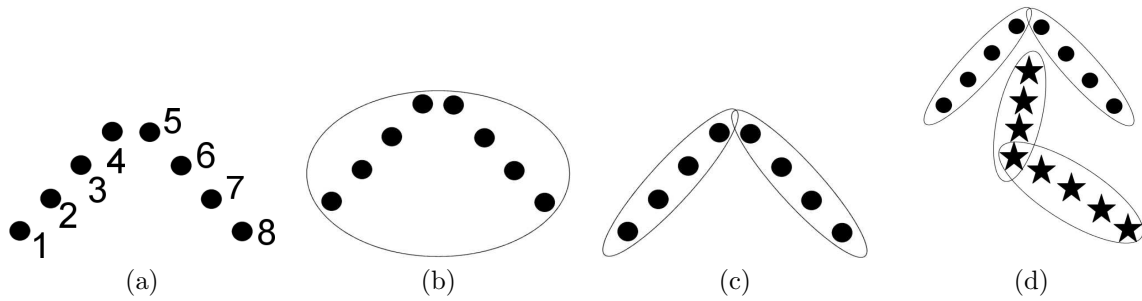


Figure 3. (a) Clustering obtained following our simple (closeness-based) approach. (b) Classification of the vectors of this class when $H = C$. (c) Classification with $H = 2C$. (d) Best classification results obtained with $H = 2C$ (circles represent the sample vectors of one class, and stars represent the sample vectors of another class).

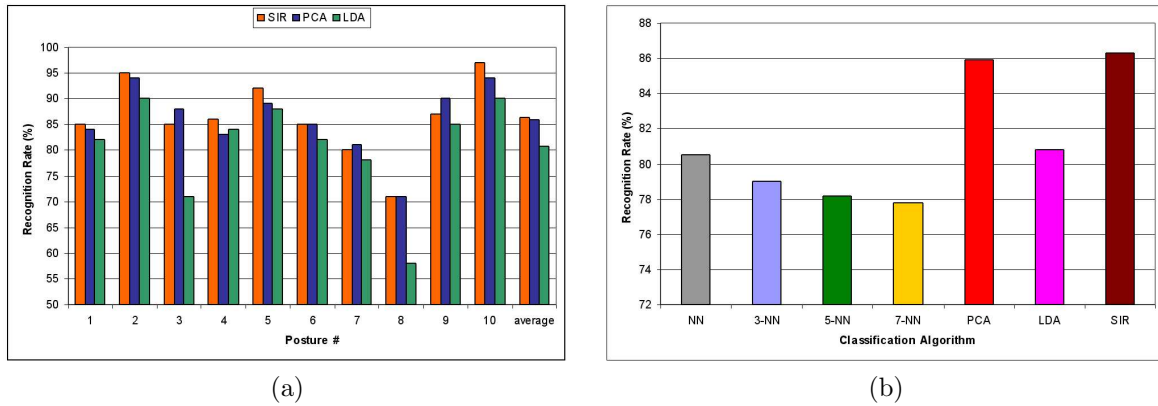


Figure 4. Experimental results. (a) Recognition rates (%) for the nine sitting postures listed in Sec. 2.3 using the LDA, PCA and SIR-based algorithms. (b) Recognition rates comparing NN and K-NN methods to those shown in (a).

The SIR-based is the leading method. The PCA approach follows closely. In fact, the differences between these two are not statistically significant. LDA is not even superior to NN. And all K -NN tested resulted with inferior results to NN.

5 Conclusions

This study generalizes the Fisher-Rao's discriminant analysis and applies it to the problem of static sitting posture classification. The recognition rates achieved by this technique are compared to those obtained with PCA, LDA, NN, and K -NN methods. We show that on average, our SIR-based method is comparable to PCA in performance, and both outperform other methods we have experimented with. We conclude that PCA and SIR can be successfully applied to new types of data encountered in human-computer interfaces such as the pressure distribution data in a chair. However, our SIR-based has the advantage that it can be successfully applied where discriminant techniques are generally superior to PCA – generally obtaining results similar to those of LDA (*Proposition 1*).

Acknowledgements

We thank the participation of Lynne Slivovsky and Christina Daniels. We also thank Kim Boyer for his comments on the subclass classification method of section 3.2.3. This research was partly supported by a National Science Foundation (NSF) Faculty Early Career Development (CAREER) Award under Grant 9984991-IIS.

References

[1] R.D. Cook, "Regression Graphics," Wiley, 1998.

[2] R.D. Cook and X. Yin, "Dimension Reduction and Visualization in Discriminant Analysis," *Aust. N.Z.J. Stat.* 43(2):147-199, 2001.

[3] R.A. Fisher, "The Statistical Utilization of Multiple Measurements," *Annals of Eugenics*, 8:376-386, 1938.

[4] J.H. Friedman, "Regularized Discriminant Analysis," *J. Am. Statistical Assoc.* 84, 165-175, 1989.

[5] K. Fukunaga, "Introduction to Statistical Pattern Recognition (2nd edition)," Academic Press, 1990.

[6] J.P. Hoffbeck and D.A. Landgrebe, "Covariance Matrix Estimation and Classification with Limited Training Data," *IEEE Trans. on Pattern Analysis and Machine Intelligence* 18(7):763-767, 1996.

[7] J.T. Kent, "Sliced Inverse regression for Dimensionality reduction: Comment," *J. American Statistical Society* 86(414):336-327, 1991.

[8] J. Li, "Sliced Inverse Regression for Dimensionality Reduction," *J. Am. Statistical Soc.* 86(414):316-327, 1991.

[9] J. Li, "High Dimansional Data Analysis Via the SIR/PHD approach," Personal Manuscript, *internet*.

[10] A.M. Martínez and A.C. Kak, "PCA versus LDA," *IEEE Trans. PAMI* 23(2):228-233, 2000.

[11] S. Peleg, D. Keren and L. Schweitzer, "Improving image resolution using subpixel motion," *Pattern Recognition Letters* 5:223-226, 2000.

[12] C.R. Rao, "Prediction of Future Observation in Growth Curve Models," *Statistical Science* 2:434-471, 1987.

[13] H.Z. Tan, I. Lu and A. Pentland, "The chair as a novel haptic user interface," In *Proc. Workshop of Perceptual User Interface*, pp. 56-57, 1997.

[14] H.Z. Tan, L.A. Slivovsky and A. Pentland, "A Sensing Chair Using Pressure Distribution Sensors," *IEEE Trans. on Mechatronics* 6(3):261-268, 2001.