6-1-2003

A Surface-based Approach for Classification of 3D Neuroanatomic Structures

Li Shen
Dartmouth College

James Ford
Dartmouth College

Fillia Makedon
Dartmouth College

Andrew Saykin
Dartmouth College

A Surface-based Approach for Classification of 3D Neuroanatomic Structures

Li Shen[1], James Ford[1], Fillia Makedon[1], Andrew Saykin[2]
[1]Dartmouth Experimental Visualization Laboratory, Computer Science,
Dartmouth College, Hanover, NH 03755
{li,jford,makedon}@cs.dartmouth.edu
[2]Brain Imaging Laboratory, Psychiatry and Radiology,
Dartmouth Medical School, Lebanon, NH 03756
saykin@dartmouth.edu

Dartmouth Computer Science Technical Report TR2003-464

Abstract

We present a new framework for 3D surface object classification that combines a powerful shape description method with suitable pattern classification techniques. Spherical harmonic parameterization and normalization techniques are used to describe a surface shape and derive a dual high dimensional landmark representation. A point distribution model is applied to reduce the dimensionality. Fisher's linear discriminants and support vector machines are used for classification. Several feature selection schemes are proposed for learning better classifiers. After showing the effectiveness of this framework using simulated shape data, we apply it to real hippocampal data in schizophrenia and perform extensive experimental studies by examining different combinations of techniques. We achieve best leave-one-out cross-validation accuracies of 93% (whole set, $N = 56$) and 90% (right-handed males, $N = 39$), respectively, which are competitive with the best results in previous studies using different techniques on similar types of data. Furthermore, to help medical diagnosis in practice, we employ a threshold-free receiver operating characteristic (ROC) approach as an alternative evaluation of classification results as well as propose a new method for visualizing discriminative patterns.

Keywords: Shape analysis, surface parameterization, classification, feature selection, statistical pattern recognition, medical image analysis

1 Introduction

Object classification via shape analysis is an important and challenging problem in machine learning and medical image analysis. Classification involves examples from distinct classes. The aim is to learn a classifier from a training set, or set of labeled examples representing different classes, and

then use the classifier to predict the class of any new example. This paper focuses on classification of 3D neuroanatomic structures using shape features in the brain imaging domain. The goal is to identify shape abnormalities in a structure of interest that are associated with a particular condition to aid diagnosis and treatment. To achieve this goal, we propose a new framework that combines a powerful 3D surface modeling technique with a set of effective pattern classification, feature selection, evaluation and visualization techniques. The proposed framework can derive a shape-based medical classifier that has clinical value. It can also be applied to 3D shape classification problems in other domains such as computer vision, image processing and pattern recognition.

Shape-based classification consists of two major steps: (1) shape representation for extracting shape parameters; and (2) pattern classification [11] for learning a classifier based on those parameters. Numerous 3D shape representation techniques have been proposed in the areas of computer vision and medical image analysis, such as, landmark-based descriptors [2, 7], deformation fields generated by mapping a segmented template image to individuals [10, 21], distance transforms [1], medial axes [25, 32, 31], and parametric surfaces [1, 4, 30]. This paper focuses on parametric surfaces using spherical harmonics. We propose a new framework of combining this representation with a set of effective pattern classification and processing techniques for classifying 3D neuroanatomic structures. Our techniques are designed for simply connected 3D objects, a category many brain structures belong to. We demonstrate these techniques using synthetic shape data as well as real hippocampal data sets extracted from magnetic resonance (MR) images.

The hippocampus is a critical structure of the human limbic system, which is involved in learning and memory processing. Several shape classification studies have been conducted for discovering hippocampal shape abnormality in the neuropsychiatric disease of schizophrenia, where classification accuracies are all estimated using leave-one-out cross-validation. Csernansky *et al.* [10] studied hippocampal morphometry using an image-based deformation representation, and achieved 80% classification accuracy through principal component analysis (PCA) and a linear discriminant. Golland, Timoner *et al.* [15, 16, 34] conducted amygdala-hippocampus complex studies using distance transformation maps and displacement fields as shape descriptors, and achieved best accuracies of 77% and 87%, respectively, using support vector machines (SVMs) [9]. We studied hippocampal shape classification in [27] using a symmetric alignment model and binary images, and achieved 96% accuracy using only the second principal component after PCA.

The above are all image-based or voxel-based approaches. We are more interested in surface-based approaches, which have the following advantages. First, compared with image-based approaches, surface-based approaches can be applied in more general situations where a surface is not embedded in an image but defined in another way such as segmented boundaries or triangulations. Second, for a 3D volumetric object, its boundary or surface actually defines the shape, and so surface-based representation may be more appropriate for shape study unless the appearance or tissue inside the object is also the focus of interest. Third, some noisy steps like resampling in the voxel-based analysis can be avoided.

The SPHARM description [4] is a parametric surface description using spherical harmonics as basis functions. Spherical harmonics was originally used as a type of surface representation for radial or stellar surfaces ($r(\theta, \phi)$) [1], and later extended to more general shapes by representing a surface using three functions of θ and ϕ [4]. Now SPHARM is a powerful surface modeling approach for arbitrarily shaped but simply-connected objects. It is suitable for surface comparison and can deal with protrusions and intrusions. Gerig, Styner and colleagues have done numerous

SPHARM studies for 3D medial shape (m-rep) modeling [32, 31], model-based segmentation [22], and identifying statistical shape abnormalities of different neuroanatomic structures [12, 32]; see [14] for a complete list. They have also done a hippocampal shape classification study [13] by using SPHARM for calculating hippocampal asymmetry, combining it with volume, and achieving 87% accuracy using SVM.

Our previous study [29] closely followed the SPHARM model and combined it with PCA and Fisher's linear discriminant (FLD) in a hippocampal shape classification study and achieved 77% accuracy. This paper extends our previous work by integrating additional classification techniques and feature selection approaches in order to obtain an improved classification. In addition, we study a threshold-free receiver operating characteristic (ROC) approach [3, 18, 33] for better understanding the behaviors of classification systems as well as propose a new method for visualization of discriminative patterns. We also discuss the computational costs involved in the study.

The rest of the paper is organized as follows. Section 2 describes the surface description approach using SPHARM expansion and point distribution model. Section 3 presents different classifiers applied in the study. Section 4 considers feature selection approaches and performs experimental studies. Section 5 examines further studies based on linear classifiers, including a threshold-free ROC analysis and a new method for visualizing discriminative patterns. Section 6 discusses the computational costs involved in the study. Section 7 concludes the paper.

2 Surface-based representation

In this study, the test data used to demonstrate our techniques are hippocampus structures extracted from magnetic resonance (MR) scans. There are 21 healthy controls and 35 schizophrenic patients involved. The left and right hippocampi in each MR image are identified and segmented by manual tracing in each acquisition slice using the BRAINS software package [20]. A 3D binary image is reconstructed from each set of 2D hippocampus segmentation results, with isotropic voxel values corresponding to whether each voxel is excluded or included. The surface of this 3D binary image is composed of a mesh of square faces (see the first picture in Figure 1 for one example of such surfaces). In this section, we present (1) how to describe such a surface using SPHARM parameterization; (2) how to normalize this SPHARM model into a common reference system so as to establish surface correspondence across different subjects as well as extract only shape information by excluding translation, rotation and scaling; (3) how to use the SPHARM model to create similar synthetic shapes that will be used as test data sets to evaluate our classification approaches; and (4) how to use a point distribution model (PDM) to obtain a more compact shape feature vector to make classification feasible.

2.1 SPHARM shape description

We adopt the SPHARM expansion technique [4] to create a shape description for closed 3D surfaces. An input object surface is assumed to be defined by a square surface mesh converted from an isotropic voxel representation (see the first picture in Figure 1 for a hippocampal surface). Three steps are involved to obtain a SPHARM shape description: (1) surface parameterization, (2) SPHARM expansion, and (3) SPHARM normalization.

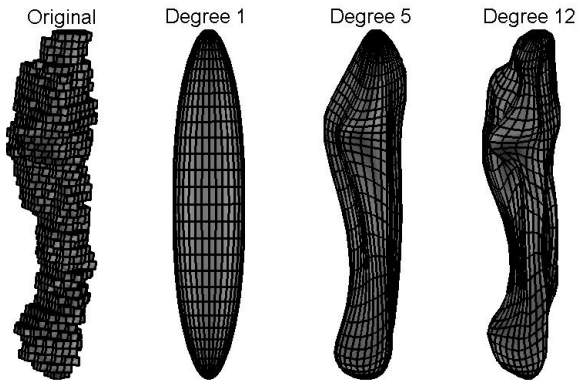


Figure 1: The first picture shows a volumetric object surface. The second, third and fourth pictures show its SPHARM reconstructions using coefficients up to degrees 1, 5 and 12, respectively.

Surface parameterization aims to create a continuous and uniform mapping from the object surface to the surface of a unit sphere. The parameterization is formulated as a constrained optimization problem with the goals of topology and area preservation and distortion minimization. The result is a bijective mapping between each point $\mathbf{v}(\theta, \phi)$ on a surface and two spherical coordinates θ and ϕ:

$$\mathbf{v}(\theta, \phi) = \begin{pmatrix} x(\theta, \phi) \\ y(\theta, \phi) \\ z(\theta, \phi) \end{pmatrix}. \tag{1}$$

The key idea in this step is to achieve a homogeneous distribution of parameter space so that the surface correspondence across subjects can be obtained later on; see [4] for details.

SPHARM expansion expands the object surface into a complete set of SPHARM basis functions Y_l^m, where Y_l^m denotes the spherical harmonic of degree l and order m. The spherical harmonics [37] are defined as

$$Y_l^m(\theta, \phi) \equiv \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}}\, P_l^m(\cos\theta)\, e^{im\phi}, \tag{2}$$

where $P_l^m(\cos\theta)$ are associated Legendre polynomials (with argument $\cos\theta$), and l and m are integers with $-l \leq m \leq l$. The associated Legendre polynomial P_l^m is defined by the differential equation

$$P_l^m(x) = \frac{(-1)^m}{2^l l!} (1-x^2)^{m/2} \frac{d^{l+m}}{dx^{l+m}} (x^2-1)^l. \tag{3}$$

The expansion takes the form:

$$\mathbf{v}(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \mathbf{c}_l^m Y_l^m(\theta, \phi), \tag{4}$$

where

$$\mathbf{c}_l^m = \begin{pmatrix} c_{xl}^m \\ c_{yl}^m \\ c_{zl}^m \end{pmatrix}. \tag{5}$$

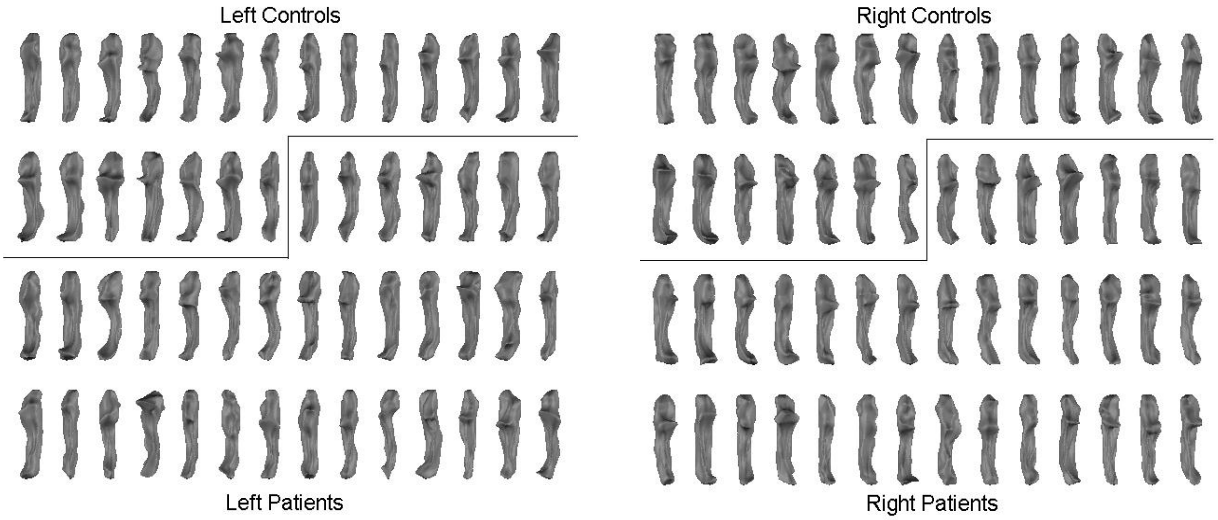


Figure 2: Normalized SPHARM reconstruction: left and right hippocampi from 21 healthy controls and 35 schizophrenic patients.

The coefficients $\mathbf{c}_l^m$ up to a user-desired degree can be estimated by solving a set of linear equations in a least squares fashion. The object surface can be reconstructed using these coefficients, and using more coefficients leads to a more detailed reconstruction. See Figure 1 for an example. Thus, a set of coefficients actually form an object surface description.

SPHARM normalization creates a shape descriptor (*i.e.*, excluding translation, rotation, and scaling) from a normalized set of SPHARM coefficients, which are comparable across objects. *Rotation invariance* is achieved by aligning the degree 1 reconstruction, which is always an ellipsoid. The parameter net on this ellipsoid is rotated to a canonical position such that the north pole is at one end of the longest main axis, and the crossing point of the zero meridian and the equator is at one end of the shortest main axis. In the object space, the ellipsoid is rotated to make its main axes coincide with the coordinate axes, putting the shortest axis along x and longest along z. *Scaling invariance* can be achieved by dividing all the coefficients by a scaling factor f. In our experiments, we choose f so that the object volume is normalized. Ignoring the degree 0 coefficient results in *translation invariance*.

After the above steps, a set of canonical coordinates (*i.e.*, normalized coefficients) can be obtained to form a shape descriptor for each object surface. Figure 2 shows the normalized reconstruction for our hippocampal data set using these shape descriptors. For simplicity, *normalized SPHARM coefficients* are hereafter referred to as *SPHARM coefficients*.

2.2 SPHARM coefficients and synthetic data generation

SPHARM coefficients are complex numbers, whose real parts and imaginary parts we treat as separate *elements*. A vector of all these elements forms a shape descriptor for a closed 3D surface. These vectors are related to the same reference system and can be compared across individuals. We assume a normal distribution for each vector element. Given a group of shapes, the mean and standard deviation of each element can be estimated to form a shape model, which may provide

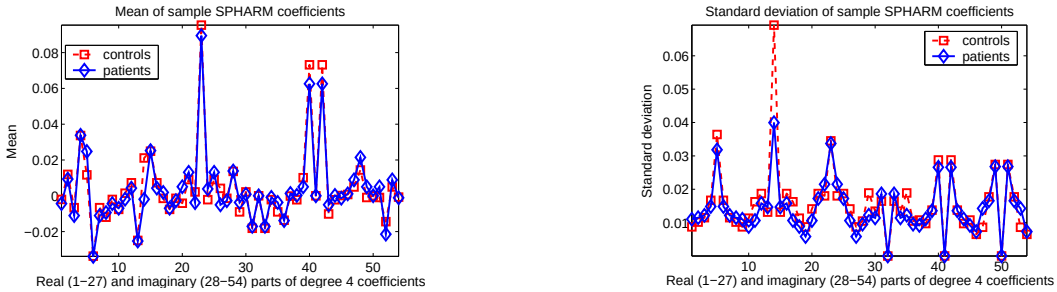


Figure 3: Mean and standard deviation of sample SPHARM coefficients for right hippocampi: real and imaginary parts of degree 4 coefficients are shown for the control group and the patient group.

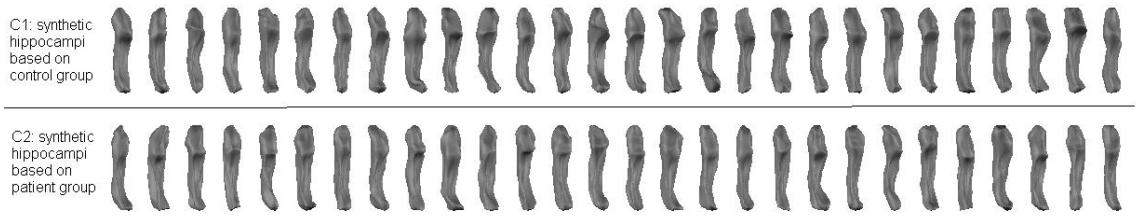


Figure 4: Synthetic right hippocampi: the first and second rows show synthetic right hippocampi generated based on the control model and the patient model, respectively.

some degree of understandings of a shape group. Figure 3 shows sample mean and standard deviation results for 2 different hippocampus groups: right hippocampi of controls (RC) and right ones of patients (RP).

The shape model described above can be used to create similar synthetic shapes. For example, a synthetic hippocampus can be constructed if, for each vector element, we draw a random number from its corresponding normal distribution with the estimated mean and standard deviation; see Figure 4 for 28 synthetic hippocampi using the RC shape model and the other 28 using the RP model, which look quite similar to real ones in Figure 2. Note that this is a very simplistic approach to synthetic shape generation, where vector elements are assumed independent. However, it is an effective one for our purpose: it can create two groups of shapes that have small and noisy group difference to evaluate our classification approach.

In fact, the SPHARM representation allows for the development of more complicated shape modeling techniques (*e.g.*, Kelemen *et al.* [22]) which can be used for synthetic data generation or even model-based segmentation.

2.3 Point distribution model

It is not easy to intuitively understand a SPHARM coefficient, since the coefficient is usually a complex number and provides a measure of the spatial frequency constituents that compose the object. However, the points of the sampled surface (called *landmarks*) can be considered as a dual representation of the same object. This is a more intuitive descriptor, and so we choose to use this representation in our study.

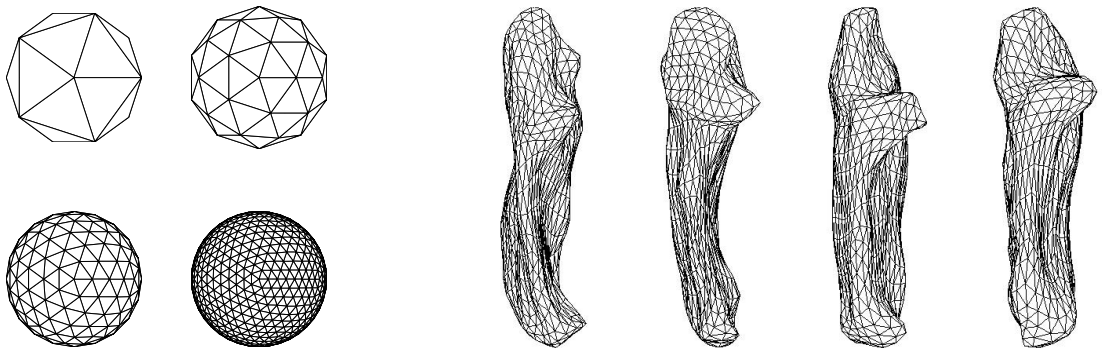


Figure 5: Landmark sampling: nearly uniform sampling of spherical surfaces by icosahedron subdivision (levels 0-3); and four sampled hippocampal surfaces (mesh vertices are landmarks).

Using a nearly uniform icosahedron subdivision of spherical surfaces [1], we obtain a dual landmark representation from the coefficients via the linear mapping described in Eq. (4). Figure 5 shows the icosahedron subdivisions of levels 0-3, as well as several sampled hippocampal surfaces using the sampling mesh of icosahedron subdivision level 3, where mesh vertices correspond to surface landmarks. Thus, each shape is represented by a set of n landmarks (*i.e.*, sampling points), which are consistent from one shape to the next.

In the experiments, we use icosahedron subdivision level 3, and each object has $n = 642$ landmarks. The mean distance ($\pm$ standard deviation) between two neighbouring landmarks is about 2.06 ± 0.86 original voxel units, where a voxel unit is 1 mm before SPHARM normalization.

Now the shape descriptor becomes a $3n$ element vector

$$\mathbf{x} = (x_1, ..., x_n, y_1, ..., y_n, z_1, ..., z_n)^T. \tag{6}$$

Given a group of N shapes, the mean shape $\bar{\mathbf{x}}$ can be calculated using

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i, \tag{7}$$

where $\mathbf{x}_i$ is the landmark shape descriptor of the i-th shape.

Clearly, we have many more dimensions than training objects. Principal component analysis (PCA) [11] is applied to reduce dimensionality to make classification feasible. This involves eigenanalysis of the *covariance matrix* $\mathbf{\Sigma}$ of the data as follows:

$$\mathbf{\Sigma} = \frac{1}{N-1} \sum_{i=1}^{N} (\mathbf{x_i} - \bar{\mathbf{x}})(\mathbf{x_i} - \bar{\mathbf{x}})^T, \tag{8}$$

$$\mathbf{\Sigma P} = \mathbf{DP}, \tag{9}$$

where the columns of $\mathbf{P}$ hold *eigenvectors*, and the diagonal matrix $\mathbf{D}$ holds *eigenvalues* of $\mathbf{\Sigma}$. The eigenvectors in $\mathbf{P}$ can be ordered according to respective eigenvalues, which are proportional to the variance explained by each eigenvector. The first few eigenvectors (with greatest eigenvalues) often explain most of variance in the data. Now any shape $\mathbf{x}$ in the data can be obtained using

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{Pb}, \tag{10}$$

where $\mathbf{b}$ is a vector containing the components of $\mathbf{x}$ in basis $\mathbf{P}$, which are called *principal components*. Since eigenvectors are orthogonal, $\mathbf{b}$ can be obtained using

$$\mathbf{b} = \mathbf{P}^T(\mathbf{x} - \bar{\mathbf{x}}). \tag{11}$$

Given a dataset of m objects, the first $m-1$ principal components are enough to capture all the data variance. Thus, $\mathbf{b}$ becomes an $m-1$ element vector, which can be thought of as a new and more compact representation of the shape $\mathbf{x}$ in the new basis of the deformation modes (*i.e.*, $\mathbf{x} - \bar{\mathbf{x}}$ is the deformation between an individual shape $\mathbf{x}$ and the mean $\bar{\mathbf{x}}$). This model is a point distribution model (PDM) [7, 22]. We apply PDM to each hippocampal data set to obtain a $\mathbf{b}$ (referred to as a *feature vector* hereafter) for each shape.

This dimensionality reduction step can also be viewed as a form of feature extraction, where the reduced representations are viewed as "features" of the orginals.

3 Classifiers

We examine several linear techniques for classifier learning, including Fisher's linear discriminants and linear support vector machines. The input data taken by these techniques are feature vectors of shapes described above. Linear techniques are simple and well-understood. Once they succeed in real applications, the results can then be interpreted more easily than those derived from complicated techniques.

3.1 Fisher's linear discriminant

Fisher's linear discriminant (FLD) is a multi-class technique for pattern classification. FLD projects a training set (consisting of c classes) onto $c-1$ dimensions such that the ratio of between-class and within-class variability is maximized, which occurs when the FLD projection places different classes into distinct and tight clumps [11].

This optimal projection $\mathbf{W}_{\text{opt}}$ is calculated as follows. Assume that we have a set of n d-dimensional samples $\mathbf{x}_1, ..., \mathbf{x}_n$, n_i in the subset $\mathcal{D}_i$ labeled ω_i, where $n = \sum_{k=1}^{c} n_k$ and $i \in \{1, ..., c\}$. Define the *between-class scatter matrix* $\mathbf{S}_B$ and the *within-class scatter matrix* $\mathbf{S}_W$ as

$$\mathbf{S}_B = \sum_{i=1}^{c} |\mathcal{D}_i|(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^T, \tag{12}$$

$$\mathbf{S}_W = \sum_{i=1}^{c} \sum_{\mathbf{x} \in \mathcal{D}_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T, \tag{13}$$

where $\mathbf{m}$ is the mean of all samples and $\mathbf{m}_i$ the mean of class ω_i. If $\mathbf{S}_W$ is nonsingular, the optimal projection $\mathbf{W}_{\text{opt}}$ is chosen by

$$\mathbf{W}_{\text{opt}} = \underset{\mathbf{W}}{\text{argmax}} \frac{|\mathbf{W}^T\mathbf{S}_B\mathbf{W}|}{|\mathbf{W}^T\mathbf{S}_W\mathbf{W}|} = [\mathbf{w}_1\mathbf{w}_2 \ldots \mathbf{w}_m] \tag{14}$$

where $\{\mathbf{w}_i \mid i = 1, 2, \ldots, m\}$ is the set of generalized eigenvectors of $\mathbf{S}_B$ and $\mathbf{S}_W$ corresponding to set of decreasing generalized eigenvalues $\{\lambda_i \mid i = 1, 2, \ldots, m\}$, *i.e.*,

$$\mathbf{S}_B \mathbf{w}_i = \lambda_i \mathbf{S}_W \mathbf{w}_i. \tag{15}$$

Note that an upper bound on m is $c - 1$; please see [11] for a detailed explanation.

In our case, we have only two classes, and so the above FLD basis $\mathbf{W}_{\text{opt}}$ becomes just a column vector $\mathbf{w}$. Once this $\mathbf{w}$ has been found, a new feature vector can be projected onto $\mathbf{w}$ to classify it. The resulting scalar value can be compared to the projections of the training set on the same basis $\mathbf{w}$. In this one-dimensional FLD space, we choose four approaches to perform classification: (1) FLD-BM, (2) FLD-1NN, (3) FLD-3NN, and (4) FLD-NM.

FLD-1NN and **FLD-3NN** are two k nearest neighbour (kNN) classifiers with $k = 1$ and $k = 3$ respectively. A kNN classifier assigns a new object to the most common class in the k nearest labelled training objects. **FLD-NM** is a nearest mean (NM) classifier, which assigns a new object to the class having the nearest mean.

FLD-BM assumes a normal distribution $\mathcal{N}(\mu_i, \sigma_i^2)$ in the FLD space for each class ω_i, where its mean μ_i and variance σ_i^2 can be estimated from the training set. Using a Bayesian model (BM), the certainty that a test subject could be explained by each class's distribution can be calculated based on the training set. That is, the *conditional probability* $P(\mathbf{x} \in \mathcal{D}_i \mid y)$ that a new object $\mathbf{x}$ belongs to class ω_i (*i.e.*, the label of $\mathcal{D}_i$), conditioned on its FLD projection being y, can then be calculated by the following equation:

$$\begin{aligned} P(\mathbf{x} \in \mathcal{D}_i \mid y) &= \frac{p(y \mid \mathbf{x} \in \mathcal{D}_i) * P(\mathbf{x} \in \mathcal{D}_i)}{\sum_{j=1}^{c} p(y \mid \mathbf{x} \in \mathcal{D}_j) * P(\mathbf{x} \in \mathcal{D}_j)} \\ &= \frac{p(y \mid y \sim \mathcal{N}(\mu_i, \sigma_i^2)) * P(\mathbf{x} \in \mathcal{D}_i)}{\sum_{j=1}^{c} p(y \mid y \sim \mathcal{N}(\mu_j, \sigma_j^2)) * P(\mathbf{x} \in \mathcal{D}_j)}. \end{aligned} \tag{16}$$

In Eq. (16), $p(y \mid \mathbf{x} \in \mathcal{D}_i)$ is the *state-conditional probability density function* (pdf) for random variable y conditioned on $\mathbf{x}$ belonging to class ω_i, which is equivalent to pdf $p(y \mid y \sim \mathcal{N}(\mu_i, \sigma_i^2))$ based on our assumption of a normal distribution for y. The *prior probability* $P(\mathbf{x} \in \mathcal{D}_i)$ of $\mathbf{x}$ belonging to class ω_i are chosen as the fraction of the dataset belonging to ω_i. FLD-BM assigns a new object to the class corresponding to the largest posterior probability computed by Eq. (16). In the case of equality, the new object joins the class having the closest mean.

3.2 Support vector machines

Support vector machines (SVMs) belong to a new generation learning system based on recent advances in statistical learning theory [36]. We apply linear SVMs in our study for a comparison with FLD-based techniques. Here we briefly describe how to train linear classifiers with SVMs.

Let $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_l, y_l)\}$, where $\mathbf{x}_i \in R^n$ and $y_i \in \{-1, 1\}$, be a set of training examples for a two-category classification problem. Define a *hyperplane* $H(\mathbf{w}, b)$ in R^n by

$$H(\mathbf{w}, b) \equiv \{\mathbf{x} \mid \mathbf{w}^{\mathbf{T}}\mathbf{x} + b = 0\}, \tag{17}$$

where $\mathbf{x}$'s are points lying on the hyperplane, $\mathbf{w}$ is normal to the hyperplane, $|b|/\|\mathbf{w}\|$ is the perpendicular distance from the hyperplane to the origin, and $\|\cdot\|$ is the Euclidean norm. Note that

$(\mathbf{w^T x} + b)/\|\mathbf{w}\|$ gives the signed distance from a point $\mathbf{x}$ to the hyperplane $H(\mathbf{w}, b)$. Thus, in a linear separable case, we can find a hyperplane $H(\mathbf{w}, b)$ such that

$$(\mathbf{w^T x_i} + b) * y_i \geq 1, \qquad i = 1, \cdots, l. \tag{18}$$

Define the *margin* as the sum of the distances from the hyperplane to the closest positive and negative exemplars. A *linearly separable SVM* aims to find the separating hyperplane with the largest margin; the expectation is that the larger the margin, the better the generalization of the classifier. For any given hyperplane satisfying the constraints in Eq. (18), the margin is $2/\|\mathbf{w}\|$. Therefore the goal is to find the hyperplane which gives the maximum margin by minimizing $\|\mathbf{w}\|^2/2$, subject to the constraints in Eq. (18).

The above scenario can be extended to a *non-separable* case by introducing *non-negative slack variables* ξ_i that measures by how much each training example violates the constraint in Eq. (18). The optimization problem is then transformed to minimizing

$$\|\mathbf{w}\|^2/2 + C(\sum_i \xi_i) \tag{19}$$

subject to constraints

$$(\mathbf{w^T x_i} + b) * y_i \geq 1 - \xi_i, \qquad i = 1, \cdots, l, \tag{20}$$

where C is a user-specified parameter for adjusting the *cost of the constraint violation*, *i.e.*, the trade-off between maximizing the margin and minimizing the number of errors.

This optimization problem is solved by a quadratic programming approach using Lagrange multipliers. Based on the resulting hyperplane $H(\mathbf{w}, b)$, a new example $\mathbf{x}$ can be classified by calculating $f(\mathbf{x}) = \text{sign}(\mathbf{w^T x} + b)$. SVM can also be extended to *nonlinear classification* via a nonlinear mapping defined by kernel functions. SVMs have been receiving increasing attention and been used successfully in many classification areas. We refer the readers to [5, 6, 9, 11, 18, 23, 36] for more technical and implementation details.

To test the effectiveness of our framework as well as compare with FLD-based techniques, we employ a publicly available SVM tool in our study. The tool we use is *OSU SVM Classifier Matlab Toolbox version 3.00* [23], the core part of which is based on *LIBSVM v2.33* [6]. Only linear SVM classifiers are tested in our experiments. We use **SVM-Ct** to denote a linear SVM classifier in which the parameter C, for specifying the cost of the constraint violation, is set as t. **SVM-C1**, **SVM-C10**, **SVM-C100** are applied in our experiments, where C takes values of 1, 10 and 100 respectively.

4 Experimental studies

We conduct experimental studies on classification in this section. Classification is performed on feature vectors after PCA using a *leave-one-out cross-validation* methodology [11]:

> Each object is removed in turn as the test case, the remaining objects forms a training set for classifier learning, the resulting classifier is tested on the removed object, and the accuracy is estimated as the mean of these leave-one-out accuracies.

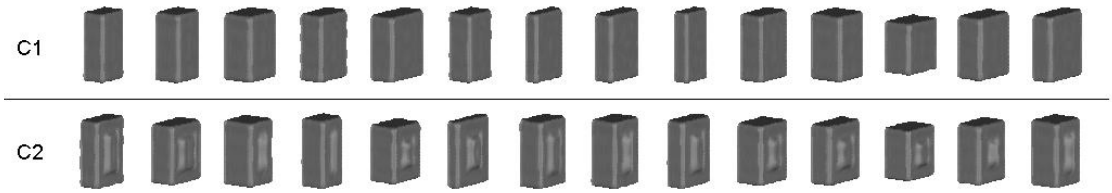


Figure 6: Two classes of synthetic surfaces: the first row shows 14 surfaces in Class C1 (or Cuboid Class), while the second shows 14 surfaces in Class C2 (or Cuboid-bump Class).

This work, unlike our previous work [29], uses PCA applied to all data in a single step, rather than constructing a new basis for each leave-one-out trial based on individual training sets. This is a simpler approach that should minimize representation errors.

In the classification, principal components are used as features, and different orderings of these features are considered. The standard ordering of principal components is by the variance amounts they explain. An alternative ordering of these components by statistical tests is also investigated in the following section. In either case, varying numbers of features are examined.

To evaluate the effectiveness of our techniques, synthetic data are created and employed in the experiments. After that, the techniques are applied to the real hippocampal data in schizophrenia and the results are reported.

4.1 Feature selection

Additional features are theoretically never unhelpful. In theory having more features can only improve or not change performance; however, in practice, each additional feature adds a parameter to the model that needs to be estimated, and mis-estimations that result from the less informative features can actually degrade performance. This trend of decreasing accuracy gains followed by actual losses of accuracy from additional features is known as the "peaking effect" or "Hughes phenomenon" [19]. Therefore, in summary, it is often helpful to select a subset of the most useful features. In our study, features are principal components, and we feel that some components are more *useful* than others for classification, but not necessarily matching the ordering of the variance amounts they explain. The following is such an example.

Figure 6 shows two classes of 14 synthetic rectangular surfaces each, with bumps centered on one face in the second class. Here we use the standard ordering of principal components (*i.e.*, by variance-accounted-for). We apply our FLD and SVM classifiers to this data, and the cross-validation accuracies using the first i components are: $< 60\%$ for $i = 1, 2$; and 100% for $i = 3, \ldots, 24$. As shown in the left part of Figure 7, the third component alone supports perfect classification, and thus should be considered more important than components that do not. Note that the third eigenvector contains the weights to create the third component. These weights can be backprojected onto the surface space and each landmark corresponds to a vector of 3 weights. The weight vectors with the largest magnitude correspond to landmark locations with the most significant contribution to the component value. The right part in Figure 7 shows the contributions of landmark locations to the third component using the color mapping onto the mean surface. From this visualization, it is apparent that the third component is focused on the most significant surface region for discriminating the synthetic classes.

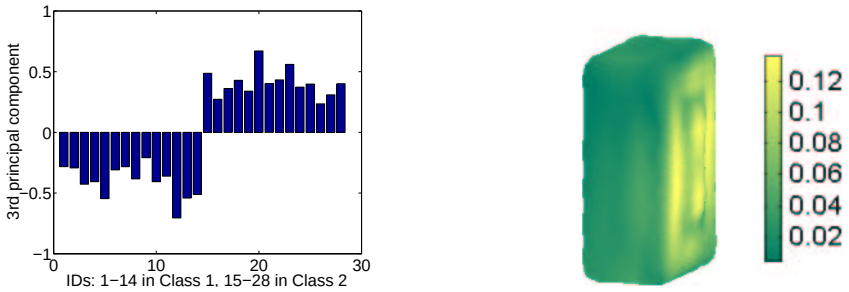


Figure 7: Left: third principal component of synthetic surfaces, which discriminates classes. Right: mean surface with colors indicating the contributions of landmark locations to the third component, where yellow/light color indicates more significant contributions while green/dark less.

To rank the effectiveness of features, we employ a simple two-sample t-test [24] on each feature and obtain a p-value associated with the test statistic

$$T = \frac{\bar{Y}_1 - \bar{Y}_2}{\sqrt{s_1^2/N_1 + s_2^2/N_2}}, \tag{21}$$

where N_1 and N_2 are the sample sizes, $\bar{Y}_1$ and $\bar{Y}_2$ are the sample means, s_1^2 and s_2^2 are the sample variances, and the samples are two sets of feature values in two respective classes. The p-value indicates the probability that the observed value of T could be as large or larger by chance under the null hypothesis that the means of Y_1 and Y_2 are the same. Thus, a lower p-value implies stronger group difference statistically and corresponds to a more *significant* feature. We hypothesize that more significant features can help more in classification. In the above example, the third principal component corresponds to $p < 10^{-15}$, while for all the other components $p \geq 0.17$.

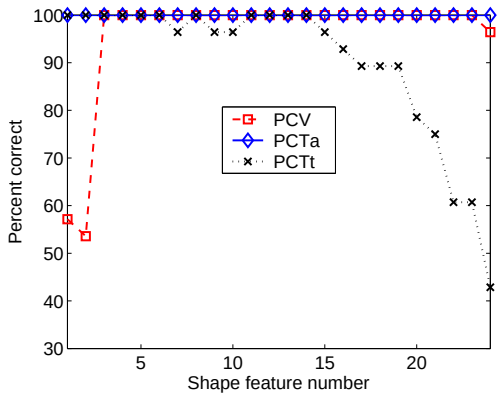
We investigate three feature selection schemes in our experiments. In each scheme, we select the first n features according to a certain ordering of principal components, where varying values of n are also considered. These orderings are as follows:

1. `PCV`: ordered by variance explained, decreasingly.
2. `PCTa`: ordered by p-value associated with t-test applied to all the objects, increasingly.
3. `PCTt`: ordered by p-value associated with t-test applied only to each leave-one-out training set, increasingly, where different leave-one-out trials could have different `PCTt` orderings.

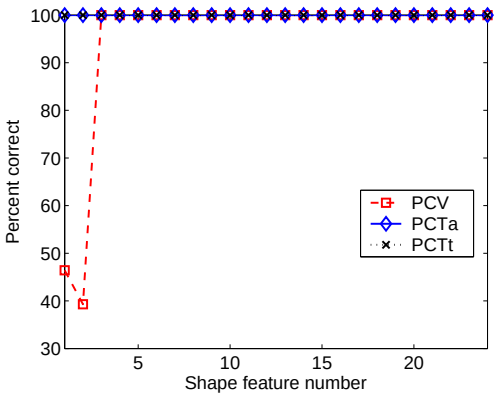
4.2 Experiments on synthetic data

We report our experimental results on two synthetic data sets using FLD-BM and SVM-C1 classifiers. The first data set consists of two classes of 14 synthetic rectangular surfaces each, with bumps centered on one face in the second class. Figure 6 shows these surfaces. Although the group shape difference is clear in this example, some variability also occurs within each group.

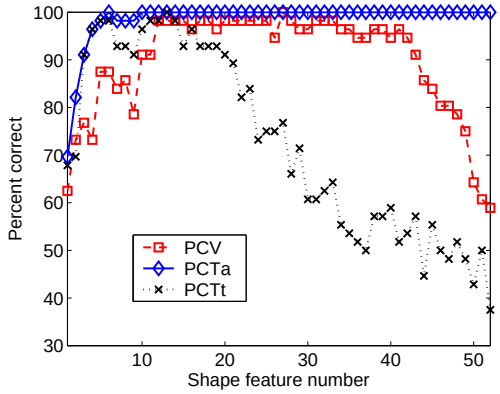
Figure 8(a) and Figure 8(b) show the results of applying FLD-BM and SVM-C1, respectively, to this data set. Both figures give us the following observations. The 100% cross-validation accuracy is very consistent if we use more than two features according to `PCV` ordering. Using `PCTa`



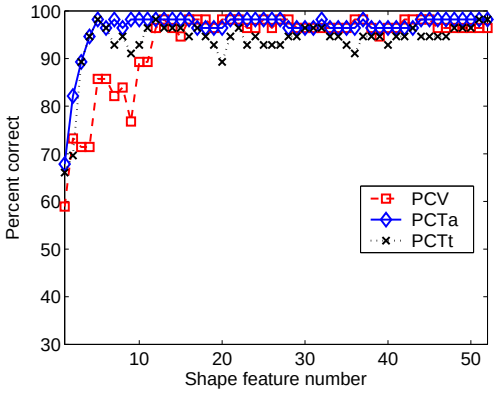
(a) FLD-BM on rectangular surfaces



(b) SVM-C1 on rectangular surfaces



(c) FLD-BM on synthetic hippocampi



(d) SVM-C1 on synthetic hippocampi

Figure 8: Sample classification results on simulated data sets using different feature selection schemes. Rectangular surfaces and synthetic hippocampi are two simulated data sets. FLD-BM and SVM-C1 are two classifiers. `PCV`, `PCTa`, `PCTt` are three feature selection schemes. On each picture, the number of features used in the classification according to a certain ordering (`PCV`, `PCTa`, or `PCTt`) is plotted on the X axis, the leave-one-out cross-validation accuracy is plotted on the Y axis.

or PCTt ordering can achieve the perfect accuracy using the minimum number of features; in this case, only one feature is needed for a perfect classification. These results suggest that our techniques can effectively detect the group difference in the presence of noisy intra-group variabilities.

The second simulated data set is formed by the synthetic hippocampi generated in Section 2.2: Class C1 consists of 28 synthetic hippocampi based on the right control (RC) shape model, and Class C2 contains 28 synthetic ones based on the right patient (RP) model. Please see Figure 4 for the visualization of both classes. Due to the minor difference between the RC and RP shape models (*e.g.*, the model difference displayed in Figure 3), these two classes of shapes have a small and noisy group difference.

Figure 8(c) and Figure 8(d) show the experimental results of applying FLD-BM and SVM-C1, respectively, to this second data set (*i.e.*, synthetic hippocampi). Using PCTa and PCTt feature selection orderings, excellent accuracies ($\geq 95\%$) can be achieved using very few (*e.g.*, 5) features. Using the PCV ordering, similar accuracies can be achieved using more (*e.g.*, 12) features. There is a performance difference between FLD-BM and SVM-C1 using either PCV or PCTt feature selection: adding too many features hurts FLD-BM but does not affect SVM-C1. In terms of the best cases, all these techniques can achieve nearly perfect leave-one-out cross-validation classification accuracies. Synthetic hippocampi generated from different groups have small but systematic differences in terms of mean shape and coefficient variance (see Figures 3). Our techniques seem to be able to capture these differences.

The above experiments show that our classification approach performs well at distinguishing clear group differences as well as small and noisy group differences. In the following section, we apply our technique to real hippocampal data to detect if there is any hidden group difference which can potentially help medical diagnosis.

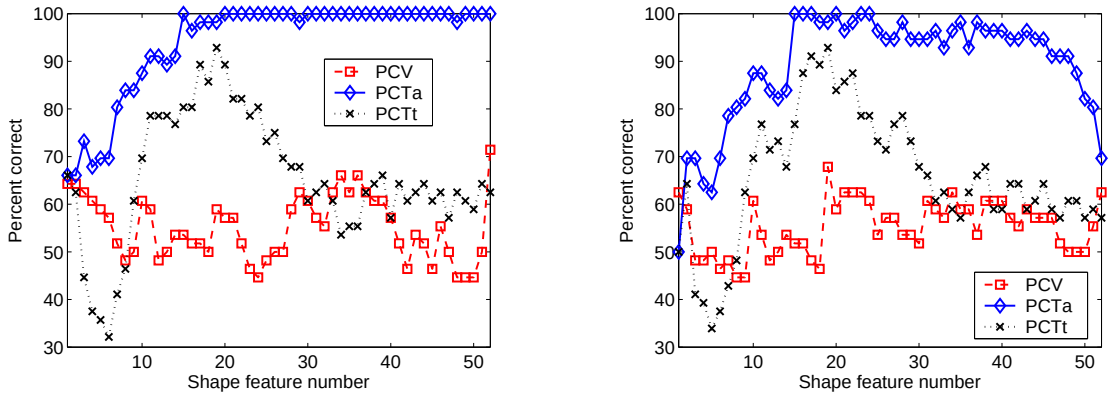
4.3 Experiments on real hippocampal data

In this section, we report our experimental results on real hippocampal data. In many clinical studies, the relatively unknown contributions of gender and handendess are controlled for by selecting subjects based on only one particular value for these parameters, typically "male" and "right-handed". Accordingly, we also present results with the right-handed male subset of our subject pool in order to facilitate comparisons with studies using these values as selection criteria.

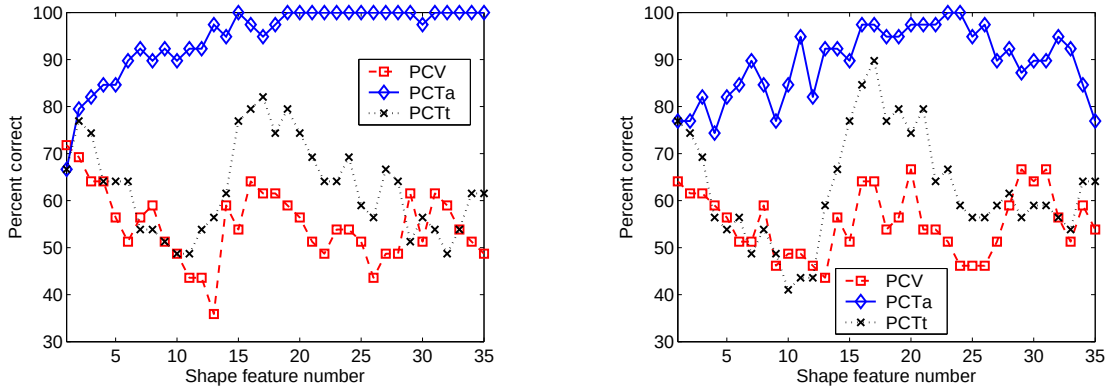
We examine two groups of subjects: (1) $\mathbf{S_{all}}$ of 35 schizophrenics and 21 controls, and (2) $\mathbf{S_{rhm}}$ of 25 schizophrenics and 14 controls, all of whom are right-handed males from $\mathbf{S_{all}}$. In each group, left and right hippocampi are studied separately. Please refer to Figure 2 for a visualization of these hippocampal shapes. We use $\mathbf{S}_X^Y$ to denote the set of Y ($\in$ {left, right}) hippocampi in $\mathbf{S}_X$, where $X \in$ {all, rhm}. Thus, there are four hipocampal data sets: $\mathbf{S}_{all}^{left}$, $\mathbf{S}_{all}^{right}$, $\mathbf{S}_{rhm}^{left}$ and $\mathbf{S}_{rhm}^{right}$.

We have seven classifiers, three feature selection schemes and four data sets. Our experiments include every combination, but due to space limitations we present only a few typical examples in detail. Although Figure 9 shows only the experimental results of applying FLD-BM and SVM-C10 classifiers to $\mathbf{S}_{all}^{left}$ and $\mathbf{S}_{rhm}^{right}$ data sets, the following observations are true for all the experiments:

1. The PCTa results show a nearly perfect classification for each classifier in the best case; however, in this case, feature selection introduces some bias, as test subjects are included in the selection process. Nevertheless, it is interesting to see that a feature subset does exist that supports nearly perfect classification.



(a) FLD-BM on S_{all}^{left} (b) SVM-C10 on S_{all}^{left}



(c) FLD-BM on S_{rhm}^{right} (d) SVM-C10 on S_{rhm}^{right}

Figure 9: Sample classification results on real data sets using different feature selection schemes. S_{all}^{left} is the set of left hippocampi in S_{all}. S_{rhm}^{right} is the set of right hippocampi in S_{rhm}. FLD-BM and SVM-C1 are two classifiers. `PCV`, `PCTa`, `PCTt` are three feature selection schemes. On each picture, the number of features used in the classification according to a certain ordering (`PCV`, `PCTa`, or `PCTt`) is plotted on the X axis, the leave-one-out cross-validation accuracy is plotted on the Y axis.

acc, fts	S_{all}		S_{rhm}	
	Left	Right	Left	Right
FLD-BM	93%, 19	79%, 2	82%, 16	82%, 17
FLD-1NN	89%, 17	79%, 11	79%, 12	87%, 17
FLD-3NN	89%, 19	79%, 2	82%, 18	87%, 17
FLD-NM	91%, 19	79%, 14	82%, 19	90%, 17
SVM-C1	80%, 19	82%, 13	69%, 12	85%, 17
SVM-C10	93%, 19	82%, 13	77%, 14	90%, 17
SVM-C100	91%, 17	82%, 15	77%, 14	90%, 17

Table 1: Best leave-one-out cross-validation accuracy using `PCTt`: Each cell shows (acc, fts), where acc is the best accuracy, and fts is the number of features used.

2. The `PCTt` results always outperform the `PCV` for each classifier in terms of the best case. The improvements range from 3% to 28% for all the cases. In `PCTt`, the classes are not separated well if there are insuficient features, while using too many introduces extra noise.

3. The performances of FLD-BM, FLD-1NN, FLD-3NN and FLD-NM are similar, and so are those of SVM-C10 and SVM-C100. However, SVM-C1 underperforms SVM-C10, which indicates the cost of constraint violation needs to be set appropriately in SVMs.

We observe that the `PCTa` results provide a kind of upper bound on the classification results – in the sense that `PCTt`, which has no knowledge of the test subject, cannot be expected to do better than `PCTa`. In addition to showing that a high level of classification accuracy is possible, the results with `PCTa` may also be instructive in determining how many dimensions are required to support perfect classification (at least in this representation), and how many seem to be too many (at least when using the SVM classifiers).

Clearly, `PCTa` cannot be used in practice, since the class of a new example is always unknown and this is the exact reason for classifying it. Thus, `PCTt` becomes a practical feature selection scheme for effective classification. In the rest of the study, we will be focusing on this scheme.

Table 1 shows the best accuracy using `PCTt` feature selection approach together with the number of features used for each case. SVM-C10 performs the best for S_{all} data set, with 93% accuracy for the left set and 82% for right. FLD-NM performs the best for S_{rhm} data set, with 82% accuracy for the left set and 90% for right. In general, FLD-NM, SVM-C10 and SVM-C100 are similar in performance. Another observation is that left hippocampi predict better in $\mathbf{S}_{all}$ while right ones predict better in $\mathbf{S}_{rhm}$. This suggests that gender and handedness may affect hippocampal shape changes in schizophrenia.

The 93% accuracy achieved for $\mathbf{S}_{all}^{left}$ greatly outperforms our previous result [29] and is competitive with the best result in previous hippocampal studies [10, 13, 15, 16, 27, 34]; please refer to Section 1 for a brief description of these studies. Note that our data set is different from the data sets in previous studies, due to a lack of shared data repositories in this domain. However, these are similar results using different techniques on similar types of data.

5 Further analyses

In this section, we present two additional analyses based on our classification framework and they are useful in medical applications. One is *receiver operating characteristic* (ROC) analysis, which trades off sensitivity (the probability patients are correctly predicted) for specificity (the probability controls are correctly predicted). This approach overcomes the problem of possible bias introduced by a fixed threshold or different size classes and is often used in visualizing the behavior of diagnostic systems [33]. The other is an approach of visualizing the discriminative pattern captured by a linear classifier to provide medical researchers with a comprehensible description of the group difference. We show these analyses using FLD-based classification, though they can also be applied to the linear SVM cases.

5.1 ROC analysis

In medical classification problems, the terms *sensitivity* and *specificity* are defined as follows: *sensitivity* is the probability of predicting disease given the true state is disease; *specificity* is the probability of predicting non-disease given the true state is non-disease. The receiver operating characteristic (ROC) curve [18, 33] is a commonly used summary for assessing the tradeoff between sensitivity and specificity. It is a plot of the sensitity versus specificity as we vary the parameters of a classification rule. In the case of a linear classifier, this can be done by setting the decision boundary at various points.

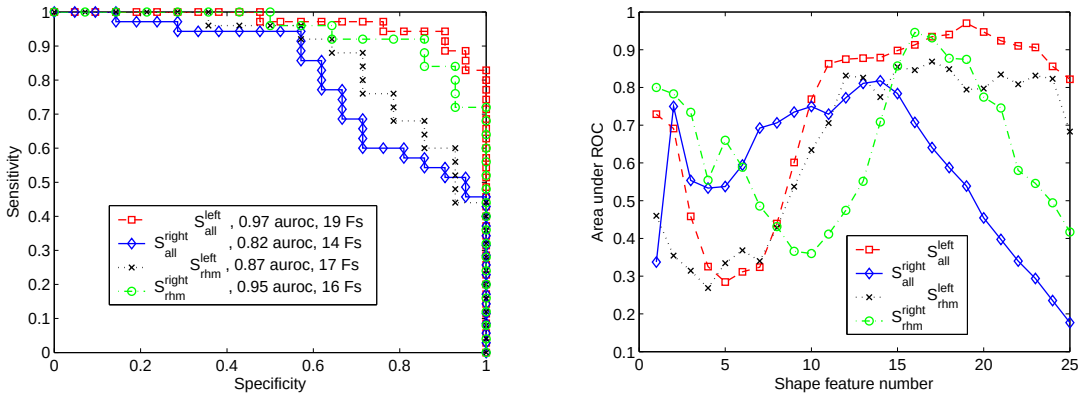
We perform ROC analysis using our PCA and FLD framework. However, in our leave-one-out experiments, each trial corresponds to an independent FLD projection. Therefore, test objects may be projected differently according to different bases, which makes the resulting scalar values incomparable across different trials. We use the following procedure to normalize these values into a standard range: for each trial, we scale and shift all the projections so that the class means for each trial's training set fall on -1 and 1, and all projections are sign-flipped, if necessary, so that the mean of the patient class is positive. Now test subject projections can then be combined, since they correspond to identically aligned training sets in the FLD space.

For each leave-one-out experiment, we calculate normalized test subject projections as described above. Given a decision threshold, a test subject is classified as a control if its normalized projection is less than the threshold; otherwise it is a patient. By varying the threshold, an ROC curve can be constructed based on the sensitivity and specificity that result at each threshold. In addition, the area under the ROC curve (AUROC) can be used as a performance measure for a classifer [3], because it is the average sensitivity over all possible specificities.

Figure 10 shows the ROC analysis results for leave-one-out cross-validation on real hippocampal data sets using the FLD classification and `PCTt` feature selection scheme. In Figure 10(a), the best ROC curves are plotted for each data set, where the number of features is selected to achieve the maximum AUROC. In Figure 10(b), the AUROC values are plotted for different experiments by varying the number of features used. Note that the ROC plot is often defined as sensitivity versus $1-$ specificity (*i.e.*, the false positive rate), dating from its origin as a signal detection technique, but for our purposes sensitivity versus specificity is equivalently useful and easier to read.

There are several advantages to using the ROC analysis as follows.

1. The ROC approach is threshold independent, which overcomes the problem of possible bias



(a) ROC curve (b) Area under ROC curve

Figure 10: ROC analysis for leave-one-out cross-validation on real hippocampal data sets using FLD classification and `PCTt` feature selection scheme. **(a)** ROC curves show the the sensitivity versus specificity as the threshold for classification is shifted. The legend shows the data set, the area under the ROC curve (AUROC) and the number of features used. **(b)** AUROC is an alternative for evaluating the performance of a classifier and is plotted on the Y axis. The number of features used in each experiment is plotted on the X axis.

introduced by a fixed threshold. It is a way to evaluate all the possible parameterizations of a classifier. Rather than using a simple heuristic or a Bayesian model to select a threshold for FLD, this approach gives the overall performance, which might be termed "discriminative power", over all the thresholds one could pick.

2. The ROC analysis inherently deals with the problem of imbalanced training sets, *e.g.*, a 5:3 ratio of patients to controls. One simple way to deal with this is to consider the accuracy of classifying patients and the accuracy of classifying controls separately, and perhaps average these two quantities to arrive at an overall accuracy estimate; the ROC approach takes this one step further and not only calculates the two accuracies independently, but does so while shifting the classification threshold over a range of values.

3. AUROC is an effective method for performance comparison between classification systems. For example, for each hippocampal data set, the best number of features selected by measuring the AUROC value in Figure 10(b) closely matches the best case shown in Table 1 by measuring the overall accuracy. The AUROC value can be thought of as an evaluation of the potential for a linear classifier to succeed on given data.

4. The ROC curve is a useful means of visualizing a classifier's performance in order to select a suitable operating point, or decision threshold. In the medical case, the cost/effect of misclassifying a patient as a normal is often higher/worse than misclassifying a normal as a patient. The ROC curve is a useful basis for minimizing misclassification cost rather than misclassification rate.

5.2 Visualization of discriminative patterns

Based on the PCA and FLD framework presented above, we introduce a method for visualizing discriminative patterns. This method shares the same idea employed by Golland *et al.* [17] for a 2D shape classification problem: for a linear classifier, the deformation showing class differences can be visualized using the normal to the separating hyperplane. Applying PCA and FLD as detailed above to a shape set, we get a discriminative value v for each shape $\mathbf{x}$:

$$v = \mathbf{x}_\delta^T * B_{pca} * B_{fld} = \mathbf{x}_\delta^T * \mathbf{w}, \tag{22}$$

where

$$\mathbf{x}_\delta = \mathbf{x} - \bar{\mathbf{x}} \tag{23}$$

is the deformation of $\mathbf{x}$ from the mean shape $\bar{\mathbf{x}}$, B_{pca} consists of a subset of eigenvectors, depending on which principal components are selected, and B_{fld} is the corresponding FLD basis. Thus $\mathbf{w}$ is a column vector that weights the contribution of each deformation element in $\mathbf{x}_\delta$ to v. Given a landmark location l, we use $\mathbf{x}_\delta(l)$ to denote the vector containing deformation fields associated with l in $\mathbf{x}_\delta$, and $\mathbf{w}(l)$ the vector of the corresponding weights in $\mathbf{w}$. Thus, the contribution made by each landmark l can be calculated as

$$C(l) = \mathbf{x}_\delta(l)^T * \mathbf{w}(l). \tag{24}$$

Based on this formula, we have two observations as follows.

1. A large magnitude of $\mathbf{w}(l)$ indicates that location l has discriminative power, since even small local deformations at this location will have a noticeable effect on the overall classification.

2. Assume Class A has more positive discriminative values v's than Class B. The vector $\mathbf{w}(l)$ actually indicates a local deformation direction towards Class A. The reason is that the location contribution $C(l) = \mathbf{x}_\delta(l)^T * \mathbf{w}(l)$ is maximized if the local deformation $\mathbf{x}_\delta(l)$ shares the same direction as $\mathbf{w}(l)$, which makes the shape more towards Class A. In contrast, $-\mathbf{w}(l)$ indicates the local deformation direction towards Class B.

We can map $\mathbf{w}(l)$ or -$\mathbf{w}(l)$ vectors onto the mean surface to show significant discriminative regions and even deformation directions towards a certain class. We note that this becomes a way of showing statistical group difference implied by the classifier model. We create such a visualization for several of our data sets using the following procedure: (1) apply PCA to all shapes in a data set; (2) order principal components using t-test over all shapes to obtain `PCTa` feature ordering; (3) apply FLD using the minimum number of features needed, according to `PCTa` ordering, to achieve a perfect discrimination between classes; (4) backproject the corresponding -$\mathbf{w}(l)$ and $\mathbf{w}(l)$ vectors onto the mean surface, and use color to code their magnitudes.

Figure 11 shows the mapping result for the synthetic rectangular surface set displayed in Figure 6, where only one feature is used for obtaining a perfect class separation. The significant region captured by the visualizaiton clearly matches our intuition on how to distinguish these two classes.

Figure 12(a) shows the result for the synthetic right hippocampus set displayed in Figure 4, where five features are used for obtaining a perfect class separation. For comparison, in Figure 12(b), we show the mappings of group mean differences $\bar{\mathbf{x}}_{C1} - \bar{\mathbf{x}}_{C2}$ and $\bar{\mathbf{x}}_{C2} - \bar{\mathbf{x}}_{C1}$ on the mean

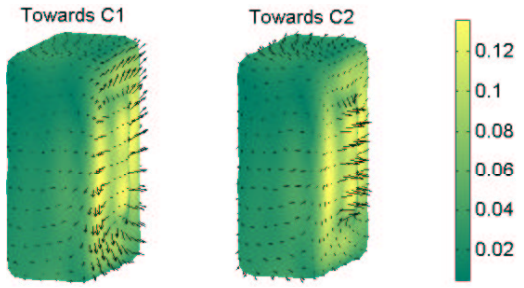


Figure 11: Discriminative patterns for S_{syn}^{rec}, the synthetic rectangular surface set displayed in Figure 6. The left plot and the right plot show the mappings of $-\mathbf{w}(l)$ and $\mathbf{w}(l)$ vectors onto the mean surfaces, suggesting deformations towards Class C1 and Class C2, respectively, since Class C2 has the more positive PCA/FLD projection. The length of each vector is scaled for better visualization. Its actual magnitude is coded in color.

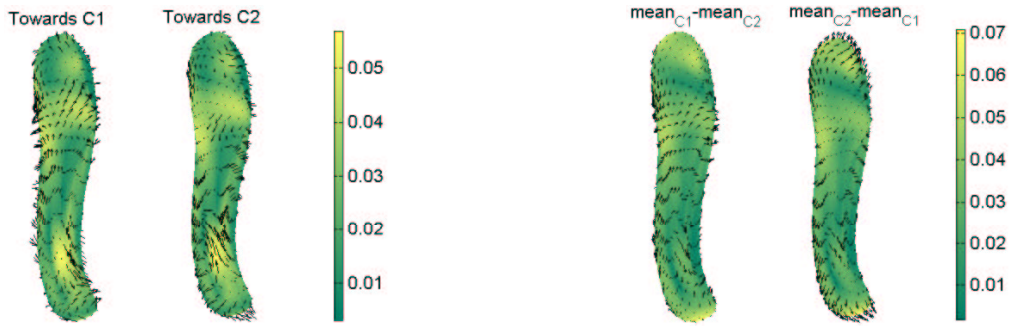
surface for the same data set, where $\bar{\mathbf{x}}_{C1}$ and $\bar{\mathbf{x}}_{C2}$ are the mean shapes of Class C1 and Class C2 in landmark representation respectively. By comparing (a) and (b), we observe that the discriminative patterns in (a) roughly capture the difference between class means in (b), which matches the intuition. In addition, (a) shows a significant discriminative (yellow/light) region in the lower middle part, while (b) shows just a small difference between class means there. By checking the data carefully, we discover the reason behind this: although this difference is small, the variance is low and the resulting discriminative power is thus fairly high.

The above results on synthetic data sets validate the effectiveness of this technique. Now we apply it to the real data sets. Figure 13 shows the results for real left and right hippocampal sets displayed in Figure 4, where 14 and 13 features are needed for obtaining perfect class separations, respectively. Mapping results show that discriminative patterns appear in the head/anterior and tail/posterior regions for both left and right hippocampi. These findings are consistent with recent reports of shape abnormality in both anterior [10, 27] and posterior regions [12, 27, 28] for hippocampi in schizophrenia. This technique visualizes statistical group difference captured by a classifer model, and can provide an intuitive, comprehensible, and useful way for visual diagonsis.

6 Computational Issues

In this section, we discuss the computation involved in the study. Figure 14 shows the major processing steps in our framework. Now we examine the computational cost for each of these steps. For most of them, we provide a time complexity measure. For some convergence procedures using iterative methods, we only report the empirical performance. Our experiments are implemented using Matlab and performed on a Dell Optiplex GX260 Pentium 4 PC with a 2.4 GHZ CPU and 512 MB of RAM, which is running WinXP Professional OS and Matlab Version 6.5.

Let n_v be the number of vertices on the square surface mesh of a volumetric object. **Surface parameterization** involves an initial parameterization and a following optimization. The initial parameterization can be done in time $O(n_v^3)$, the time required for setting up and solving n_v simultaneous linear equations with n_v unknowns [8]. The optimization is the most time consuming step in the framework. An iterative procedure is employed for achieving a local minimum, and the



(a) Discriminative patterns for S_{syn}^{hip} (b) Differences between group means for S_{syn}^{hip}

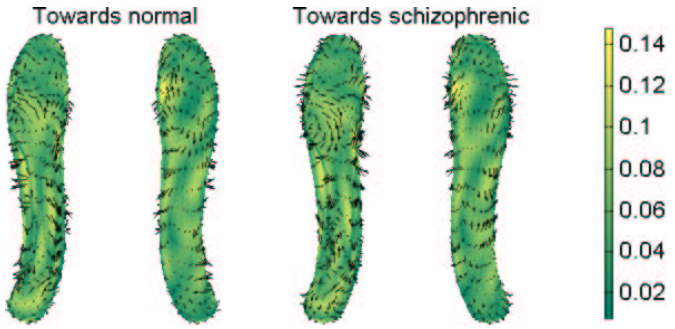
Figure 12: **(a)** Discriminative patterns for S_{syn}^{hip}, the synthetic right hippocampus set displayed in Figure 4. The left plot and the right plot show the mappings of $-\mathbf{w}(l)$ and $\mathbf{w}(l)$ vectors onto the mean surfaces, suggesting deformations towards Class C1 and Class C2, respectively, since Class C2 has the more positive PCA/FLD projection. The length of each vector is scaled for better visualization. Its actual magnitude is coded in color. **(b)** Group mean differences $\bar{\mathbf{x}}_{C1} - \bar{\mathbf{x}}_{C2}$ and $\bar{\mathbf{x}}_{C2} - \bar{\mathbf{x}}_{C1}$ are mapped onto the mean surface for S_{syn}^{hip} data set, where $\bar{\mathbf{x}}_{C1}$ and $\bar{\mathbf{x}}_{C2}$ are the mean shapes of Class C1 and Class C2 in landmark representation respectively. Again, the magnitude of each local landmark difference vector is coded in color.

number of iterations required for the convergence differs for different surfaces. Please see [4] for more details about the algorithms in this step. In our experiments, where we have $n_v = 2480 \pm 357$ (mean $\pm$ standard deviation), the initial parameterization typically can be done within 3 seconds. The typical running time for optimization ranges from 15 minutes to 3 hours, with a few worst cases of 7 – 8 hours.

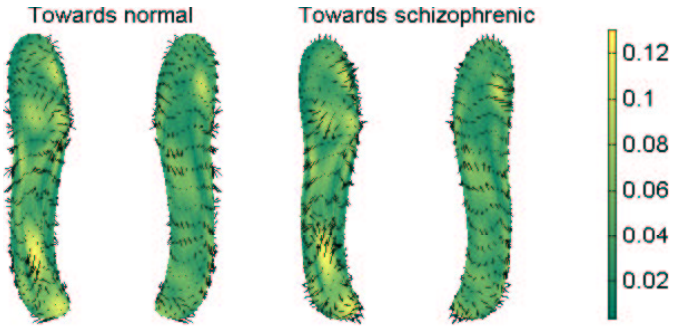
Let n_c be the number of SPHARM coefficients used in the expansion. The major computations of both **SPHARM expansion** and **SPHARM normalization** steps are to solve three overdetermined sets (for x, y and z coordinates respectively) of n_v simultaneous linear equations with n_c unknowns in a least squares fashion, where n_c is chosen to be significantly smaller than n_v for better surface fitting. Please refer to Eq. 4 and also [4] for more details. Since solving an overdetermined set of m equations with n unknowns can be done in time $O(n^2 * m)$ [8], the cost of both steps is $O(n_c^2 * n_v)$. In our experiments, we pick $n_c = 169$, and both steps can be done within 3 seconds.

Let n_s be the number of shapes in a data set, and n_l be the resolution of the landmark representation (*i.e.*, each shape has n_l landmarks). The **point distribution model** (PDM) step involves two substeps. In the first substep, conversions from SPHARM coefficients to landmarks are performed for n_s shapes, which takes $O(n_s * n_l * n_c)$ time (Eq. 4). In the second substep, PCA is performed on n_s shapes to reduce each landmark representation ($3 * n_l$ coordinates) to a feature vector ($n_s - 1$ features). The main computation is the eigenanalysis of the covariance matrix (Eq. 8 and Eq. 9), which takes time $O(n_l^3)$. In the case of $n_s \ll n_l$, the computational time for PCA can be improved to $O(n_s^2 * n_l)$, by using the Gram matrix for eigenanalysis according to [35]. In our experiments, for $n_s = 56$, $n_l = 642$ and $n_c = 169$, the whole PDM step can be done in 6 seconds.

Feature selection via t-tests computes a re-ordering of $n_s - 1$ features by running t-tests on n_s (for `PCTa` ordering) or $n_s - 1$ (for `PCTt` ordering) examplars. The running time is $O(n_s^2)$, which



(a) Discriminative patterns for S_{all}^{left}



(b) Discriminative patterns for S_{all}^{right}

Figure 13: Discriminative patterns for (a) S_{all}^{left} and (b) S_{all}^{right}, the data sets shown in Figure 2, by mapping the weight vectors $\mathbf{w}(l)$ to the mean surfaces. In each of (a-b), $-\mathbf{w}(l)$ vectors are mapped onto the first two views and indicate the directions towards a more normal shape, while $\mathbf{w}(l)$ vectors are mapped onto the last two views showing the directions towards a more schizophrenic shape. Note that the schizophrenic class has the more positive PCA/FLD projection. The length of each vector is scaled for better visualization. Its actual magnitude is coded in color. Yellow/light color indicates more discriminative power while green/dark indicates less.

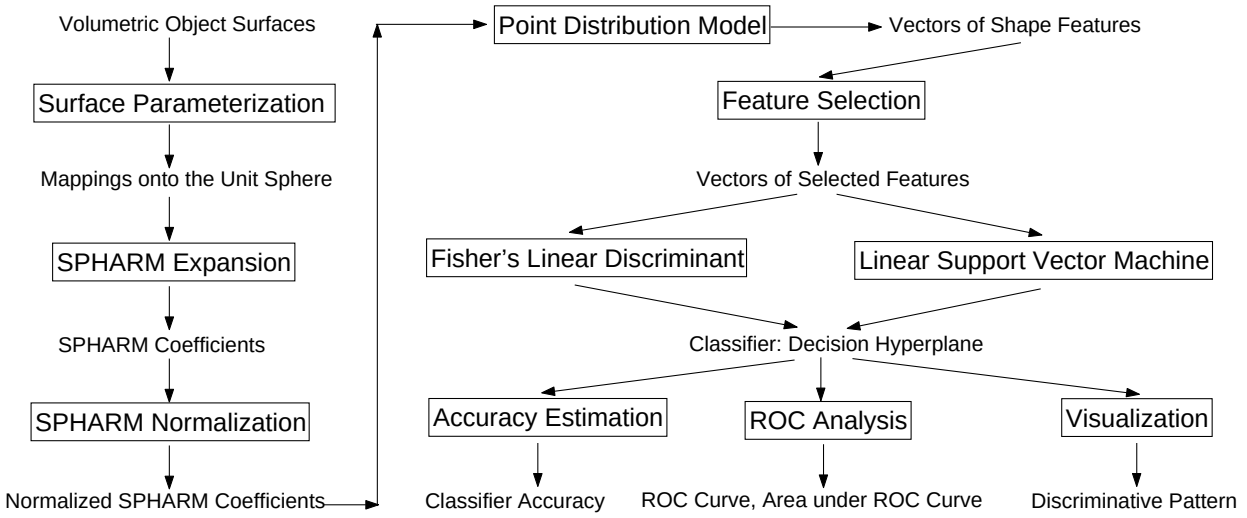


Figure 14: Major steps in the framework. Boxes refer to processing steps, while unboxed labels identify the data or results they generate.

is trivial due to a very small n_s (56 or 39) in our study.

Let n_f be the number of selected features. **Fisher's linear discriminant** (FLD) calculates the generalized eigenvectors of $n_f \times n_f$ between-class and within-class scatter matrices (Eq. 12, Eq. 13 and Eq. 14). This can be done in time $O(n_f^2 * n_s + n_f^3) = O(n_f^2 * n_s)$ for setting up the scatter matrices and solving the generalized eigenvector problem, where $O(n_s)$ shape feature vectors are involved in FLD. The **linear support vector machines** (SVM) implementation we use solves an optimization problem using quadratic programming and Lagrange multipliers, which involves an iterative procedure. The iterations required for the convergence depends on the input data and parameter settings, and so we only report its empirical performance in the next paragraph.

Accuracy estimation employs leave-one-out cross-validation, involving n_s individual training processes. In the FLD case, the total cost becomes $O(n_f^2 * n_s^2)$; and the typical running time ranges from 0.07 to 0.8 seconds, for $n_s = 56$ and $n_f \in \{1, \cdots, 53\}$, in our experiments. In the SVM case, the performance of this procedure depends on the parameter C, which specifies the cost of the constraint violation. With $n_s = 56$ and $n_f \in \{1, \cdots, 53\}$, the typical running times are 0.08 – 2.0 seconds for $C = 1$ and $C = 10$, and 0.2 – 12 seconds for $C = 100$.

ROC analysis requires normalizing leave-one-out projections in the discriminant space and computing the ROC curve and the area under the ROC curve. This can be done in time $O(n_s^2)$. **Visualization** involves backprojecting the vector normal to the separating hyperplane onto the original surface represented by n_l landmarks, which takes only $O(n_l)$ time. The costs of these final steps are trivial when compared to the earlier processing stages in the framework.

Data sets in the brain imaging domain are often relatively small due to the difficulty and expense of data collection. Thus, according to the above analysis, the computational cost is usually not a problem here, since all the above steps except *surface parameterization* are very efficient for small sample set learning and *surface parameterization* is still feasible in our case. In fact, earlier work has been done [26] on improving the efficiency of *surface parameterization*. We are also studying

more efficient and scalable approaches for parameterizing larger objects and make this framework applicable to more general cases.

7 Conclusions

This paper presents a new technique for 3D brain structure classification that combines a powerful surface modeling method with suitable pattern classification and processing techniques. The SPHARM description is chosen to model a closed 3D surface. It is a relatively new and powerful parametric surface description, which enforces surface continuity and regularization in a natural way while preserving anatomical structures and shape. Using this approach, different object surfaces can be parameterized and normalized to a common reference system to derive a detailed landmark representation comparable across objects. The choice of point distribution model for dimensionality reduction and feature extraction makes classification feasible for small sample cases and facilitates intuitive visualization.

Several linear classifiers (four FLD and three linear SVM variants) together with different feature selection approaches are employed for classification, where feature selection involves using the standard principal component ordering by variance-accounted-for as well as alternative orderings by significance as assessed using a t-test. These techniques are first validated using simulated data and then applied to real hippocampal data.

Exhaustive experimentation on hippocampal data in schizophrenia reveals that the proposed `PCTt` feature selection technique works effectively with most classifiers and improves the leave-one-out cross-validation accuracy significantly. We achieve the best accuracies of 93% for the whole set and 90% for right-handed males, competitive with the best results in similar studies using different techniques on similar types of data. Our results suggest the left hippocampus is a stronger predictor in the whole set while the right one is stronger in right-handed males.

Based on our classification framework, a threshold-free ROC analysis is also employed, where the ROC curve represents all potential discrimination performances by varying the decision criterion, and AUROC is used as an alternative to evaluate the performance of a classifier. In addition, to interpret a classifier in the original shape domain, we introduce an effective method for visualizing discriminative patterns. This approach visualizes the statistical group difference captured by a classifier model and provides an intuitive way to help doctors in visual diagnosis.

The proposed techniques can also be applied to other 3D shape classification problems in computer vision and image processing, where the involved objects are arbitrarily shaped but simply connected. Interesting future topics include (1) extending this framework to learning applications with very large data sets, and (2) developing more efficient surface parameterization techniques.

Acknowledgements

This work is supported by NSF IDM 0083423, NARSAD, NH Hospital and Ira DeCamp Foundation. We thank Hany Farid and Martin Styner for valuable discussions. We are grateful to Annette Donnelly, Laura A. Flashman, Tara L. McHugh and Molly B. Sparling for creating hippocampal traces.

References

[1] D. H. Ballard and C. M. Brown. *Computer Vision*. Prentice-Hall, Englewood Cliffs, N.J., 1982.

[2] F. L. Bookstein. Shape and the information in medical images: A decade of the morphometric synthesis. *Computer Vision and Image Understanding*, 66(2):97–118, 1997.

[3] A. P. Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.

[4] Ch. Brechbühler, G. Gerig, and O. Kubler. Parametrization of closed surfaces for 3D shape description. *Computer Vision and Image Understanding*, 61(2):154–170, 1995.

[5] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[6] C. C. Chang and C. J. Lin. *LIBSVM: a Library for Support Vector Machines*, 2001. Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

[7] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models-their training and application. *Computer Vision and Image Understanding*, 61:38–59, 1995.

[8] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. MIT Press, McGraw-Hill, New York, NY, 1990.

[9] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines (and other kernel-based learning methods)*. Cambridge University Press, Cambridge, U.K. ; New York, 2000.

[10] J. G. Csernansky, S. Joshi, L. Wang, J. W. Halleri, M. Gado, J. P. Miller, U. Grenander, and M. I. Miller. Hippocampal morphometry in schizophrenia by high dimensional brain mapping. *Proc. National Academy of Sciences USA*, 95:11406–11411, September, 1998.

[11] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification (2nd ed)*. Wiley, New York, NY, 2000.

[12] G. Gerig, M. Styner, M. Chakos, and J. A. Lieberman. Hippocampal shape alterations in schizophrenia: Results of a new methodology. In *11th Biennial Winter Workshop on Schizophrenia*, February 26, 2002.

[13] G. Gerig and M. Styner. Shape versus size: Improved understanding of the morphology of brain structures. In *Proc. MICCAI'2001: 4th International Conference on Medical Image Computing and Computer-Assisted Intervention, LNCS 2208*, pages 24–32, Utrecht, The Netherlands, October 14-17, 2001.

[14] G. Gerig. *Selected Publications*. http://www.cs.unc.edu/∼gerig/pub.html, 2003.

[15] P. Golland, B. Fischl, M. Spiridon, N. Kanwisher, R. L. Buckner, M. E. Shenton, R. Kikinis, A. Dale, and W. E. L. Grimson. Discriminative analysis for image-based studies. In *Proc. of MICCAI'2002: 5th International Conference on Medical Image Computing And Computer Assisted Intervention, LNCS 2488*, pages 508–515, Tokyo, Japan, September 25-28, 2002.

[16] P. Golland, W. E. L. Grimson, M. E. Shenton, and R. Kikinis. Small sample size learning for shape analysis of anatomical structures. In *Proc. MICCAI'2000: 3th International Conference on Medical Image Computing and Computer-Assisted Intervention, LNCS 1935*, pages 72–82, Pittsburgh, Pennsylvania, USA, October 11-14, 2000.

[17] P. Golland and W. E. L. Grimson R. Kikinis. Statistical shape analysis using fixed topology skeletons: Corpus callosum study. In *Proc. IPMI'1999: 16th International Conference on Information Processing and Medical Imaging, LNCS 1613*, pages 382–387, 1999.

[18] T. Hastie, R. Tibshirani, and J. H. Friedman. *The elements of statistical learning : data mining, inference, and prediction*. Springer series in statistics. Springer, New York, 2001.

[19] G. F. Hughes. On the mean accuracy of statistical pattern recognizers. *IEEE Transactions on Information Theory*, IT-14(1):55–63, 1968.

[20] Iowa MHCRC Image Processing Lab. *Brains Software*. http://moniz.psychiatry.uiowa.edu.

[21] S. C. Joshi, M. I. Miller, and U. Grenander. On the geometry and shape of brain submanifolds. *International Journal of Pattern Recognition and Artificial Intelligence, special issue on Magnetic Resonance Imaging*, 11(8):1317–1343, 1997.

[22] A. Kelemen, G. Szekely, and G. Gerig. Elastic model-based segmentation of 3-D neuroradiological data sets. *IEEE Transactions on Medical Imaging*, 18:828–839, 1999.

[23] J. Ma, Y. Zhao, and S. Ahalt. *OSU SVM Classifier Matlab Toolbox (ver 3.00)*. http://eewww.eng.ohio-state.edu/∼maj/osu_svm/, 2002.

[24] NIST/SEMATECHR. *e-Handbook of Statistical Methods*. http://www.itl.nist.gov/div898/handbook/, 2002.

[25] S. M. Pizer, D. S. Fritsch, P. Yushkevich, V. Johnson, and E. Chaney. Segmentation, registration, and measurement of shape variation via image object shape. *IEEE Transactions on Medical Imaging*, 18(10):851–865, 1999.

[26] M. Quicken, Ch. Brechbühler, J. Hug, H. Blattmann, and G. Székely. Parameterization of closed surfaces for parametric surface description. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2000*, volume 1, pages 354–360. IEEE Computer Society, June 2000.

[27] A. J. Saykin, L. A. Flashman, T. McHugh, C. Pietras, T. W. McAllister, A. C. Mamourian, R. Vidaver, L. Shen, J. C. Ford, L. Wang, and F. Makedon. Principal components analysis of hippocampal shape in schizophrenia. In *International Congress on Schizophrenia Research*, Colorado Springs, Colorado, USA, March 29 - April 2, 2003.

[28] M. E. Shenton, G. Gerig, R. W. McCarley, G. Szekely, and R. Kikinis. Amygdala-hippocampal shape differences in schizophrenia: the application of 3D shape models to volumetric mr data. *Psychiatry Research-Neuroimaging*, 115:15–35, August 20, 2002.

[29] L. Shen, J. Ford, F. Makedon, and A. Saykin. Hippocampal shape analysis: Surface-based representation and classification. In M. Sonka and J. M. Fitzpatrick, editors, *Medical Imaging 2003: Image Processing, Proc. of the SPIE*, volume 5032, pages 253–264, San Diego, CA, USA, February 2003.

[30] L. H. Staib and J. S. Duncan. Model-based deformable surface finding for medical images. *IEEE Transactions on Medical Imaging*, 15(5):720–731, 1996.

[31] M. Styner, G. Gerig, J. Lieberman, D. Jones, and D. Weinberger. Statistical shape analysis of neuroanatomical structures based on medial models. *Medical Image Analysis*, to appear, 2003.

[32] M. Styner, G. Gerig, S. Pizer, and S. Joshi. Automatic and robust computation of 3D medial models incorporating object variability. *International Journal of Computer Vision*, to appear, 2003.

[33] J. A. Swets and R. M. Pickett. *Evaluation of diagnostic systems : methods from signal detection theory*. Academic Press, 1982.

[34] S. J. Timoner, P. Golland, R. Kikinis, M. E. Shenton, W. E. L. Grimson, and W. M. Wells III. Performance issues in shape classification. In *Proc. MICCAI'2002: 5th International Conference on Medical Image Computing and Computer-Assisted Intervention, LNCS 2488*, pages 355–362, Tokyo, Japan, September 25-28, 2002.

[35] M. Turk and A. Pentland. Eigenfaces for recognition. *J. Cognitive Neuroscience*, 3:71–86, 1994.

[36] V. Vapnik. *Statistical Learning Theory*. John Wiley and Sons, 1998.

[37] E. W. Weisstein. *Eric Weisstein's World of Mathematics: Spherical Harmonic*. http://mathworld.wolfram.com/SphericalHarmonic.html.