1

Model Enforcement: A Unified Feature Transformation Framework For Classification And Recognition

Mohamed Kamal Omar and Mark Hasegawa-Johnson

Abstract—**Bayesian classifiers rely on models of the *a priori* and class-conditional feature distributions; the classifier is trained by optimizing these models to best represent features observed in a training corpus according to certain criterion. In many problems of interest, the true class-conditional feature probability density function (PDF) is not a member of the set of PDFs the classifier can represent. Previous research has shown that the effect of this problem may be reduced either by improving the models, or by transforming the features used in the classifier. This paper addresses this model mismatch problem in statistical identification, classification, and recognition systems. We formulate the problem as the problem of minimizing the relative entropy, also known as the Kullback-Leibler distance, between the true conditional probability density function and the hypothesized probabilistic model. Based on this formulation, we provide a computationally efficient solution to the problem based on volume-preserving maps; existing linear transform designs are shown to be special cases of the proposed solution. Using this result, we propose the symplectic maximum likelihood transform (SMLT), a non-linear volume-preserving extension of the maximum likelihood linear transform (MLLT). This approach has many applications in statistical modeling, classification, and recognition. We apply it to the maximum likelihood estimation of the joint probability density function (PDF) of order statistics and show a significant increase in the likelihood for the same number of parameters. We provide also phoneme recognition experiments that show recognition accuracy improvement compared to using the baseline Mel-Frequency Cepstrum Coefficient (MFCC) features or using MLLT. We present an iterative algorithm to jointly estimate the parameters of the symplectic map and the probabilistic model for both applications.**

I. INTRODUCTION

Given a set of realizations of a random vector and a hypothesized model of its probability density function, the purpose of this work is to find a transform of this random vector and a set of model parameters that jointly minimize an empirical estimate of the relative entropy between its true probability density function and the hypothesized model. The first stage in many pattern recognition and coding tasks is to generate a good set of features from the observed data. The set should be compact and capture all class discriminating information in the case of recognition and all information needed to reconstruct the observed data with sufficient quality in the case of coding. This set of features is usually chosen based on the available knowledge about the problem, or based on data-driven approaches to achieve compactness and discrimination goals. In both cases, the features also should satisfy the assumptions imposed on them by the recognizer or the decoder.

Statistical pattern recognition and classification systems are based on the assumption that the conditional probability density functions of the features can be approximated. Many probabilistic models in statistical recognition and classification systems approximate the features' joint PDF by a Gaussian PDF or a mixture of Gaussian PDFs. Since the measurements are not necessarily jointly normal, power transforms are used in statistical analysis to get features that satisfy the normality assumption better [1]. Moreover, in many high-dimensional applications, the values of the correlation between different features are ignored. This is achieved by assuming that the observations are conditionally independent given some intermediate class label (e.g., given the Gaussian component label in a diagonal-covariance Gaussian mixture model [2], or given the class label in a naive Bayes classifier [3]). The computational efficiency requirements often motivate this assumption, although it is known to be unjustified in many applications of interest, e.g., in speech [4], image [5], and text [3] applications. This makes the problem of finding the features that are best represented with these models equivalent to the problem of finding the conditionally independent components of the original features for each one of these intermediate class labels. Previous approaches to this problem formulated it as a redundancy reduction problem that can be solved by using a more relaxed model or by using a linear transform of the data. In [6], we formulated the problem as a non-linear independent component analysis (NICA) problem. We showed that using the features generated using NICA in speech recognition increased the phoneme recognition accuracy compared to the baseline system and compared to systems that used linear transforms like linear ICA [7], linear discriminant analysis (LDA) [8], and maximum likelihood linear transform (MLLT) [9]. We showed also that the NICA algorithm described in [6] can be formulated as a generalization of the MLLT.

In this work, we will introduce a unified information-theoretic approach to feature transformation that makes no assumptions about the true probability density function of the original data and can be applied for any probabilistic model with arbitrary constraints. Both power transforms and redundancy reduction approaches can be formulated as special cases of what we call a model enforcement approach: the model enforcement approach estimates a non-linear transform and the parameters of the probabilistic model that jointly minimize the relative entropy between the true joint feature PDF and its hypothesized model. In the next section, we will give a very brief introduction to statistical modeling for classification and recognition. We describe the main previous approaches to feature trans-

forms and their limitations in section III. An information-theoretic formulation of the problem is described in section IV. An iterative algorithm is described in section V to jointly estimate the parameters of the transform of the features and the parameters of the model. Then, experiments based on an efficient implementation of this algorithm are described in section VI. Finally, section VII provides discussion of the results and a summary of this work. In this paper, a subscript is used as an index of a component of a random vector, and a superscript is used as an index of a realization of the random vector. Capital letters are used to denote the random variables and the corresponding small letters to denote their realizations. Both vectors and matrices are in boldface to be distinguished from scalars.

II. Parametric Approach For Statistical Modeling

Bayes rule is the optimal classification rule if the underlying distribution of the data is known. In practice, we do not know the underlying distribution. There are two main approaches to this problem: parametric and non-parametric [8]. In non-parametric approaches like kernel-based approaches the decision boundaries between the classes are estimated directly instead of trying to estimate the conditional density of the classes while parametric approaches estimate a parametric model of the conditional PDFs. In this paper, we will limit our discussion to the parametric approaches.

In parametric statistical modeling for classification and recognition, a probabilistic model is chosen and its parameters are trained to optimize a certain criterion under the assumption that the true PDF of the features can be approximated well by the model. Parameter optimization takes place without questioning the validity of this assumption. Since the features are usually chosen based on prior knowledge about the task using heuristic approaches, this assumption is in most cases unjustified.

Information theory provides a measure by which we can say how well a PDF is approximated by another PDF [10]. This measure is called the divergence, Kullback-Leibler distance, or the relative entropy and is defined by

$$R(P,\hat{P}) \;=\; E_P\left[\log\left(\frac{P}{\hat{P}}\right)\right], \tag{1}$$

where P is the true PDF and $\hat{P}$ is the approximate PDF. An important property of the relative entropy is that

$$R(P,\hat{P}) \geq 0$$

with equality if and only if

$$\hat{P} = P$$

Most parametric statistical classification systems use maximum likelihood estimation (MLE) or Bayesian methods to estimate the parameters of the model. The popularity of MLE is attributed to the existence of efficient algorithms to implement it, like the expectation-maximization (EM) algorithm, and to its consistency and asymptotic efficiency, if the true PDF belongs to the admissible set of parameterized PDF models [11].

In the MLE method, the parameters, λ^*, are estimated given a set of i.i.d observations, $\{\mathbf{x}^i\}_{i=1}^N$, by maximizing the functional

$$L_{emp} \;=\; \sum_{i=1}^{N} \log \hat{P}(\mathbf{x}^i,\lambda), \tag{2}$$

with respect to the parameters λ.

Maximizing this empirical functional is equivalent to minimizing an empirical estimate of the relative entropy between the true PDF and the hypothesized PDF model

$$R_{emp}(P,\hat{P}) \;=\; -H(\mathbf{x}) - \frac{1}{N}\sum_{i=1}^{N} \log \hat{P}(\mathbf{x}^i,\lambda), \tag{3}$$

where $H(\mathbf{x})$ is the differential entropy of the random vector $\mathbf{x}$.

Vapnik and Chervonenkis show that the necessary and sufficient condition of the consistency of this maximization problem is, [12],

$$Pr\left(\sup_{\lambda\in\mathbf{\Lambda}} |R(P,\hat{P}) - R_{emp}(P,\hat{P})| \geq \epsilon\right) \to 0 \qquad \text{for } N\to\infty \text{ and } \forall\epsilon>0, \tag{4}$$

where $\{\mathbf{x}^i\}_{i=1}^N$ are generated by any admissible PDF $\hat{P}(\mathbf{x},\lambda_0)$, $\forall\lambda_0\in\mathbf{\Lambda}$.

However, as we do not know the true PDF, we can not guarantee small approximation error. A small approximation error can be achieved by using a complex structure of the hypothesized models that can approximate a large set of PDFs. On the other hand, this increases the computational and conceptual complexity of the system, and increases the required amount of training data to get a good estimate of the model parameters.

An important property of any classification or recognition model that is related to consistency is its generalization ability. The generalization ability is a monotonically increasing function of the ratio of the number of available training vectors and the VC dimension of the family of the hypothesized PDFs [12]. This means that the requirements of generalization ability conflict with the requirements of decreasing the approximation error.

One way of controlling the number of parameters is by using a relatively simple probabilistic model, and a transform of the observation vector to a new feature vector whose PDF is better modeled by the hypothesized PDF based on certain criterion. Many previous approaches to feature transformation show improvement in classification and recognition accuracy compared to using more complex probabilistic models for the same number of parameters [9], [5], and [6]. Most of these methods, as will be discussed in next section, are linear transformations that use a

number of parameters equal to the square of the dimension of the feature vector. Our approach provides a generalization to non-linear transformations that is more flexible in selecting the number of the parameters of the transform, as it is linear in the dimension of the input features.

III. Transformations Of Multivariate Data

Many important results in statistical analysis and pattern recognition follow from the assumption that the population being sampled or investigated is either normally distributed or conditionally independent given the class label. For this reason, many methods have been proposed that transform the measurements to better satisfy the assumptions of normality and/or conditional independence. In this section, we will give very brief examples of previous approaches to transform multivariate data such that these assumptions are better satisfied.

A. Transformations To Approximate Normality

The assumption of normality that most statistical analysis approaches are based on is seriously violated in many interesting problems. A frequently discussed solution in the statistical literature is to transform the original measurements to features that better satisfy the normality assumption. The transformation may be based on theoretical considerations or use a data-driven approach. Univariate examples of the former type are the logistic transformation for binary data [13], and the variance stabilizing transformations for the binomial, the Poisson, and the correlation coefficient [14].

There are many examples of data-driven transformations. Tukey introduced a family of power transformations such that the transformed values are a monotonic function of the observations over some admissible range for univariate analysis [15]. This family was modified in [16], where maximum likelihood and Bayesian methods were used to estimate the transformation parameter. These power transforms were extended to the multivariate case by using a number of scalar transforms equal to the dimension of the observation vector in [1]. Conceptual and computational simplicity were the main reasons to limit the suggested transforms to a family of power transforms.

B. Transformations For Redundancy Reduction

As described in section I, one way of controlling the number of parameters of the probabilistic model in classification and recognition systems is by assuming that the features are decorrelated or independent. This degrades the performance of the system, if the features used in the statistical recognition or classification system do not satisfy this assumption. Recent approaches to address this problem can be classified into two major categories. The first category tries to decrease the number of parameters required for a full account of the features' interdependence by tying the parameters of the class-conditional probabilistic models. In other words, this category tries to reduce the redundancy in the model parameters. In the second category, the original feature space is transformed to a new feature space that better satisfies the assumption of conditional independence or decorrelation. In other words, this category tries to decrease the redundancy in the features themselves. Methods in this category typically use linear transforms, and therefore typically require a number of trainable parameters equal to the dimension of the input feature vector times the dimension of the output feature vector. In the following, we will describe examples of redundancy reduction based on feature transformation.

B.1 Principal Component Analysis

Principal component analysis [8], and the closely related Karhunen-Loève transform are classic techniques in statistical data analysis, feature extraction, and data compression. Given a random vector $\mathbf{x}$ and a number of observations from this random vector, no explicit assumptions on the probability density of the vectors are made in PCA, as long as the first- and second-order statistics can be estimated from the observed data. Also, no generative model is assumed for the vector $\mathbf{x}$, but there are extensions to PCA like probabilistic principal component analysis (PPCA) [17], and like the sensible PCA (SPCA) approach in [18] that associate a generative model with PCA. The PCA transform is constructed from the eigenvectors of the sample covariance matrix with maximum corresponding eigenvalues. This transform is the unique unitary transform of a given dimension such that the elements of the output vector are uncorrelated and its variance is maximized.

Since there are many sources of variability in most real-life signals and some of them are irrelevant to the classification or recognition task, selecting the direction of maximum variance for projection does not always minimize the recognition error. Therefore, PCA is sometimes used in these tasks to calculate the principal components of the Fisher covariance matrix of the classes

$$S_{wb} = W^{-1}B, \tag{5}$$

where W is the within-class scatter matrix, and B is the between-class scatter matrix [8]. This approach is called linear discriminant analysis (LDA).

Campbell [19] has shown that linear discriminant analysis is related to the maximum likelihood estimation of parameters for a Gaussian model, with *a priori* assumptions on the structure of the model. Hastie and Tibshirani [20] further generalized this result by assuming that class distributions are a mixture of Gaussians. Kumar [21] generalized LDA to the case of classes with different covariance matrices and referred to this generalization as heteroscedastic discriminant analysis (HDA). HDA can be formulated as a maximum likelihood estimation problem for normal populations with common covariance matrix in the rejected subspace.

B.2 Independent Component Analysis

ICA defines a generative model for the observed multivariate data [22], [7]. These data are typically given as a

large database of samples. In the model, the data variables are assumed to be linear mixtures of some unknown latent variables, and the mixing system is also unknown. The latent variables are assumed non Gaussian and mutually independent, and they are called the independent components of the observed data. These independent components, also called sources or factors, can be found by ICA.

ICA can be seen as an extension to principal component analysis and factor analysis. The goal of ICA is to estimate the independent sources and the mixing coefficients given only observations that are a linear mixture of the latent independent source signals. In contrast to PCA, ICA not only decorrelates the sources but also reduces higher-order statistical dependencies, attempting to make the components as independent as possible.

There are many approaches to solving the ICA problem, including information maximization, maximum likelihood estimation, negentropy maximization, higher-order moments and cumulants approximations of differential entropy, and nonlinear PCA. In [23], it is shown that all these different approaches lead to the same iterative learning algorithm.

The maximum likelihood linear transform (MLLT) introduced in [9] can be formulated as a maximum likelihood ICA for data generated by each Gaussian component of a Gaussian mixture model [6]. MLLT estimates the parameters of a linear transform in order to maximize the likelihood of the training data given a diagonal-covariance Gaussian mixture model; the transformed features are better represented by the model than the original features. This is motivated by the fact that the diagonal covariance models impose a constraint on the likelihood of the features which results in underestimating its value.

As stated before, ICA algorithms assume that the components are mixed linearly to generate the observation data. However, in many interesting applications, this assumption is unjustified or unacceptable. In [6], an extension of the ICA algorithms to nonlinearly mixed sources was used to reduce the redundancy of the features used in speech recognition.

C. Limitations of Previous Approaches

Transformations to achieve normality were constrained to using a restricted family of power transforms and to a Gaussian hypothesized model. These transforms were scalar transforms, i.e. each transformed feature is obtained from a single input measurement.

An important limitation of existing feature-based redundancy reduction approaches is the assumption that a linear transform of the features is enough to satisfy the model. For example, ICA algorithms assume that the factors are mixed linearly to generate the observation vectors. In many interesting applications, this assumption is unjustified or unacceptable. An example is the speech recognition problem, as all acoustic features used in speech recognition can not be modeled as a linear mixture of independent sources of variations in the speech signal. In image and face recognition also, there are deformations like bending which result in correlations that can not be compensated for by a linear transform. In case of Gaussian or mixture of Gaussian hypothesized PDF, this sufficiency of linear transformation assumption is equivalent to assuming that the true conditional joint PDFs of the features are Gaussian or mixture of Gaussian PDFs respectively. This is due to the fact that any linear transformation of a Gaussian random vector results in a Gaussian random vector. This limitation was alleviated in the non-linear independent component analysis approach proposed in [6]. However, the statistical independence constraint is only one of many possible constraints that may be imposed on the probabilistic models used in classification and recognition systems.

IV. A Unified Information-Theoretic Approach to Model Enforcement

The goal of this section is to generalize feature transformation in two ways. First, we will provide a feature transformation framework that makes no assumptions about the probabilistic model and the constraints imposed on it. This provides us with the flexibility needed to address problems in which the model is not necessarily Gaussian and does not assume the features are uncorrelated or independent, but assumes a certain parametric form of the features' conditional PDFs. Second,we will provide a non-linear transform, as opposed to previous linear transforms, that is based on this framework. This non-linear transform is a vector-based transform, as opposed to previous scalar power transforms. The number of parameters of this transform is linear in the dimension of the input feature vector, while it is quadratic for linear transforms. We will show also how all previous transforms to normality and redundancy reduction approaches discussed in section III are special cases of the information-theoretic model enforcement approach proposed here.

A. Problem Formulation

Motivated by the discussion of the previous sections, we will choose any hypothesized parametric family of distributions to be used in our probabilistic model, and search for a map of the features that improves the validity of our model. To do that, we will need the following theorem.

Theorem 1: Let $\mathbf{y} = f(\mathbf{x})$ be an arbitrary one-to-one map of the random vector $\mathbf{x}$ in $\Re^n$ to $\mathbf{y}$ in $\Re^n$, and let $\hat{P}_\Lambda(\mathbf{Y})$ be a hypothesized parametric family of density functions. The map $f^*(.)$ and the set of parameters $\mathbf{\Lambda}^*$ minimize the relative entropy between the hypothesized and the true PDFs of $\mathbf{y}$ if and only if they also maximize the objective function

$$V = E_{P(\mathbf{y})}\left[\log\left(|det\left(\mathbf{J}_f\right)|\right) + \log \hat{P}_\Lambda(\mathbf{y})\right], \qquad (6)$$

where $\mathbf{J}_f$ is the Jacobian matrix of the map $f(.)$.

Proof:

We will rewrite the expression for the relative entropy after an arbitrary transformation, $\mathbf{y} = f(\mathbf{x})$, of the input

random vector $\mathbf{x}$ in $\Re^n$, as

$$R(P(\mathbf{y}), \hat{P}(\mathbf{y})) \;=\; -H(P(\mathbf{y})) - E_{P(\mathbf{y})}\left[\log\left(\hat{P}(\mathbf{y})\right)\right], \quad (7)$$

where $H(P(\mathbf{y}))$ is the differential entropy of the random vector $\mathbf{y}$ based on its true PDF $P(\mathbf{y})$.

The relation between the output differential entropy and the input differential entropy is in general [24],

$$H(P(\mathbf{y})) \;\leq\; H(P(\mathbf{x})) + \int_{\Re^n} P(\mathbf{x}) \log\left(|det\left(\mathbf{J}_f\right)|\right) d\mathbf{x}, \quad (8)$$

where $P(\mathbf{x})$ is the probability density function of the random vector $\mathbf{x}$, for an arbitrary transformation, $\mathbf{y} = f(\mathbf{x})$, of the random vector $\mathbf{x}$ in $\Re^n$, with equality if $f(\mathbf{x})$ is invertible.

Therefore the relative entropy can be written as

$$\begin{aligned} R(P(\mathbf{y}), \hat{P}(\mathbf{y})) \;=\; & -H(P(\mathbf{x})) - E_{P(\mathbf{x})}\left[\log\left(|det\left(\mathbf{J}_f\right)|\right)\right] \\ & -E_{P(\mathbf{y})}\left[\log \hat{P}(\mathbf{y})\right], \end{aligned} \quad (9)$$

for an invertible map $\mathbf{y} = f(\mathbf{x})$.

The expectation of a function $g(\mathbf{x})$ for an arbitrary one-to-one map $\mathbf{y} = f(\mathbf{x})$ can be written as [24],

$$E_{P(\mathbf{x})}\left[g(\mathbf{x})\right] \;=\; E_{P(\mathbf{y})}\left[g(f^{-1}(\mathbf{y}))\right], \quad (10)$$

where $f^{-1}(.)$ is the inverse map.

Therefore

$$\begin{aligned} R(P(\mathbf{y}), \hat{P}(\mathbf{y})) \;=\; & -H(P(\mathbf{x})) \\ & -E_{P(\mathbf{y})}\left[\log\left(|det\left(\mathbf{J}_f\right)|\right) + \log \hat{P}(\mathbf{y})\right]. \end{aligned} \quad (11)$$

Equation 11 proves the theorem. ■

Theorem 1 states that minimizing the relative entropy is equivalent to maximizing the sum of the expected log likelihood and a cost function; the cost function is determined by the determinant of the Jacobian matrix of the transform. This cost function guarantees that maximizing the likelihood of the transformed features will not be at the expense of their information content measured by their differential entropy.

B. A Maximum Likelihood Approach to Model Enforcement

For a nonlinear feature transformation, the Jacobian matrix of the transformation is a function of the values of the feature vectors. This makes the maximization of the objective function for a high-dimensional input feature vector computationally expensive. A significant reduction in the computational complexity is achieved by an important special case. This special case that reduces the problem to maximum likelihood estimation (MLE) of the model and map parameters is given in the following lemma, but first we need to define volume-preserving maps in $\Re^n$, where n is an arbitrary positive integer.

Definition: A C^∞ map $f : S_{\mathbf{x}} \rightarrow S_{\mathbf{y}}$ where $S_{\mathbf{x}} \subset \Re^n$ and $S_{\mathbf{y}} \subset \Re^n$ is said to be volume-preserving if and only if $|det\left(\mathbf{J}_f\right)| = 1 \;\forall \mathbf{x} \in S_{\mathbf{x}}$.

Lemma: Let $\mathbf{y} = f(\mathbf{x})$ be an arbitrary one-to-one C^∞ volume-preserving map of the random vector $\mathbf{x}$ in $\Re^n$ to $\mathbf{y}$ in $\Re^n$, and let $\hat{P}_\Lambda(\mathbf{y})$ be a hypothesized parametric family of density functions. The map $f^*(.)$ and the set of parameters Λ^* jointly minimize the relative entropy between the hypothesized and the true PDFs of $\mathbf{y}$ if and only if they also maximize the expected log likelihood based on the hypothesized PDF.

Using the definition of the volume-preserving maps, the proof of the lemma is straightforward. The Lemma proves that the maximum likelihood criterion used in MLLT is the appropriate criterion for any volume-preserving transform. By reducing the problem to MLE problem, efficient algorithms based on the incremental EM algorithm can be designed [25].

C. Generality of Model Enforcement Approach

Theorem 1 generalizes previous approaches in two ways. First, transforms can be designed to satisfy arbitrary constraints on the hypothesized PDF, not necessarily those that impose an independence or decorrelation constraint on the features. Second, it can also be applied to any parameterized probabilistic model not necessarily Gaussian. To show the generality of theorem 1 and its wide range of applications, we relate it with previous methods.

Transformations to normality described in section II are a special case of theorem 1 by constraining the PDF model to be Gaussian and the transform to be a power transform.

PCA may be viewed as a special case of theorem 1 under two equivalent constraints. First, if the transform is constrained to be linear and the model PDF is constrained to be a diagonal-covariance Gaussian, then theorem 1 reduces to PCA. Equivalently, if the true feature PDF is assumed to be Gaussian, and the model PDF is constrained to be a diagonal-covariance Gaussian, theorem 1 reduces to PCA. Probabilistic PCA (PPCA) is a generalization of PCA that can be shown as an application of theorem 1 when the hypothesized model of the joint PDF is not necessarily Gaussian.

ICA also can be shown as a special case of theorem 1 when the hypothesized model assumes statistical independence of the transformed features and the transform is constrained to be linear. Nonlinear ICA removes the constraint that the transform must be linear. Factor analysis is also a special case of theorem 1 by assuming that the hypothesized joint PDF is Gaussian with special covariance structure.

MLLT is a special case of theorem 1 by using a linear volume-preserving map of the features and assuming the hypothesized joint PDF is Gaussian or a mixture of Gaussians. As we highlighted before, these two assumptions of linearity and Gaussianity together are equivalent to the assumption that the original features are Gaussian.

It should be noted that all linear maps designed to improve the satisfaction of the features of a given model are

special cases of the lemma, as any linear map is equivalent to a linear volume-preserving map multiplied by a scalar.

V. Implementation of the Maximum Likelihood Approach

In the previous section, we showed that by using a volume-preserving map, the model enforcement problem is reduced to maximizing the likelihood of the output components. This section therefore develops a maximum likelihood volume-preserving nonlinear transform algorithm. The resulting algorithm may be considered a nonlinear generalization of MLLT with a more flexible parameter count than MLLT; experiments in section VI show that the algorithm outperforms MLLT with fewer trainable parameters. In this section, we use a volume-preserving map to generate the new set of features. The maximum likelihood approach using volume-preserving maps is a good compromise between the two extremes of previous linear approaches with their simplicity and computational efficiency but inadequacy in many applications, and the nonlinear approaches with their generality but computational complexity associated with calculating the determinant of the Jacobian matrix.

A. Symplectic Maps

Symplectic maps are volume-preserving maps that can be represented by scalar functions [26]. This very interesting result allows us to jointly optimize the parameters of the symplectic map and the model parameters using the EM algorithm or one of its incremental forms [25].

Let $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2)$, and $\mathbf{y} = (\mathbf{y}_1, \mathbf{y}_2)$, with $\mathbf{x}_1, \mathbf{x}_2, \mathbf{y}_1, \mathbf{y}_2 \in \Re^{\frac{n}{2}}$, then any reflecting symplectic map can be represented by

$$\mathbf{y}_1 = \mathbf{x}_1 - \frac{\partial V(\mathbf{x}_2)}{\partial \mathbf{x}_2}, \tag{12}$$

$$\mathbf{y}_2 = \mathbf{x}_2 - \frac{\partial T(\mathbf{y}_1)}{\partial \mathbf{y}_1}, \tag{13}$$

where $V(\cdot)$ and $T(\cdot)$ are two arbitrary scalar functions [27]. We use two multi-layer feed-forward neural networks to get a good approximation of these scalar functions [28].

$$V(\mathbf{u}, \mathbf{A}, \mathbf{C}) = \sum_{j=1}^{H} c_j S(\mathbf{a}_j \mathbf{u}), \tag{14}$$

$$T(\mathbf{u}, \mathbf{B}, \mathbf{D}) = \sum_{j=1}^{H} d_j S(\mathbf{b}_j \mathbf{u}), \tag{15}$$

where $S(.)$ is a nonlinear function like sigmoid or hyperbolic tangent, $\mathbf{a}_j$ is the jth row of the $H \times \frac{n}{2}$ matrix $\mathbf{A}$, and c_j is the jth element of the $H \times 1$ vector $\mathbf{C}$, $\mathbf{b}_j$ is the jth row of the $H \times \frac{n}{2}$ matrix $\mathbf{B}$, and d_j is the jth element of the $H \times 1$ vector $\mathbf{D}$. The parameters of these two neural networks and the parameters of the model are jointly optimized to maximize the likelihood of the training data.

B. Joint Optimization of The Map and Model Parameters

We will explain in this section how the parameters of the volume-preserving map and the probabilistic model can be jointly optimized to maximize the likelihood of the estimated features. We will assume that the system is an HMM-based recognizer [29]. However, this approach can be applied to any statistical classification, detection, or recognition system for which a set of hidden variables can be defined. We will assume also that the scalar functions in the symplectic map are represented by three-layer feed-forward neural networks (NN) with the nonlinearity in the NNs represented by hyperbolic tangent functions, and therefore,

$$\mathbf{y}_1 = \mathbf{x}_1 - \sum_{j=1}^{H} c_j \mathbf{a}_j^T [1 - S^2(\mathbf{a}_j \mathbf{x}_2)], \tag{16}$$

$$\mathbf{y}_2 = \mathbf{x}_2 - \sum_{j=1}^{H} d_j \mathbf{b}_j^T [1 - S^2(\mathbf{b}_j \mathbf{y}_1)]. \tag{17}$$

The derivation for any other non-linear function is a straightforward replication of the derivation provided here.

Define $\mathbf{\Phi}^k = (\mathbf{\Lambda}^k, \mathbf{W}^k)$ to be the set of the recognizer parameters, $\mathbf{\Lambda}^k$, and the symplectic parameters, $\mathbf{W}^k$, at iteration k of the algorithm. Using the EM algorithm, the auxiliary function [30] to be maximized, with respect to $\mathbf{\Phi}^{k+1}$, is

$$Q(\mathbf{\Phi^k}, \mathbf{\Phi^{k+1}}) = E_{\hat{P}(\xi|\mathbf{Y}, \mathbf{\Phi}^k)}[\log \hat{P}(\mathbf{y}, \zeta|\mathbf{\Phi}^{k+1})|\mathbf{y}, \mathbf{\Phi}^k], \tag{18}$$

where $\zeta \in \xi$ is the state sequence corresponding to the sequence of observations $\mathbf{x} \in \Re^{n \times T}$ that are transformed to the sequence $\mathbf{y} \in \Re^{n \times T}$, and T is the sequence length in frames. In this case, the hidden variables for the EM algorithm are the HMM states, ζ^t for $1 \leq t \leq T$, and the complete data is the set of features and HMM states,$(\mathbf{y}^t, \zeta^t)$, at each instance t. The transformed features $\mathbf{y}^t$ are observable variables as they are obtained from the observed feature vector $\mathbf{x}^t$ by an invertible transformation $\mathbf{y}^t = f(\mathbf{x}^t)$. The auxiliary function can be written as

$$Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1}) = \sum_{\zeta \in \xi} \frac{\hat{P}(\mathbf{y}, \zeta|\mathbf{\Phi}^k)}{\hat{P}(\mathbf{y}|\mathbf{\Phi}^k)} \log \hat{P}(\mathbf{y}, \zeta|\mathbf{\Phi}^{k+1}). \tag{19}$$

Given a particular state sequence ζ, $\hat{P}(\mathbf{y}, \zeta|\mathbf{\Phi^k})$ can be written as

$$\hat{P}(\mathbf{y}, \zeta|\mathbf{\Phi}^k) = \pi_{\zeta^0} \prod_{t=1}^{T} \hat{P}(\zeta^t|\zeta^{t-1}, \mathbf{\Phi}^k) \hat{P}(\mathbf{y}^t|\zeta^t, \mathbf{\Phi}^k), \tag{20}$$

where π_{ζ^0} is the probability of starting the sequence in state ζ^0, $\hat{P}(\zeta^t|\zeta^{t-1}, \mathbf{\Phi}^k)$ is the state transition probability from ζ^{t-1} to ζ^t given the current parameters $\mathbf{\Phi}^k$, and

$\hat{P}(\mathbf{y}^t|\zeta^t, \mathbf{\Phi}^k)$ is the probability of the observation vector $\mathbf{y}^t \in \Re^n$ given the state ζ^t and the current parameters $\mathbf{\Phi}^k$.

Then, the auxiliary function becomes

$$\begin{aligned} Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1}) &= \sum_{\zeta \in \xi} \frac{\hat{P}(\mathbf{y}, \zeta|\mathbf{\Phi}^k)}{\hat{P}(\mathbf{y}|\mathbf{\Phi}^k)} \\ &\left(\log \pi_{\zeta^0} + \sum_{t=1}^{T} \log \hat{P}(\zeta^t|\zeta^{t-1}, \mathbf{\Phi}^{k+1}) \right) \\ &+ \sum_{\zeta \in \xi} \frac{\hat{P}(\mathbf{y}, \zeta|\mathbf{\Phi}^k)}{\hat{P}(\mathbf{y}|\mathbf{\Phi}^k)} \log \hat{P}(\mathbf{y}^t|\zeta^t, \mathbf{\Phi}^{k+1}). \end{aligned} \tag{21}$$

The updating equations for the HMM parameters based on this formulation are the same as mentioned in [2], and therefore will not be derived here. To calculate the updating equations of the symplectic parameters, we note that

$$\begin{aligned} &\sum_{\zeta \in \xi} \frac{\hat{P}(\mathbf{y}, \zeta|\mathbf{\Phi}^k)}{\hat{P}(\mathbf{y}|\mathbf{\Phi}^k)} \sum_{t=1}^{T} \log \hat{P}(\mathbf{y}^t|\zeta^t, \mathbf{\Phi}^{k+1}) \\ &= \sum_{l=1}^{L} \sum_{t=1}^{T} \frac{\hat{P}(\mathbf{y}, \zeta^t = l|\mathbf{\Phi^k})}{\hat{P}(\mathbf{y}|\mathbf{\Phi^k})} \log \hat{P}(\mathbf{y}^t|\zeta^t = l, \mathbf{\Phi}^{k+1}), \end{aligned} \tag{22}$$

where L is the total number of states.

Therefore, the derivative of the auxiliary function with respect to y_j for $j = 1, 2, \cdots, n$ is given by

$$\begin{aligned} \frac{\partial Q(\mathbf{\Phi^k}, \mathbf{\Phi^{k+1}})}{\partial y_j} &= \sum_{l=1}^{L} \sum_{t=1}^{T} \frac{\hat{P}(\mathbf{y}, \zeta^t = l|\mathbf{\Phi}^k)}{\hat{P}(\mathbf{y}|\mathbf{\Phi}^k)} \\ &\frac{\partial \log \hat{P}(\mathbf{y}^t|\zeta^t = l, \mathbf{\Phi}^{k+1})}{\partial y_j}. \end{aligned} \tag{23}$$

If a mixture of densities is used to model each state, then the derivative of the auxiliary function becomes

$$\begin{aligned} \frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial y_j} &= \sum_{l=1}^{L} \sum_{m=1}^{K_l} \sum_{t=1}^{T} \frac{\hat{P}(\mathbf{y}, \zeta^t = l, \rho_{\zeta^t t} = m|\mathbf{\Phi}^k)}{\hat{P}(\mathbf{y}|\mathbf{\Phi}^k)} \\ &\frac{\partial \log \hat{P}(\mathbf{y}^t|\zeta^t = l, \rho_{\zeta^t t} = m, \mathbf{\Phi}^{k+1})}{\partial y_j}, \end{aligned} \tag{24}$$

where $\rho_{\zeta^t t}$ is the mixture component at time t in the mixture of the state ζ^t, and K_l is the number of densities in each mixture.

These equations are written for one input sequence of observations, and a summation over all training patterns, i.e. sequences of observations, is excluded to simplify the equations. Since the update equations for the symplectic parameters do not need to explicitly mention the structure of the recognizer, we will merge the summation over all states and densities to a summation over densities. These reductions are only to improve the tractability of the following equations and have no effect on the derivation. After modifying the notation,

$$\begin{aligned} \frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial y_j} &= \sum_{t=1}^{T} \sum_{m=1}^{K} \frac{\hat{P}(\mathbf{y}, m|\mathbf{\Phi}^k)}{\hat{P}(\mathbf{y}|\mathbf{\Phi}^k)} \frac{\partial \log \hat{P}(\mathbf{y}^t|m, \mathbf{\Phi}^{k+1})}{\partial y_j}, \end{aligned} \tag{25}$$

where K is the total number of Gaussian PDFs in all HMM states.

We will assume that the recognizer models the conditional PDF of the observation as a mixture of diagonal-covariance Gaussians and therefore

$$\begin{aligned} \frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial y_j} &= \sum_{t=1}^{T} \sum_{m=1}^{K} \frac{\hat{P}(\mathbf{y}, m|\mathbf{\Phi}^k)}{\hat{P}(\mathbf{y}|\mathbf{\Phi}^k)} \frac{(\mu_{mj} - y_j^t)}{\sigma_{mj}^2}, \end{aligned} \tag{26}$$

where μ_{mj}, and σ_{mj}^2 are the mean and the variance of the jth element of the mth PDF respectively.

In the following, we will derive the updating equation for the four sets of parameters used in the symplectic map, namely $\mathbf{A}, \mathbf{B}, \mathbf{C}$, and $\mathbf{D}$. Let the non-linear function used in both feed-forward neural networks be the hyperbolic tangent as stated before.

Starting with A and B, to calculate the update equation for a symplectic parameter a_{qr} and b_{qr} for $q = 1, 2, \cdots, H$, and for $r = 1, 2, \cdots, \frac{n}{2}$, we have to calculate the partial derivative of the auxiliary function with respect to these parameters. These partial derivatives are related to the partial derivatives of the auxiliary function with respect to the features by the following relation

$$\begin{aligned} \frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial a_{qr}} &= \sum_{j=1}^{\frac{n}{2}} \frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial y_{1j}} \frac{\partial y_{1j}}{\partial a_{qr}} \\ &+ \sum_{j=1}^{\frac{n}{2}} \frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial y_{2j}} \frac{\partial y_{2j}}{\partial a_{qr}}, \end{aligned} \tag{27}$$

and

$$\begin{aligned} \frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial b_{qr}} &= \sum_{j=1}^{\frac{n}{2}} \frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial y_{2j}} \frac{\partial y_{2j}}{\partial b_{qr}}, \end{aligned} \tag{28}$$

where

$$\frac{\partial y_{1j}}{\partial a_{qr}} = \begin{cases} 2x_{2r} \sum_{h=1}^{H} \left(c_h a_{hj} S(\mathbf{a}_h \mathbf{x}_2)[1 - S^2(\mathbf{a}_h \mathbf{x}_2)] \right) \\ \qquad \text{for } r \neq j \\ 2x_{2r} \sum_{h=1}^{H} \left(c_h a_{hj} S(\mathbf{a}_h \mathbf{x}_2)[1 - S^2(\mathbf{a}_h \mathbf{x}_2)] \right) \\ -c_q[1 - S^2(\mathbf{a}_q \mathbf{x}_2)] \\ \qquad \text{for } r = j \end{cases} \tag{29}$$

$$\frac{\partial y_{2j}}{\partial a_{qr}} = \sum_{k=1}^{\frac{n}{2}} \frac{\partial y_{1k}}{\partial a_{qr}} \frac{\partial y_{2j}}{\partial y_{1k}}, \tag{30}$$

$$\frac{\partial y_{2j}}{\partial y_{1k}} = -\sum_{h=1}^{H} \left(d_h b_{hj} b_{hk} S(\mathbf{b}_h \mathbf{y}_1)[1 - S^2(\mathbf{b}_h \mathbf{y}_1)] \right), \quad (31)$$

and

$$\frac{\partial y_{2j}}{\partial b_{qr}} = \begin{cases} 2y_{1r} \sum_{h=1}^{H} \left(c_h b_{hj} S(\mathbf{b}_h \mathbf{y}_1)[1 - S^2(\mathbf{b}_h \mathbf{x}_2)] \right) \\ \qquad \text{for } r \neq j \\ 2y_{1r} \sum_{h=1}^{H} \left(c_h b_{hj} S(\mathbf{b}_h \mathbf{y}_1)[1 - S^2(\mathbf{b}_h \mathbf{x}_2)] \right) \\ -d_q [1 - S^2(\mathbf{b}_q \mathbf{y}_1)] \\ \qquad \text{for } r = j \end{cases} \quad (32)$$

For $\mathbf{C}$ and $\mathbf{D}$, the derivation will follow the same procedure, but the resulting equations are much simpler. The partial derivative of the auxiliary function with respect to the symplectic parameter c_q and d_q for $q = 1, 2, \cdots, H$, are related to the partial derivatives of the auxiliary function with respect to the features by the following relation

$$\frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial c_q} = \sum_{j=1}^{\frac{n}{2}} \frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial y_{1j}} \frac{\partial y_{1j}}{\partial c_q} + \sum_{j=1}^{\frac{n}{2}} \frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial y_{2j}} \frac{\partial y_{2j}}{\partial c_q}, \quad (33)$$

and

$$\frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial d_q} = \sum_{j=1}^{\frac{n}{2}} \frac{\partial Q(\mathbf{\Phi}^k, \mathbf{\Phi}^{k+1})}{\partial y_{2j}} \frac{\partial y_{2j}}{\partial d_q}, \quad (34)$$

where

$$\frac{\partial y_{1j}}{\partial c_q} = a_{qj}[1 - S^2(\mathbf{a}_q \mathbf{x}_2)], \quad (35)$$

$$\frac{\partial y_{2j}}{\partial c_q} = \sum_{k=1}^{\frac{n}{2}} \frac{\partial y_{1k}}{\partial c_q} \frac{\partial y_{2j}}{\partial y_{1k}}, \quad (36)$$

and

$$\frac{\partial y_{2j}}{\partial d_q} = b_{qj}[1 - S^2(\mathbf{b}_q \mathbf{y}_1)]. \quad (37)$$

To update the symplectic parameters in each iteration, the symplectic parameters that maximize the likelihood can be estimated at each iteration using gradient based optimization algorithms. Equations 27- 37 can be used for updating the symplectic parameters iteratively until the value of the likelihood is maximized.

The steps of the generalized EM iterative algorithm to update the symplectic parameters and the HMM parameters are

1. Initialize the symplectic parameters and the HMM parameters.
2. Calculate the transformed feature vectors y using the current symplectic maps and the input feature vectors as in Equations 16 and 17.
3. Using the current value of the parameters $\mathbf{\Phi}^k$, estimate the auxiliary function.
4. Using the current HMM parameters, estimate the symplectic parameters that maximize the auxiliary function by using a gradient-based optimization algorithm.
5. Update the transformed feature vectors y using the current symplectic maps and the input feature vectors as in Equations 16 and 17.
6. Estimate the HMM parameters that maximize the auxiliary function using the current symplectic parameters.
7. Iterate (starting from 3) until convergence.

In our experiments, we used the conjugate gradient algorithm to update the symplectic parameters at each iteration. The computational complexity of updating the symplectic parameters using the conjugate gradient algorithm is $O(2(n+1)HN + nH^2N)$ which compares favorably to $O(n^2N)$ for linear approaches for large n, where n is the dimension of the feature vector, H is the number of hidden units in the neural network, and N is the number of feature vectors in the training data.

VI. Experiments and Results

We will apply the symplectic maximum likelihood transform (SMLT) described in the previous section to two different problems of high-dimensional probabilistic model estimation. The first is the estimation of the joint PDF of an example of order statistics, and the second is the estimation of the joint PDF of the Mel-frequency cepstrum coefficients of a speech utterance using Gaussian mixture hidden Markov model as the hypothesized probabilistic model. In the first set of experiments, we compare the likelihood obtained at each iteration to the likelihood obtained without using any transformation of the measurements, and the likelihood obtained by using maximum likelihood linear transformation (MLLT) of the measurements with all methods having approximately the same number of total parameters. In the second set of experiments, the phoneme recognition accuracies obtained by the three methods are compared. In both set of experiments, the conjugate gradient algorithm was used to update the symplectic parameters in each iteration. The number of hidden nodes of the neural network used in constructing the symplectic map is three in all experiments. Therefore, the total number of symplectic parameters in each experiment is $3n + 6$,where n is the dimension of the feature vector. In all experiments, initializing the symplectic parameters by very small values compared to the dynamic range of the original features gave the best results that are reported here.

A. Order Statistics

Order statistics are important features that are usually used in classification and coding. Examples of order statistics are the five largest wavelet coefficients, or the median

of a given set of values. The joint distribution of a collection of order statistics obtained from a set of i.i.d. random variables can be calculated exactly given the probability density function of these random variables [31]. Given N realizations of the random vector $\mathbf{x}$ of length n with $\{x_i\}_{i=1}^{n}$ are iid random variables, let $y_i = G(x_i)$. Define $\mathbf{y} = [y_1 \cdots y_n]^{'}$. Let $\mathbf{z} = [z_1 \cdots z_M]^{'}$ be obtained from $\mathbf{y}$ by sorting into ascending order and selecting the first M values. Let $C_{Y_i}(y_i)$, and $P_{Y_i}(y_i)$ be the cumulative distribution function (CDF) and PDF of y_i $\forall i$, respectively. Then, the joint PDF of $\mathbf{Z}$ is given by

$$P_{\mathbf{Z}}(z_1, z_2, \cdots, z_M) = \frac{N!}{(N-M)!} \prod_{i=1}^{M} P_{Y_i}(z_i) [1 - C_{Y_i}(z_M)]^{(N-M)} \quad (38)$$

In this experiment, we generated a set of N iid realizations of Gaussian random vectors $\{\mathbf{x}^j\}_{j=1}^{N}$ of length $n = 100$ with zero mean and identity covariance matrix, and transformed each component to $y_i^j = |x_i^j|$. After sorting the one hundred transformed components of each random vector in ascending order, we took the first thirty components, i.e. $M = 30$. These 30 components of each realization were used to estimate the symplectic parameters and the parameters of a Gaussian mixture (GM) probabilistic model of the joint probability density function of these 30 components. The parameters are estimated to maximize the likelihood of the training data using the algorithm described in the previous section. The log likelihood of the training data using (SMLT+GM) is compared to the log likelihood achieved using the (MLLT+GM) approach as described in [9] and discussed briefly in section III, and to the log likelihood achieved using the EM algorithm to train a Gaussian mixture model using the same data without transformation (GM). The hidden variables in this experiment are the identity of the Gaussian PDF in the mixture. The Gaussian mixture model in the three methods is initialized using the Linde-Buzo-Gray (LBG) algorithm [32]. The MLLT transform was initialized with a matrix very close to the identity matrix by using very small off-diagonal values. The symplectic parameters are initialized by very small values compared to the dynamic range of the original features. We considered four other random initializations for the MLLT and the SMLT transforms and the resulting log likelihood were the same or less than those reported here for both methods. The number of training vectors N was chosen to be equal to 2×10^7. The comparison of the three methods is shown in figure 1. The figure shows significant increase in the log likelihood by using the symplectic map. Since an increase in the likelihood can be achieved by increasing the number of parameters of the model, e.g. by increasing the number of Gaussian densities in the mixture, a comparison of the number of parameters used in each method is provided in table I. The table shows that the increase in the likelihood using SMLT is achieved using fewer parameters than both GM and MLLT. To compensate for the additional number of transformation parameters needed by SMLT and MLLT, we used a different number of Gaussian PDFs in the mixture for each method. The number of Gaussian PDFs used by each method is provided in table II

TABLE I
TOTAL NUMBER OF PARAMETERS FOR EACH METHOD

Method	Number of Parameters
GM	1952
MLLT+GM	1937
SMLT+GM	1926

TABLE II
NUMBER OF GAUSSIAN PDFS IN THE MIXTURE FOR EACH METHOD

Method	Number of PDFs
GM	32
GM+MLLT	17
GM+SMLT	30

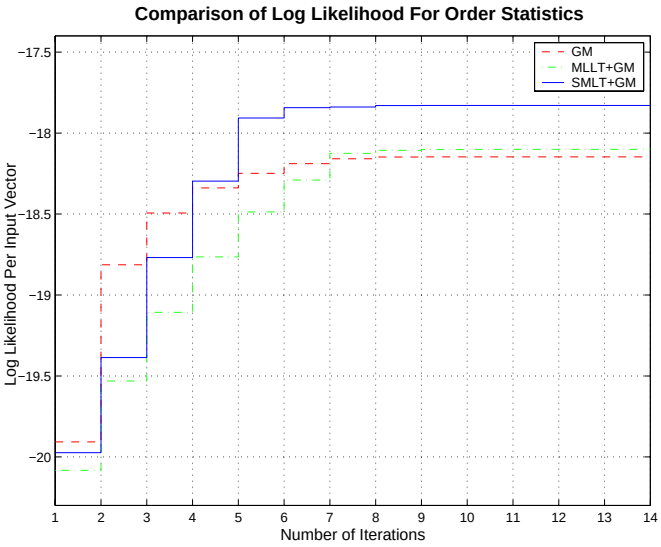

Fig. 1. Comparison of Log Likelihoods for Order Statistics

B. Modeling of Dynamic Patterns Using HMM

To test the performance of our approach on modeling patterns of variable length, we take the speech signal as an example. Most speech recognition systems use a Gaussian mixture HMM-based recognizer and use the Mel-frequency cepstrum coefficients (MFCC) and their deltas as the input acoustic features to the recognizer [33]. In our experiments, the 61 phonemes defined in the TIMIT database are mapped to 48 phoneme labels for each frame of speech as described in [34]. A three-state left-to-right model of each phoneme is trained using the EM algorithm. The number of mixtures per state ranged from four to thirteen based on the number of frames of training data assigned to the state. The SMLT approach is applied to an input feature vector

that consists of twelve MFCC coefficients, energy, and their deltas. These acoustic features are calculated for the whole training subset of the TIMIT database, and the parameters of the symplectic map and the HMM models are jointly optimized to maximize the likelihood as described in the previous section. The SMLT and MLLT transforms are initialized the same way as in the previous set of experiments. We considered four other random initializations for the MLLT and the SMLT transforms and the resulting phoneme recognition accuracies were the same or less than those reported here for both methods. The parameters of the triphone models are tied together using the same approach as in [35].

TABLE III
PHONEME RECOGNITION ACCURACY

Recognizer	Recog. Accuracy	No. of Parameters
MFCC+GM	73.7%	25407324
MLLT+GM	74.6%	25407311
SMLT+GM	75.6%	25407302

The phoneme recognition results and the total number of parameters for the three methods are provided in table III. It shows an improvement in the recognition accuracy using the SMLT approach as compared to MLLT and the baseline system. Previous phoneme recognition accuracy results on the TIMIT database, [6], verify that the improvement in recognition accuracy achieved here by using SMLT is significant.

VII. DISCUSSION

This work proposes a model enforcement approach to feature transform design for statistical classification, identification, and recognition systems. This approach calculates a one-to-one map of the features to minimize the relative entropy of the true PDF of the features and the hypothesized PDF. The model enforcement criterion is shown to be a generalization of a wide variety of existing transform design criteria including redundancy reduction transforms, and transformation to normality techniques. A useful special case of the model enforcement approach is that of the symplectic maximum likelihood transform (SMLT), in which a volume-preserving map is optimized jointly with the model parameters to minimize the relative entropy. A computationally efficient EM-based iterative algorithm for SMLT optimization is described. This iterative algorithm was applied to two important statistical modeling problems: estimation of the joint PDF of order statistics using a Gaussian mixture, and modeling the MFCC coefficients of the speech signal using an HMM. In the first application, an improvement in the log likelihood is achieved using the SMLT approach compared to MLLT and compared to using the original features. This improvement is achieved with a total number of parameters less than other methods in both cases. Phoneme recognition experiments also show significant improvement in recognition accuracy achieved by SMLT compared to the other two methods.

The model enforcement approach is intended to provide a general framework for many interesting feature transformations to reduce inaccuracy of statistical models. The paper provides two example applications; several other special cases can be defined by the choice of the parametric form of the map, constraints on the determinant of its Jacobian matrix, and the form of the parameterized likelihood function. The choice of a certain solution is related to the complexity of the problem and the nature of the features used in the system. The main advantage of this general formulation is the avoidance of strict assumptions about the features or the model as in previous approaches.

VIII. ACKNOWLEDGMENT

This work was supported by NSF award number 0132900. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. The authors would like to thank the three anonymous reviewers, Stephen Levinson, and Brian Kingsbury for their useful comments on the manuscript.

REFERENCES

[1] D. F. Andrews, R. Gnanadesikan, J. L. Warner, "Transformations of Multivariate Data," *Biometrics*, vol. 27, pp. 825-840, 1971.

[2] B. H. Juang, Stephen E. Levinson, and M. M. Sondhi, "Maximum Likelihood Estimation for Multivariate Mixture Observations of Markov Chains," *IEEE Trans. on Information Theory*, vol. IT 32, No.2, March 1986.

[3] Dan Roth, "Learning to resolve natural language ambiguities: a unified approach," *Proceedings of 15th Conference of the American Association for Artificial Intelligence,*, AAAI Press, Menlo Park, Madison, pp. 806-813, 1998.

[4] A. Ljolje, "The importance of cepstral parameter correlations in speech recognition," *Computer, Speech, and Language*, vol. 8, pp. 223-232, 1994.

[5] Marian Stewart Bartlett, *Face Image Analysis by Unsupervised Learning,* Kluwer Academic Publishers, Boston, 2001.

[6] Mohamed Kamal Omar, and Mark Hasegawa-Johnson, "Approximately Independent Factors of Speech Using Symplectic Maps," *IEEE Trans. on Speech and Audio Processing,* In Press.

[7] Te-Won Lee, *Independent Component Analysis,* Kluwer Academic Publishers, Boston, 1998.

[8] Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification,* Wiley, New York, NY, 2000.

[9] R. A. Gopinath, "Maximum Likelihood Modeling With Gaussian Distributions For Classification," *IEEE Proceedings of ICASSP*, Seattle, Washington, 1998.

[10] Thomas M. Cover, and Joy A. Thomas, *Elements of Information Theory,* Wiley, New York, NY, 1997.

[11] H. V. Poor, *An Introduction to Signal Detection and Estimation,* Springer-Verlag, New York, 1994.

[12] Vladimir N. Vapnik, *Statistical Learning Theory,* Wiley, New York, NY, 1998.

[13] D. R. Cox, *The Analysis of Binary Data,* Methuen, London, UK, 1970.

[14] J. W. Tukey, *Exploratory Data Analysis Reading,* Addison-Wesley, MA, 1977.

[15] J. W. Tukey, "On the comparative anatomy of transformations," *Ann. Math. Statist.*, vol. 28, pp. 602-632, 1957.

[16] G. E. P. Box, and D. R. Cox, "An analysis of transformations," *Journal of Royal Statist. Soc.*, vol. 26, pp. 211-252, 1964.

[17] M. E. Tipping, and C. Bishop, "Mixtures of Principal Component Analysis," *Proc. of IEEE 5th Int. Conf. Artificial Neural Networks*, 1997.

[18] S. Roweis, "EM Algorithms for PCA and SPCA," *Advances in Neural Information Processing Systems*, MIT press, Cambridge, MA, vol. 10, pp. 626-632, 1998.

[19] N. Campbell, "Canonical variate analysis - a general formulation," *Australian Journal of Statistics*, vol. 26, pp. 86-96, 1984.
[20] T. Hastie, and R. Tibshirani, *Discriminant Analysis by Gaussian Mixtures,* Technical Report, AT&T Bell Laboratories, 1994.
[21] N. Kumar, *Investigation of silicon-auditory models and generalization of linear discriminant analysis for improved speech recognition,* Ph.D. dissertation, John Hopkins Univ., Baltimore, MD, 1997.
[22] Aapo Hyvarinen, Juha Karhunen, and Erkki Oja, *Independent Component Analysis,* Wiley, New York, NY, 2001.
[23] Te-Won Lee, Mark Girolami, Anthony J. Bell, and Terrence J. Sejnowski, "A Unifying Information-Theoretic Framework For Independent Component Analysis, " *Computers & Mathematics with Applications* , vol. 31 (11), pp. 1-21, March 2000.
[24] Athanasios Papoulis, *Probability, Random Variables, and Stochastic Processes,* McGraw-Hill, New York, 1991.
[25] R. Neal and G. Hinton, "A View of the EM Algorithm that Justifies Incremental, Sparse, and other Variants," *Learning in Graphical Models*, Kluwer Academics, 1998.
[26] Ralph Abraham, Jerrold E. Masden, *Foundation of Mechanics,* The Benjamin/Cummings Publishing Company, 1978.
[27] L. C. Parra, *Symplectic nonlinear component analysis,* In Advances in Neural Information Processing Systems, 8, MIT Press, Cambridge, MA., pp. 437-443, 1996.
[28] K. Hornik, M. Stinchcombe, H. White, "Multilayer Feedforward Neural Networks Are Universal Approximators, " *Neural Network*, 2, pp. 359-366, 1989.
[29] L. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. of the IEEE*, vol. 77, No. 2, pp. 257–286, 1989.
[30] X. D. Huang, Y. Ariki, and M. A. Jack, *Hidden Markov Models For Speech recognition,* Edinburgh University Press, Edinburgh, UK, 1990.
[31] H. A. David, *Order Statistics,* Wiley, New York, 1981.
[32] Y. Linde, A. Buzo, and R. M. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Trans. on Communications*, vol. 28, pp. 84-94, Jan. 1980.
[33] S. B. Davis, and P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," *IEEE Trans. Acoust., Speech, and Sig. Proc.*, vol. 28, pp. 357-366, Aug. 1980.
[34] Kai-Fu Lee, and Hsiao-Wuen Hon, "Speaker Independent Phone Recognition Using Hidden Markov Models," *IEEE Trans. on ASSP*, vol. 37, pp. 1641-1648, November 1989.
[35] S. Young, and P. Woodland, "State Clustering in hidden Markov model continous speech recognition," *Computer, Speech, and Language*, 8(4), pp. 369-383, October 1994.