Optimal detection and classification of diverse short-duration signals

Paul M. Baggenstoss
Naval Undersea Warfare Center
Newport RI, 02841
401-832-8240 (TEL)
Email: p.m.baggenstoss@ieee.org

***Abstract*—Recent theoretical advances in class-dependent feature extraction are reviewed. These advances, culminating in the multi-resolution HMM (MR-HMM) statistical model are proposed for the detection and classification of transient signals that are composed of diverse components with widely varying structure and resolution.**

I. Feature Bottleneck

In audio surveillance and scene monitoring [1], [2], the large variety of signal types requires the application of several signal-processors (feature extractors). For example, bird calls, which are narrow-band signals, might require FFT processing, but short-duration clicks might require special time-domain processing with fine time resolution. As the number and variety of signal types grow, the list of features needed to characterize all the signals grows. The problem is made worse, not better, by the classical (Bayesian) decision theory. The classical Bayesian classifier [3] is given by

$$\hat{k} = \arg\max_{k=1}^{M} \left\{ p(\mathbf{x}|H_k)\ p(H_k) \right\}, \tag{1}$$

where $\mathbf{x} \in \mathcal{R}^N$ is the raw (time-series) data, H_k is the statistical hypothesis of class k, $p(H_k)$ is the *a priori* class probability. When the probability density functions (PDFs) $p(\mathbf{x}|H_k)$ are not known, they have to be estimated. The curse of dimensionality [4] makes it impractical to estimate these PDFs. The usual practice is to extract as much meaningful information as possible from $\mathbf{x}$ into a set of features $\mathbf{z} = T(\mathbf{x}), \mathbf{z} \in \mathcal{R}^D, \quad D \ll N$. The classifier is then re-cast in terms of feature set $\mathbf{z}$

$$\hat{k} = \arg\max_{k=1}^{M} \left\{ p(\mathbf{z}|H_k)\ p(H_k) \right\}. \tag{2}$$

This classifier implicitly assumes that the feature set $\mathbf{z}$ is a sufficient statistic. Unfortunately, in problems with many diverse signals, the dimension of the feature $\mathbf{z}$ needed to adequately solve the problem is still much too high for accurate PDF estimation of $p(\mathbf{z}|H_k)$. Attempts to reduce dimension include feature selection, linear regression, principal component analysis (PCA), Fisher discriminant analysis (FDA) and manifold learning [5], [2]. While these methods are sometimes effective, this does not solve the fundamental problem that all the information must be squeezed into one common feature, what can be called the *feature bottleneck*.

II. Class-Specific Method (CSM)

CSM refers to a set of related theoretical approaches that avoid the feature bottleneck.

A. Class-Specific Features (CSF)

The first embodiment of CSM is CSF [6], [7], [8] in which the Bayesian classifier (1) is divided by the constant term $p(\mathbf{x}|H_0)$

$$\hat{k} = \arg\max_{k} \left\{ \frac{p(\mathbf{x}|H_k)}{p(\mathbf{x}|H_0)}\ p(H_k) \right\}, \tag{3}$$

where H_0 is a reference hypothesis, also called "dummy hypothesis" by Van Trees [9]. According to the formal definition of a sufficient statistics [10], a feature $\mathbf{z}_k = T_k(\mathbf{x})$ is a sufficient statistic for the binary test between H_0 and H_k if

$$\frac{p(\mathbf{x}|H_k)}{p(\mathbf{x}|H_0)} = \frac{p(\mathbf{z}_k|H_k)}{p(\mathbf{z}_k|H_0)}. \tag{4}$$

If we select class-dependent sufficient statistics that are sufficient statistic for H_0 vs. H_k, we obtain the CSF classifier [6],

$$\hat{k} = \arg\max_{k} \left\{ \frac{p(\mathbf{z}_k|H_k)}{p(\mathbf{z}_k|H_0)}\ p(H_k) \right\}, \tag{5}$$

which represents an important departure from (2) because it integrates feature extraction into the classifier and does not require one common featureset. The feature bottleneck is avoided because rather than requiring one feature set $\mathbf{z}$ that is a sufficient statistic for the combined problem (2), the individual sufficient statistic $\mathbf{z}_k$ need only be sufficient for distinguishing the corresponding class H_k from H_0. As proof that the feature bottleneck is avoided, Kay [11] notes an example where low-dimensional class-specific sufficient statistics exist for each class, but no combined sufficient statistic $\mathbf{z}$ exists, no matter how large, for the Bayesian feature classifier (2).

While CSF is a significant theoretical advance, there are issues. First, the CSF theory does not explain what happens when the features are not sufficient statistics. There are also numerical issues that force the use of an analytic expression for $p(\mathbf{z}_k|H_0)$. Luckily, analytic expressions can be found for the most widely-used feature extraction methods [12]. The saddle-point approximation can be used in cases where only the moment generating function (MGF) can be derived [12]. Still, one issue with CSF remains: the *one class, one feature* assumption is not realistic in real-world scenarios where data collections are diverse and not representable by a single feature.

B. Class-Specific Feature Mixture (CSFM)

To get around the *one class, one feature* assumption of CSF, CSFM assumes that class H_k is composed of sub-classes

represented by an *additive mixture* PDF. We assume that class H_k is composed of L subclasses $H_{k,l},\ 1 \leq l \leq L$, that have relative probabilities of occurrence α_l and individual sub-class PDFs $p(\mathbf{x}|H_{k,l})$. The mixture PDF for H_k is given by:

$$p(\mathbf{x}|H_k) = \sum_{l=1}^{L} \alpha_l \; p(\mathbf{x}|H_{k,l}), \tag{6}$$

where $\sum_{l=1}^{L} \alpha_l = 1$.

If we assume that each sub-class has a different feature (sufficient statistic) to distinguish it from H_0, and divide by the common factor $p(\mathbf{x}|H_0)$, we get

$$\frac{p(\mathbf{x}|H_k)}{p(\mathbf{x}|H_0)} = \sum_{l=1}^{L} \alpha_{k,l} \; \frac{p(\mathbf{x}|H_{k,l})}{p(\mathbf{x}|H_0)} = \sum_{l=1}^{L} \alpha_{k,l} \; \frac{p(\mathbf{z}_{k,l}|H_{k,l})}{p(\mathbf{z}_{k,l}|H_0)}. \tag{7}$$

Rather than use LM different feature sets, we use the same L features for each class. The CSFM classifier

$$\hat{k} = \arg\max_{k=1}^{M} \left\{ \sum_{l=1}^{L} \alpha_{k,l} \; \frac{p(\mathbf{z}_l|H_{k,l})}{p(\mathbf{z}_l|H_0)} \right\} \; p(H_k) \tag{8}$$

may be interpreted as a *data-specific* feature classifier because for each data sample $\mathbf{x}$, the factor $1/p(\mathbf{z}_l|H_0)$ has a dominant effect, effectively picking one feature to classfy the sample. Practical experiments show that CSFM produces very significant error reduction over CSF.

C. PDF Projection Theorem (PPT)

Despite the elegant way in which CSF and CSFM circumvent the feature bottleneck, there remains a theoretical concern: what happens if the features are not completely sufficient? The PPT [13], [14], [15], [16], [17] resolves this question by proving that CSF and CSFM are indeed valid Bayesian classifiers. If we multiply (5) by the common term $p(\mathbf{x}|H_0)$ and re-arrange, we get

$$\hat{k} = \arg\max_{k} \left\{ \frac{p(\mathbf{x}|H_0)}{p(\mathbf{z}_k|H_0)} \; p(\mathbf{z}_k|H_k) \; p(H_k) \right\}, \tag{9}$$

with no effect on the classification decision, but huge effect on the interpretation. The PPT [14] proves that the term in braces in (9) excluding $p(H_k)$

$$G_k(\mathbf{x}; T_k, H_0) = \frac{p(\mathbf{x}|H_0)}{p(\mathbf{z}_k|H_0)} \; p(\mathbf{z}_k|H_k) \tag{10}$$

is a PDF, so it integrates to 1 over $\mathbf{x}$ regardless of the feature transformation $T_k(\mathbf{x})$ (aside from some mild regularity conditions). The PPT further proves that $G_k(\mathbf{x}; T_k, H_0)$ is a member of the class of PDFs that *generate* [1] feature PDF $p(\mathbf{z}_k|H_k)$ through feature transformation $T_k(\mathbf{x})$. PDF $G_k(\mathbf{x})$ is called the *projected* PDF, since it projects the PDF $p(\mathbf{z}_k|H_k)$ back to the raw-data, forming the Bayesian classifier

$$\hat{k} = \arg\max_{k} \left\{ G_k(\mathbf{x}; T, H_0) \; p(H_k) \right\}, \tag{11}$$

defined on the *raw data*, not on a feature set. This means that $\mathbf{z}_k$ *does not need to be a sufficient statistic* for H_k vs H_0, so (5), (8), and (9) are all valid Bayesian classifiers! Sufficiency is only a goal for optimal performance. There is an even more important implication of PPT: we can make the reference hypothesis class-dependent via

$$\hat{k} = \arg\max_{k} \left\{ G_k(\mathbf{x}; T, H_{0,k}) \; p(H_k) \right\}. \tag{12}$$

We can now choose each $\mathbf{z}_k$ and $H_{0,k}$ jointly to optimize the "sufficiency" of each binary test H_k vs $H_{0,k}$. In the remainder of the paper, we assume for simplicity a common reference hypothesisi H_0, but this is not necessary.

D. The Chain Rule

The Chain-rule makes constructing class-specific classifiers based on multi-stage feature extraction much easier. Many feature transformations are made of stages, for example $\mathbf{y} = T_y(\mathbf{x}),\ \ \mathbf{w} = T_w(\mathbf{y}),\ \ \mathbf{z}_k = T_z(\mathbf{w})$, suggesting the chain-rule form of (10),

$$G_k(\mathbf{x}; T_k, H_0) = \left[\frac{p(\mathbf{x}|H_{0x})}{p(\mathbf{y}|H_{0x})}\right] \left[\frac{p(\mathbf{y}|H_{0y})}{p(\mathbf{w}|H_{0y})}\right] \left[\frac{p(\mathbf{w}|H_{0w})}{p(\mathbf{z}|H_{0w})}\right] p(\mathbf{z}_k; H_k), \tag{13}$$

where H_{0x}, H_{0y}, H_{0w} are stage-dependent statistical hypotheses. This suggests an elegant modular software framework for a feature extraction chain: `[y,J]=stage1(x,J);` , then `[w,J]=stage2(y,0);` , etc., where variable `J` accumulates the log-PDF ratios $\log\{p(\mathbf{x}|H_{0x})/p(\mathbf{y}|H_{0x})\}$, $\log\{p(\mathbf{y}|H_{0y})/p(\mathbf{w}|H_{0y})\}$, ...

E. Maximum Entropy (ME)

While the PPT is a theoretical advance, the question still remains "is (10) the *best* PDF that generates $p(\mathbf{z}_k|H_k)$?" What is meant by "best" PDF? The maximum entropy principle can be used to quantify "best" in this sense. Essentially, the maximum entropy principle holds that the PDF we choose should have the most *disorder* of all PDFs that meet the given requirements, which insures that we have not intruduced any bias or hidden assumptions. So what is disorder and what are the requirements? The requirements are that the PDF $G_k(\mathbf{x}; T_k, H_0)$ in (10) must generate $p(\mathbf{z}_k|H_k)$ (see explanation of what this means in section II-C), which the PPT proves, so the requirement is met. The disorder of PDF $p(\mathbf{x})$ is defined mathematically as the entropy

$$Q = \int_{\mathbf{x}} \log p(\mathbf{x}) \; p(\mathbf{x}) \; \mathrm{d}\mathbf{x}.$$

If the PDF has no special requirements to meet, the ME PDF is the uniform distribution [18]. Given mean and variance constraints, it is the Gaussian PDF [18]. It was recently discovered [19] that under additional mild conditions[2], $G_k(\mathbf{x}; T_k, H_0)$ it is the maximum entropy PDF [20], [18] over all PDFs that generate $p(\mathbf{z}_k|H_k)$. This further strengthens the theory and provides a reason for the use of Gaussian or exponential PDF as the reference hypothesis.

III. Multi-Resolution HMM

The MR-HMM [21] extends CSM to joint segmentation and classification. We assume the reader has knowledge of the classical hidden Markov model (HMM) [22].

[1] The PDF $G(\mathbf{x})$ is said to *generate* feature PDF $g(\mathbf{z})$ if random samples of $G(\mathbf{x})$ passed through the feature transformation $\mathbf{z} = T(\mathbf{x})$ have exactly distributiuon $g(\mathbf{z})$.

[2] That the feature $\mathbf{z}$ contain a norm $d = \|\mathbf{x}\|_p$ and $p(\mathbf{x}|H_0)$ depend on $\mathbf{x}$ only through d. Examples: H_0 is *iid* Gaussian noise and $d = \|\mathbf{x}\|_2$, or exponential noise and $d = \|\mathbf{x}\|_1$.

A. Motivation and Overview

Up to now, we have assumed that $\mathbf{x}$ can be analyzed using a single feature set. But, in practice, short-duration signals are composed of several components, each with different duration and spectral content. We could apply the classical HMM [22] in which the data is segmented into uniform-sized segments from which a common feature set is extracted. The sequence of features $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2 \ldots \mathbf{z}_T]$ is then modeled as an HMM. The HMM assumes that the data consists of various sub-classes, called states, which can occur at arbitrary times and for arbitrary lengths of time within the time-series. The HMM is defined by a set of parameters $\mathbf{\Lambda}$ that are estimated from training data. Then, the HMM is used to calculate the likelihood function $L(\mathbf{Z}; \mathbf{\Lambda})$ for any input data $\mathbf{x}$ and MR-HMM parameters $\mathbf{\Lambda}$. This is used as a Bayesian classifier: $\hat{k} = \arg\max_{k=1}^{M} \; L(\mathbf{x}; \mathbf{\Lambda}_k) \; p(H_k)$. In addition to calculating the likelihood function, we may compute the *a posteriori* state probabilities, which indicate where the various states are likely to have occurred within the time-series. As can be seen, the HMM provides many desirable features for the analysis of diverse short-duration signals. The main problem with the HMM is the existence of the feature bottleneck in calculating the feature $\mathbf{z}$ and the use of a fixed segmentation size.

In our view of diverse short-duration signals, signal components can occur with arbitrary time and/or frequency resolution, requiring the application of a variety of feature extraction methods, covering a variety of time and frequency resolutions. Thus, we would like to adapt the HMM to allow for variable segments size and multiple feature extraction methods. Also, rather than first segmenting the data, then later applying feature extraction to the chosen segments, it should consider all possible ways to segment the data. All these things are done by the MR-HMM. Like the HMM, the MR-HMM provides a likelihood function for use in classification. In addition, it estimates precise signal component location and sub-class identity. It uses the PPT in order to apply various feature extraction methods in a single model.

B. Segmentation and Proxy HMM

For the MR-HMM, time is measured in short time segments of length K called *base segments*. The base segments are analogous to the fixed segments of the classical HMM. Let $\mathbf{x}_i^b, \quad 1 \leq i \leq T$, be the base segments, where T is the total number of base segments. We don't actually carve up the data into base segments, but it is useful to define them. Instead, we assume that the data is broken into segments that are (a) selected from a pre-determined set of segment sizes made of an integer number of base segments, and (b) classified as one of L sub-classes. We define a *segmentation* as one possible sequence of segments and sub-classes. Refer to Figure 1 in which there are $L = 2$ sub-classes (background and noise burst). On the top is the time-series of length $T = 34$ base segments wherein we see two bursts. Let the allowable segment sizes be $a = 1, \quad b = 2, \quad c = 3, \quad d = 4, \quad e = 5$. Then, two possible segmentations for the time-series are

$$\begin{aligned} \mathbf{q}_1 &= [1d, 1d, 2e, 1d, 1c, 2c, 1d, 1d, 1d], \\ \mathbf{q}_2 &= [1d, 1d, 2e, 1c, 1d, 2c, 1d, 1d, 1d], \end{aligned}$$

where the number represents the sub-class and the letter the segment size. The segmentations are represented as dotted boxes drawn on top of the time-series. These segmentations differ in the way that the gap between the two bursts is divided, either as 1d, 1c, or 1c, 1d.

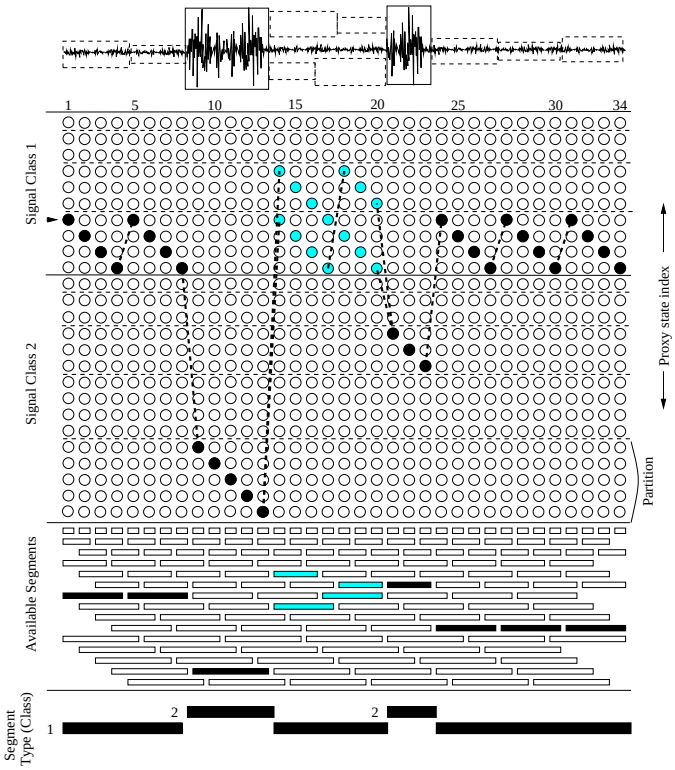

Fig. 1. Illustration of the relationship between MR-HMM segmentation and proxy HMM trellis path.

In Figure 1, "Available Segments", we see all the allowable segment sizes and time shifts. The state trellis ("proxy state index" in figure 1) is divided into "partitions", each representing a choice of sub-class and segment size with vertical extent equal to the segment length. Once the system transitions into the first state of a partition, it is forced to complete the segment, counting out the states called wait-states. All possible segmentations map to a unique path through the trellis. The paths corresponding to segmentations $\mathbf{q}_1$ and $\mathbf{q}_2$ are shown as dotted lines.

The proxy HMM is a notional Markov model with N_p states consisting of all wait-states ($N_p = 25$ in Figure 1). Due to the forced wait-state counting, the proxy HMM has a very structured state transition matrix (STM). The parameters of the proxy HMM are the STM $\mathbf{A} = \{A_{i,j}\}, \; 1 \leq i \leq N_p, \; 1 \leq j \leq N_p$ and the initial state probabilities $\boldsymbol{\pi} = \{\pi_i\}, \; 1 \leq i \leq N_p$. We initialize these to "flat" probabilities, then iteratively re-estimate them.

C. MR-HMM likelihood calculations using CSM

All likelihood calculations for the MR-HMM are based on the conditional PDF

$$p(\mathbf{x}|\mathbf{q}) = \prod_{s \in \mathbf{q}} p(\mathbf{x}_s | m_s),$$

where s is a segment within segmentation $\mathbf{q}$, m_s is the subclass identity, and $\mathbf{x}_s$ is the time-series data in the segment. Unless

we know the likelihood functions $p(\mathbf{x}_s|m_s)$ defined on the raw-data segments, we need to use CSM. The PDFs $p(\mathbf{x}_s|m_s)$ are obtained using the PPT (equation 10) through a feature specific to sub-class m_s on a segment of length k_s base segments:

$$p(\mathbf{x}_s|m_s) = \frac{p(\mathbf{x}_s|H_0)}{p(\mathbf{z}_{k_s,m_s}|H_0)}\, p(\mathbf{z}_{k_s,m_s}|m_s),$$

where $\mathbf{z}_{k,m}$ is the feature used for a segment of size k and sub-class m, $p(\mathbf{z}_{k,m}|m)$ is the PDF of that feature assuming sub-class m, and $p(\mathbf{z}_{k,m}|H_0)$ is the PDF of that feature under the reference hypothesis. The PPT insures that $p(\mathbf{x}_s|m_s)$ is a valid PDF, and the ME criterion assures that we have introduced no hidden assumptions.

Clearly the MR-HMM requires a large amount of front-end processing to calculate all the necessary features and PDFs for all segments (seen in Figure 1 "Available segments") assuming each possible sub-class. While the front-end processing is huge, the problem is often not computational, but how to make sense of all the outputs. The MR-HMM likelihood function is

$$L(\mathbf{x}) = \sum_{\mathbf{q}\in\mathcal{Q}} p(\mathbf{x}|\mathbf{q})\; p(\mathbf{q}), \tag{14}$$

where $\mathcal{Q}$ is the set of all possible segmentations, and $p(\mathbf{q})$ is the *apriori* probability of that segmentation. Calculating (14) is impractical because the number of possible segmentations is combinatorially large and it is unclear how $p(\mathbf{q})$ is determined. Yet, it can be easily calculated using the proxy HMM since each segmentation $\mathbf{q}$ corresponds to a distinct path through the proxy HMM state trellis. If we had the likelihood functions of each base-segment, the proxy likelihood function would be

$$L_r(\mathbf{x}) = \sum_{\mathbf{q}\in\mathcal{Q}} p_r(\mathbf{x}|\mathbf{q})\; p(\mathbf{q}),$$

where r stands for "p(r)oxy", and where

$$\log p_r(\mathbf{x}|\mathbf{q}) = \sum_{i=1}^{T} \log p(\mathbf{x}_i^b|w_i(\mathbf{q})),$$

where $\mathbf{x}_i^b$ is the i-th base segment and $w_i(\mathbf{q})$ is the proxy HMM state at time step i corresponding to segmentation $\mathbf{q}$. The well-known HMM *forward procedure* [22] calculates the total proxy likelihood function $L_r(\mathbf{x})$ efficiently using dynamic programming and without enumerating the segmentations. The trick in computing $L(\mathbf{x})$ for the MR-HMM is to use the proxy HMM forward procedure after replacing the proxy conditional PDFs $p_r(\mathbf{x}|\mathbf{q})$ by the MR-HMM conditional PDFs $p(\mathbf{x}|\mathbf{q})$. This is done by dividing the MR-HMM segment PDFs $\log p(\mathbf{x}_s|m_s)$ into k_s equal parts. We let $\log p(\mathbf{x}_i^b|w_i(\mathbf{q})) = (1/k_s)\;\log p(\mathbf{x}_s|m_s)$. The notation is a bit cumbersome, but can be explained as follows: *Choose a particular segmentation* $\mathbf{q}$. *For this segmentation, any base segment time* i *lies within a particular segment* s *of length* k_s. The classical forward procedure then produces $L(\mathbf{x})$. Furthermore, by calculating the *backward procedure* on the proxy HMM and combining with the forward procedure, we obtain the gamma probability $\gamma_i(m)$, which is the *a posteriori* probability that the system is in proxy state m at base segment i given all the available data. This is illustrated in Figure 1 as the filled-in circles in the proxy state trellis. These filled-in circles correspond to when $\gamma_i(m)$ has a high value. In the gap between the two pulses, is a case when probability is shared between more than one candidate path. If we sum up all the gamma probabilities for a given sub-class, we get an indication of the probability of each sub-class (illustrated at very bottom of figure 1).

D. Parameter Re-estimation

The parameters of the proxy-HMM ($\mathbf{A}$, $\boldsymbol{\pi}$) can be re-estimated using the standard Baum-Welsh algorithm [22]. The feature PDF estimates which are used to obtain $p(\mathbf{x}_q|q)$ from the PPT formula (10) can also be re-estimated in a straightforward way using the gamma probabilities [21].

E. MR-HMM synthetic example

For illustration, we created an MR-HMM with three sub-classes: background noise, low-frequency (LF) burst, and high-frequency (HF) burst. Feature extraction consisted of autoregressive (AR) analysis with model order depending on the sub-class. Derivation of the projected PDF (10) for AR is found in the tutorial [13]. In Figure 2 (top), we see a spectrogram exhibiting an LF and an HF noise burst. In the center of the figure are the *a posteriori* proxy state probabilities $\gamma_i(m)$. These are summed, collecting all wait states in each sub-class, to obtain the sub-class probabilities (bottom). Note that at some times, there is considerable uncertainty about which trellis path is in effect as evidenced by probability sharing. But, the path probabilities sum up to provide an almost binary decision about the sub-class.

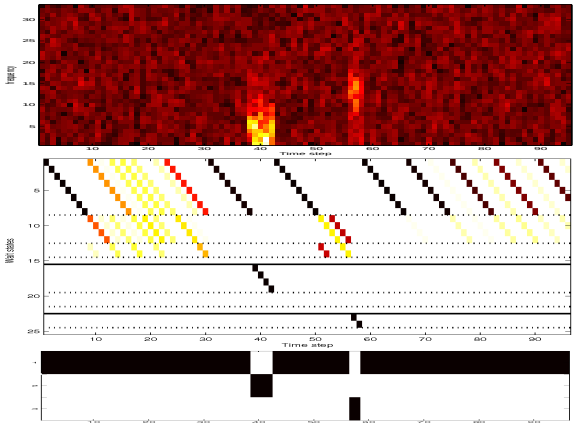

Fig. 2. Illustration of MR-HMM using synthetic data.

F. MR-HMM speech example

It is hard to find acoustic data with more diverse spectral and temporal content than human speech. Recordings of specific spoken phonemes are analogous to classes of diverse short-duration signals. We used the MR-HMM to model and distinguish the sounds "gr" and "kr", composed of the phoneme sequences "GCL-G-R" and "KCL-K-R". The modeling involved careful study of the phonetic structure including short sub-phoneme units visible only with much finer temporal resolution than used in automatic speech recognition (ASR), which was encoded into the state transition diagram (Figure 3) and state transition matrix $\mathbf{A}$. Feature extraction consisted of MEL frequency cepstral coefficients (MFCC) [23]

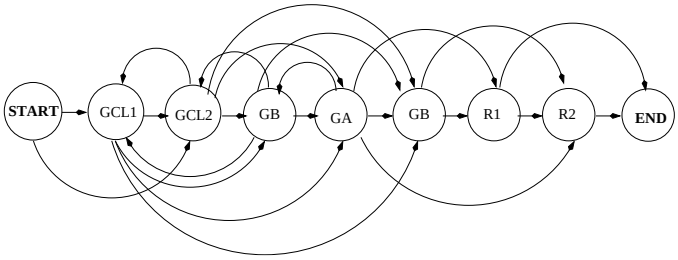

Fig. 3. State transition model for "GCL-G-R".

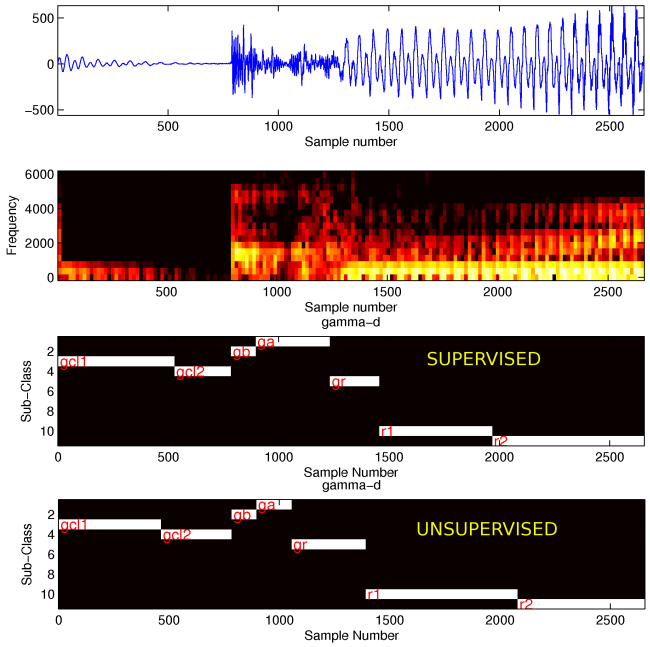

Fig. 4. Illustration of MR-HMM using speech data (12kHz sample rate).

for a variety of model orders and segment lengths. Figure 4 shows the time-series and spectrogram of an utterance of "GCL-G-R". Under that, the MR-HMM sub-class probabilities are shown. During the training phase (SUPERVISED), finely labeled data is used to force the gamma probabilities to the ground-truth. In the UNSUPERVISED output, the MR-HMM is allowed to freely estimate the gamma probabilities. As opposed to conventional methods that use a coarse time resolution, the MR-HMM gamma probabilities give precise clues to the presence and location of sub-phonetic units, useful in phonetic analysis. Using the MR-HMM, we could detect the short "burst" element of about 100 samples length, (sub-class 2, indicated as "gb" in the state probabilities in Figure 4). Standard ASR processing uses typically 30 millisecond time resolution (360 samples). Also, in classifying "DCL-D-R" vs "TCL-T-R", and "GCL-G-R" vs "KCL-K-R", using the TIMIT data corpus [24], MR-HMM provided a 33 and 27 percent error reduction with respect to the standard MFCC/HMM [3].

IV. Current Work

Work continues in extending and applying CSM.

[3]implemented using the HTK toolkit [25]

A. Data generation

While the primary use of the PPT is to calculate the likelihood function (10) for use in the classifier (9), it is also theoretically possible to generate raw data from this PDF. Fueled by the discovery of the ME connection to PPT, we have sought practical approaches to generate raw synthetic data from these ME distributions and have made significant advances in the practical realization of this data generation for a wide range of important feature transformations [19], [26]. This opens the door to many new applications of Monte Carlo and sampling methods [27], [28]. It also makes possible a theoretical framework for combining generative and discriminative classifiers into true generative models that combine the best aspects of each.

B. Generative vs. Discriminative?

The CSM variants described here are *generative* methods (GM) that model the data using PDFs, describing statistically how the data for each class is generated. They are in effect *data descriptive* models. Most current work in machine learning and classifier theory involves *discriminative* methods (DM) like support vector machines (SVMs) [29] that directly learn the mapping from data to class identity. It is widely held that DMs outperform GMs when tested head-to-head, but this is valid only when comparing them *on the same feature set*. The class-specific method has freed GMs to use different features and has created an opportunity to combine GMs and DMs in new ways. The most promising way is to operate the DM classifier using it's own feature set in a rejection mode, passing data to the CSM classifier only if accepted. Any two or more class hypotheses that are accepted will be compared based on CSM likelihood value. Because CSM is based on a raw-data GM, it is possible generate random synthetic raw data samples from the GM, then pass them through the DM to measure the acceptance rate for the synthetic data. This determines the normalization factor to use for the combined cascaded DM/GM model so that the combined model is a true GM.

C. Conclusion

We have provided overviews of CSM and MR-HMM. We've illustrated how MR-HMM can be used to model and classify short-duration signals using arbitrary data segmentation and multiple features with a single statistical model.

References

[1] A. Bregman, *Auditory Scene Analysis: The Perceptual Organization of Sound*. Cambridge, MA, USA: MIT Press, 1990.

[2] Duda and Hart, *Pattern Classification and Scene Analysis*. Wiley, 1973.

[3] J. Berger, *Statistical Decision Theory and Bayesian Analysis*. Berlin Heidelberg: Springer, 1993.

[4] R. E. Bellman, *Adaptive Control Processes*. Priceton, New Jersey, USA: Princeton Univ. Press, 1961.

[5] K. Fukunaga, *Introduction to Statistical Pattern Recognition (2nd ed)*. San Diego: Academic Press, 1990.

[6] P. M. Baggenstoss, "Class-specific features in classification." *IEEE Trans Signal Processing*, pp. 3428–3432, December 1999.

[7] H. C. B. Caputo, "A Marxist approach to object recognition: kernel-specific classifiers," in *Proceedings of the 2004 symposium on image analysis (SSBA)*, 2004.

[8] Z. J. Wang and P. Willett, "Joint segmentation and classification of time-series using class-specific features," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 34, no. 2, pp. 1056–1067, Apr 2004.

[9] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part 1, Detection, Estimation, and Linear Modulation Theory*. Wiley, 1968.

[10] D. R. Cox and D. V. Hinkley, *Theoretical Statistics*. London: Chapman and Hall, 1974.

[11] S. Kay, "Sufficiency, classification, and the class-specific feature theorem," *IEEE Trans. Information Theory*, vol. 46, no. 4, pp. 1654–1658, July 2000.

[12] S. M. Kay, A. H. Nuttall, and P. M. Baggenstoss, "Multidimensional probability density function approximation for detection, classification and model order selection," *IEEE Trans. Signal Processing*, pp. 2240–2252, Oct 2001.

[13] P. M. Baggenstoss, "The class-specific classifier: Avoiding the curse of dimensionality (tutorial)," *IEEE Aerospace and Electronic Systems Magazine, special Tutorial addendum*, vol. 19, no. 1, pp. 37–52, January 2004.

[14] ——, "The PDF projection theorem and the class-specific method," *IEEE Trans Signal Processing*, pp. 672–685, March 2003.

[15] V. Estellers and P. M. Baggenstoss, "Class-specific classifiers in audio-visual speech recognition," in *EUSIPCO 2010*, 2010.

[16] Y. Sun and P. Willett, "Automated classification of signals with duration-dependent segments via class-specific features and gibbs sampling," in *IEEE Aerospace Conference*, 2012, pp. 1–11.

[17] T. Beierholm and P. M. Baggenstoss, "Speech music discrimination using class-specific features," in *Proc. ICPR 2004*, 2004.

[18] J. N. Kapur, *Maximum Entropy Models in Science and Engineering*. Wiley (Eastern), 1993.

[19] P. M. Baggenstoss, "Maximum entropy PDF projection (submitted)," *IEEE Trans. Information Theory*, 2013.

[20] A. Y. Khincin, *Mathematical Foundations of Information Theory*. Mineaola, NY: Dover, 1957.

[21] P. M. Baggenstoss, "A multi-resolution hidden markov model using class-specific features," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5165–5177, Oct 2010.

[22] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, February 1989.

[23] P. Mermelstein, "Distance measures for speech recognition, psychological and instrumental," *Pattern Recognition and Artificial Intelligence*, p. 374388, 1976.

[24] J. S. Garofolo, "Timit acoustic-phonetic continuous speech corpus," *Linguistic Data Consortium*, 1993.

[25] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. A. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book, Version 3.4*. Cambridge University Engineering Department, 2006.

[26] P. M. Baggenstoss, "Sampling from maximum entropy spectral models," *IEEE Trans. on Signal Processing (submitted)*, 2013.

[27] B. D. Ripley, *Stochastic Simulation*. New York: Wiley and Sons, 1987.

[28] S. Weinzeirl, "Introduction to Monte Carlo methods," *Report NIKHEF-00-012, Cornell University Library*, 2000.

[29] B. Schoelkopf, C. Burges, and V. N. Vapnik, "Extracting support data for a given task," in *Proc. 1st Int. Conf. Knowledge Discovery Data Mining*, U.M.Fayyad and R. Uthurusamy, Eds. Menlo Park, CA: AAAI Press, 1995.