Contents

Automated Object Model Acquisition for Adaptive Perceptual Systems[*]

Malini K. Bhandaru   Victor R. Lesser
Dept. of Computer Science
Univ. of Mass at Amherst
Amherst, MA 01003
malini, lesser@cs.umass.edu

August 4, 1994

Abstract

For complex perceptual tasks that are characterized by object occlusion and non-stationarity, recognition systems with adaptive signal processing front-ends have been developed. These systems rely on hand-crafted symbolic object models, which constitutes a knowledge acquisition bottleneck. We propose an approach to automate object model acquisition that relies on the detection of signal processing discrepancies and their resolution. The approach is applied to the task of acquiring acoustic-event models for the Sound Understanding Testbed (SUT).

1 Introduction

Complex perceptual tasks are characterized by varying signal-to-noise ratio, unpredictable object activity and possible object occlusion. Successful object recognition depends on the extraction of adequate disambiguating features. For these highly variable, non-stationary scenarios, such features are neither stationary nor easily identifiable. To meet the challenge of recognition in such environments, *Adaptive Perceptual*

[*]This work was supported by the Rome Air Development Center of the Air Force Systems Command under contract F30602-91-C-0037 and by the Office of Naval Research under contract N00014-92-J-1450. The content does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

Systems [13, 17, 33, 37] have emerged. These systems adapt their signal processing front-end in response to variations in the incoming signal data.

Recognition in adaptive perceptual systems proceeds through the interaction of two processes: feature extraction and interpretation/matching. Feature extraction involves the application of selected signal processing algorithms (SPAs) with default parameter settings. During interpretation/matching, the extracted features are compared against *object models.* Failure to account adequately for some or all data indicates a need for SPA and/or parameter adaptation in order to extract additional/alternate features. Symbolic object models are preferred since they are accessible for purposes of data interpretation, guiding feature extraction and predicting evidence interaction. Typically the object models have been hand-crafted, a tedious and error prone activity that constitutes a knowledge acquisition bottleneck.

Automating model acquisition for adaptive perceptual systems has received relatively limited attention [37, 38, 51]. Much of this effort relies on human intervention such as suggesting alternate SPA parameterizations when feature inadequacies are encountered, or in the initialization of critical parameters. In the case of vision applications, the availability of adequately segmented images is assumed [37, 39].

The learning task we seek to address is stated as follows: *given a set of training instances (signal/label pairs), and a finite set of parameterized SPAs, to seek for each training instance SPA parameterizations that serve to extract features that enable the induction of models that are collectively unambiguous and capture the intrinsic characteristics of the objects.* To meet this goal we systematically search the space of SPAs and their parameters. To avoid an exhaustive search, we exploit both generic object models and discrepancies that occur in the course of inducing the object models to guide search. Knowledge about the SPAs, their parameters and the domain is used to reason about the discrepancies and suggest alternate processing contexts.

The learning paradigm will be discussed in the context of the SUT [32], an adaptive perceptual systems for non-speech sound recognition. The sounds, also known as acoustic-events, may occur together. Examples of acoustic events are: a foot step, telephone ring, and a hair dryer coming on. See Figure 4 for the SUT model for a hair dryer operating at high speed. Adaptivity in the SUT comes from viewing signal processing as a bi-directional search in the SPA/parameter space: low level processing is aimed at achieving signal processing results that are free of discrepancies while high level processing is aimed at finding valid signal interpretations and support for expectations.

In Section 2 we describe the ramifications of being able to vary SPA parameters. In Section 3 we briefly describe the Sound Understanding Testbed and give an example of a sound model. In Section 4 we briefly discuss learning effort in the area of model acquisition. We discuss discrepancies and their diagnosis in Section 5 before discussing in detail our learning approach in Section 6. We discuss model revision, stability, and the use of generic models in Section 8. Finally, in Section 10 we indicate the status of the work and present our conclusions in Section 11.

2 Parameterized SPAs

Distinct parameterizations of an SPA extract the same class of features, but the actual values extracted may be very different. This is because in the mathematical formulation of the SPAs, the parameters are used to capture assumptions about the underlying signal. To emphasize the relationship between the features extracted and the *processing context*: SPAs and their parameter settings, the features extracted are called *SPA-correlates*. SPA-correlates obtained under different parameterizations of an SPA are however comparable through the use of knowledge about the underlying signal processing theory that is used as the basis for the SPA implementation [33]. An SPA parameterization may expose some salient aspects of an object while obscuring others.

For instance, consider the analysis of time-amplitude waveform data corresponding to an acoustic-event composed of two constant-frequency components with an inter-component spacing of 15 Hz. With a sampling frequency of 10 KHz, a Fourier Transform based algorithm for frequency analysis would be unable to expose the relevant frequency detail unless a data window that affords the minimum required frequency resolution is used. This is illustrated using the Short-Time Fourier Transform (STFT) algorithm [40] for spectral analysis in Figure 1. Note that the uncertainty in frequency spread i.e., "width" of each component reduces when the data is processed with greater frequency resolution. The need to disambiguate among similar objects influences the search in the SPA/parameter space in the direction of extracting greater detail.

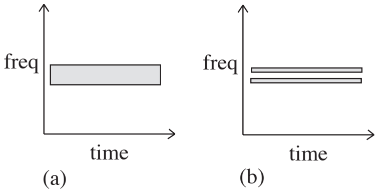

Figure 1: Signal corresponding to two closely spaced steady frequency components, processed with (a) shorter and (b) longer STFT window. Note the better frequency resolution obtained in (b) due to the longer window.

Secondly, certain SPA parameterizations afford a view of the signal data that leads to a *simple*[1]physical explanation. For instance, consider the analysis of a near-linear rising chirp[2] sound sampled at a frequency of 8KHz. If the signal data is processed using the STFT algorithm [40] with a small window and narrow decimation (128 and 64 data points respectively) before connecting peaks in consecutive spectra, we would obtain results as shown in Figure 2a. Keeping all other processing parameters constant, but using a much longer window (1024 data points), a broken curve as shown in Figure 2b would be obtained. While the former may be interpreted as a "chirp", the latter could be interpreted either as a chirp not processed appropriately or the presence of several sound sources, each of which emits a short burst of sinusoidal activity that is separated by approximately 10 Hz. Further, the latter interpretation indicates that the activity is highly synchronized in the sense that as a lower frequency source decays, the next higher one becomes active. Given the rarity of finding distinct physical events that are so highly synchronized, this interpretation requires too many assumptions making it not simple. Simple interpretations map to the notion of intrinsic characteristics of an object, originating in the physics of the excitation production mechanism. We shall revisit this concept in Section 8.3.

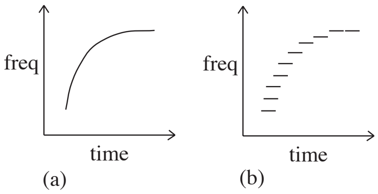

Figure 2: Semi-linear chirp processed with (a) shorter and (b) longer STFT window. Note the broken contours of (b) due to insufficient time resolution.

3 SUT

Before we go further, we briefly describe the Sound Understanding Testbed (SUT) [32] and the recognition task being addressed. The SUT seeks to identify acoustic-events given waveform data (a sequence of time-amplitude pairs). The SUT is based on the IPUS (Integrated Processing and Understanding of Signals) architecture [33] which views the process of signal understanding as a bi-directional search in the space of SPAs and their parameters. The bottom-up search for SPAs and their parameters

[1] The principle also known as Occam's Razor or the law of parsimony may be stated as follows: entities must not be multiplied beyond what is necessary, that is an argument must be shaved down to its absolutely essential and simplest terms.

[2] A *chirp* is a rising/falling frequency modulated component.

is aimed at achieving signal processing results that are free of common discrepancies, which are detectable through the application of two or more SPAs, each with distinct strengths, and comparing their signal processing results (discrepancy detection is discussed further in Section 5). The top-down search is guided by the desire to find valid signal interpretations based on expectations about the environment. Analysis proceeds with the interaction of the different special purpose knowledge sources (KSs).

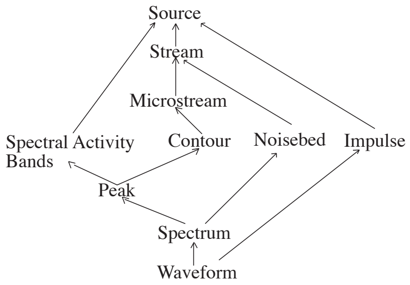

Figure 3: Data Abstraction Levels used in the Sound Understanding Testbed

The abstraction levels used in the interpretation process are shown in Figure 3. Windowed waveform data that is analyzed for its spectral content is represented at the spectrum level. Peaks are localized regions of higher energy in a spectrum. Criteria such as the absolute cut-off energy and the relative magnitude of a peak with respect to its neighbors are critical factors in determining the peaks that are selected from a spectrum. Contours are a sequence of peaks that move forward in time and share the same energy or frequency or energy-frequency trend. Noisebeds are regions of seemingly uncorrelated spectral activity. Contours that are consecutive in time and bear certain frequency and energy relationships are grouped together to form a microstream. Microstreams that are synchronized either in their onset times, energy behavior with respect to time, or whose frequencies are harmonically related, are grouped together to form streams. Groups of streams support a source level hypothesis. Periodic sources would display a repeating pattern of stream support units.

To date the SUT database consists of 37 models. The models were acquired by manually analyzing several recordings of each sound. The tediousness of the task provides the motivation for this work. For example, consider the sound produced by a hair dryer. The acoustic signal is due to the working of a motor and the forcing of air through a nozzle. The component frequencies of the sound are harmonically related with a fundamental whose frequency is that of the power line. The relative energies of the harmonics are dependent on the speed of operation and the hair dryer construction (differing for different manufacturer models). Noisebeds, which are an artifact of the air flow through the nozzle of the hair dryer, surround the primary frequency components. The operation of a hair dryer exhibits two distinct phases, the transition or chirp phase that corresponds to the hair dryer being turned on or off, and a steady phase when it is operating at either high or low speed. The processing parameters that best bring out the time frequency characteristics in the two phases are in opposition (refer Section 2).

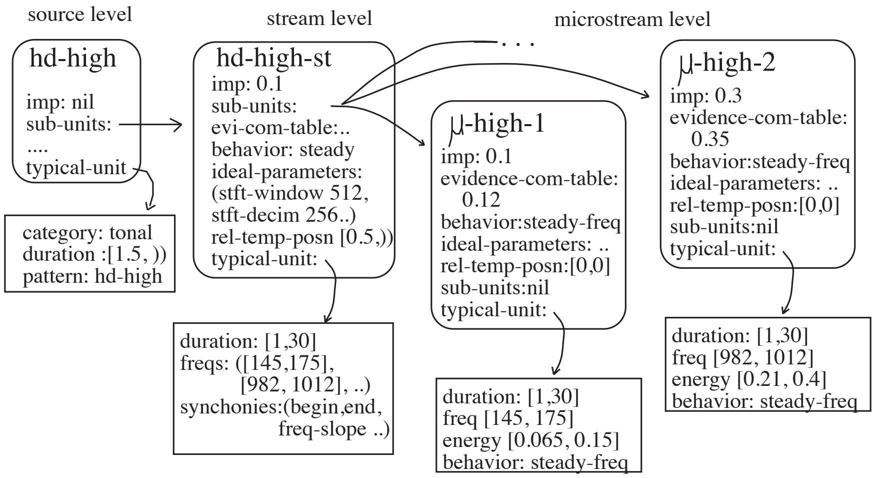

Figure 4: Acoustic Event Model for an Hair Dryer operating at a high speed.

The models are specified at the source, stream and microstream levels of data abstraction. In Figure 4, we show portions of the SUT model for a hair dryer operating at high speed. The source level unit is made of a single stream level unit: hd-high-st, which in turn is made of several microstream units, two of which (μ-high-1 and μ-high-2) are shown. Note that the microstream durations are approximately equal and since their relative temporal offsets with respect to the start time of the stream is zero, their onset and end times are said to be in synchrony. A stream level representation that captures the complete behavior of a sound source, in terms of its constituent events, is possible. For example, the hair dryer sound may be specified as:

$$HDon(HDhigh + HDlow)^{*}HDoff$$

where HDon, HDhigh, HDlow and HDoff denote the hair dryer coming-on, operating at high speed, operating at low speed and going-off events, respectively. The above representation indicates that the HDon and HDoff events are mandatory, and that the HDon event precedes in time the HDoff event. In contrast, the HDhigh and HDlow events may each occur zero or more times (denoted by the *), and in any order (denoted by the +). Our goal is to first acquire models for each of the constituent events of a sound source. Eventually we plan to extend the work to building representations that capture such complex temporal patterns.

4 Learning in Perceptual Systems

Perceptual systems that exploit learning techniques have either acquired object models or assumed the existence of such models and learned object recognition strategies. Systems that acquire object models [31, 34, 36] have predominantly employed fixed signal processing front-ends. Systems that learn recognition strategies [17, 30, 37], have instead employed adaptive signal processing front-ends. This latter class of systems dynamically select SPAs with preset parameters from a large but finite set in order to meet the needs of the recognition task at hand.

We shall examine both classes of learning systems. Apart from highlighting the gap which we propose to fill, the goal of this section is to detail the techniques used and assumptions made by the learning component in the fixed front-end systems and indicate why they are inadmissible for learning in adaptive front-end systems. In our discussion of the systems that learn recognition strategies, it shall become clear that they capture several capabilities that are desirable in the automation of model acquisition.

4.1 Learning Object Models

It is tedious to identify manually a set of features suitable for disambiguation purposes in situations where either the object class is large or the data are contaminated with noise. This provides one of the motivations for automating object model acquisition. With respect to speech recognition, the object class is swelled by the size of the vocabulary and variations in pronunciation by different speakers (in speaker-independent recognition tasks). Yet another illustration of the problem is found in the domain of vision where "pose" varies with the observation perspective. Radar and Sonar data are typically contaminated with noise. Subsymbolic approaches, such as artificial neural nets (ANNs) [49], hidden Markov models (HMMs) [48] and hybrids thereof [12] have successfully been used for object modeling in systems with fixed front-ends for feature extraction. These approaches have yielded performance improvements over hand-crafted versions. In comparison, little work has been done with object model acquisition for systems with adaptive front-ends.

Subsymbolic models have been shown to yield accurate low-level model matching. However, they are relatively inaccessible for purposes of interpretation and prediction. For tasks where multiple objects may co-occur, their representational "opaqueness" causes training data requirements to grow combinatorially: both in the number of objects that may co-occur and in the spatio-temporal relationships the objects may exhibit. Consequently, such systems are not well suited to recognition tasks for complex scenarios. Where ANN techniques have been used for recognition in scenarios where objects may co-occur, the operating assumption is that the individual signal contributions and/or features of each object are separately available. For instance, in vision this amounts to requiring that segmented data be provided to the ANN classifier. In contrast, adaptive perceptual systems seek to adapt their signal processing front-end to facilitate data segregation and ensure reliable object identification.

4.1.1 Subsymbolic Approaches: HMMs and ANNs

In this section we describe several successful subsymbolic approaches for object modeling. These approaches are distinguished in their use of fixed front-ends for feature extraction, use of vector quantization[3](VQ) [23] techniques for data compression and the large number of object classes that they model. In particular, we discuss subsymbolic object model acquisition systems that use HMM and ANN techniques.

HMMs in Speech Recognition: Hidden Markov Models [31, 45, 48] have been predominantly used in speech recognition. They are a generalization of dynamic programming and provide a rigorous approach to developing robust statistical models. In HMMs, the view of measuring acoustic similarity as a template matching problem is generalized to a problem of finding an optimal path (using the Viterbi algorithm) through a recognition model [31]. The probability of the speech data given a recognition model is computed. The major advantage in HMMs, from a time normalization perspective, is that a local constraint function can be re-estimated, or optimized, by an iterative training procedure. Re-estimation allows the assimilation of the statistical characteristics of the training data, in turn optimizing performance on the training set. We present below a brief description of several HMM based recognition systems and indicate their common features.

Tangora [3], designed for speaker-independent isolated word recognition, handles vocabulary sizes of 5,000 to 20,000 words. It uses a VQ front-end, operating with a single codebook of 200 elements and uses word units for training. The SPHINX [31] system for speaker-independent connected speech recognition handles 997 words. It uses VQ, maintaining three codebooks of 256 prototype vectors each (for cepstral coefficients, differenced cepstral coefficients and power with differenced power). SPHINX uses generalized triphones [4] for the basic training units and exploits grammar con-

[3] In Vector Quantization (VQ), clustering techniques are used on the data to identify a finite number of representative elements that reduce some error measure. These are then saved in a codebook for future use where actual data elements are replaced by the most similar element in the codebook. Data compression results from the fact that a finite number of vectors or elements has fewer representational requirements.

[4] A triphone is a sequence of three phones, and captures significant co-articulation effects. With 50 phones in the English language, the set of possible triphones is very large. However, the high degree of similarity among them enabled the use of clustering techniques to achieve data compression to yield a smaller set of generalized triphones.

straints from the limited application domain to aid in the recognition task.

The AT&T digit recognition system [47] and the Texas Instruments system [14] are word-based[5] speaker-independent digit recognition systems. In the digit recognition task, grammar constraints are minimal: any digit may appear after any other. This places a greater need for good acoustic matching to achieve high performance. In the Texas Instruments system, a discrimination transformation was designed to maximize discrimination between the correctly recognized data and the confusion class for each state in the HMM word model.

ANNs in Perceptual Systems: ANNs [49] have been extensively exploited in recognition tasks chiefly because they are capable of representing a variety of statistical properties of data distributions in an automatic manner. They are good at generalization and in the retrieval of stored patterns that most closely resemble an input pattern.

The hidden nodes in ANNs often become feature detectors and differentiate between important classes. That is, they capture higher order features that are combinations of the primitive features. Though this aspect frees the user from a need to construct optimal features, it still requires that quality primitive features be provided.

Speech recognition in particular is complex due to several factors: high degree of variability and overlap of information in the acoustic signal, need for high computation rates, multiplicity of analyses that must be performed (phonetic, phonemic, syntactic, semantic, and pragmatic), and a lack of any comprehensive speech theory. ANNs have resulted in improved performance on subtasks of the speech recognition task and on the overall task. Lippman [35] provides a good review of the state of the art in using ANNs for speech recognition. ANNs have also demonstrated good performance for Sonar [1, 36] and Radar [26] domains.

The ANN approaches accept as input a fixed set of features and use one of several training techniques [49] for inducing a classifier. Several net architectural parameters such as the number of input and output nodes, training method, number of hidden layers, number of nodes within them, and learning rate require careful selection. Classification performance is sensitive to these parameters. Several approaches for automatically selecting them for a given classification task are currently available [2, 28]. The methods chiefly differ in whether they converge from a larger number of nodes and layers to a smaller set or work the other way round.

[5] A speech recognition system is termed word-based when the basic training units are words.

4.1.2 Symbolic Object Models

In this section we discuss symbolic approaches to acquiring object models, which compared to the amount of work on subsymbolic approaches is limited. Murase and Nayar [39] acquire object models that do not use multiple levels of data abstraction for visual object recognition. Vadala [51] developed a semi-automatic approach for acquiring sound source models requiring multiple levels of data abstraction.

Adaptive Front-End: Vadala's [51] semi-automatic approach for sound source model acquisition is based on the principles of the SUT [32, 33]. It requires user guidance based on "viewing" and "listening" to the acoustic signal to estimate the key SPA parameters such as the STFT window length and number of peaks to select from the spectra. To ensure that sufficient frequency resolution is obtained, peaks were identified based on the depth of inter-peak valleys as a percentage of the peak height. The peaks were grouped into time frequency energy patterns or contours using a least squares line fit[6]. Sound source models were built on the lines of those used in the SUT (refer Section 3).

Several interesting experiments were carried out to show how the ideal evidence gathering procedure varied with the type of noise added to the signal. For example, with the addition of white noise, the window length had to be longer than when the signal was not contaminated to enable the extraction of salient time frequency energy information. When objects may co-occur, the presence of one could potentially affect the processing of another. A possible extension to an automated model acquisition system could be categorizing objects on the basis of their effects on the processing of others.

Vadala applied his technique to "ringing" sounds, in particular to four different telephones, a bicycle bell and a burglar alarm. Vadala also derived a general telephone model by combining the individual telephone models. Our intention is to extend his work to more objects, and fully automate the model acquisition process.

Fixed front-end: We shall next discuss two successful systems [39, 37] for object model acquisition in the visual domain based on fixed signal processing front-ends. The front-ends are characterized as fixed because the image segmentation is fixed and the set of features available for model representation is fixed.

Visual Appearance Murase and Nayar [39] address the problem of learning object models from images for object recognition and pose estimation. They formulate

[6] Our experiments in the SUT indicate that the least squares fit contouring criterion is very sensitive to outliers, i.e., a few points that do not conform to the majority behavior.

the problem as one of matching visual appearance rather than shape. For each object, a large set of images is obtained by automatically varying pose and illumination, which is then compacted using principal-component techniques [20]. The resulting lower-dimensional representational subspace, called eigenspace, is parameterized by pose and illumination. Each object is represented as a hypersurface in this space.

Given an unknown input image, the recognition system projects the image onto the eigenspace. Object identity is established by the hypersurface the projection meets. Object pose is determined by the location of the projection on the hypersurface. Such a representational approach is especially useful when the objects of interest do not conform to CAD wire-frame models, often the case with most naturally occurring objects.

This work assumes that the recognition system does not have to handle image segmentation. Partial object occlusion is not addressed. Consequently, a fixed signal processing front-end is sufficient and the learning subsystem concerns itself only with variations in object illumination and pose in the construction of robust models.

Symbolic Descriptions Ming and Bhanu in TRIPLE [37] combine structured conceptual clustering (SCC) and explanation based learning (EBL) techniques to acquire and refine object models for aircraft recognition. New object models are acquired as and when the objects are encountered and models revised when better training data becomes available.

The modeling process in TRIPLE begins with segmenting the images using a fixed segmentation procedure. Target related regions are next identified and region labeling performed. Simultaneously, a hypothesize and test approach is used to determine target orientation. A fixed set of symbolic features, such as, fuselage length, wing span, and wing sweep angle are next extracted, using sets of production rules, from the region borders. Object models are defined in terms of these features whose values may be ranges. A classification tree is built from the models using SCC, which is used by both the recognition and learning routines. During tree traversal, if a leaf node is reached, recognition is said to succeed. The opposite, indicates that a new, or very distorted object has been encountered and requires further investigation. The EBL component exploits knowledge about valid feature relationships to decide whether the features extracted indeed represent a new target or whether the data is too distorted or the knowledge base incomplete to make a valid conclusions. The EBL component examines new features that are detected in an object model for relevance. The classification tree is modified to include the newly detected relevant features. Model revision is gradual - feature values are moved in unit increments along the direction of the newly acquired target features. TRIPLE has successfully automated construction of a model database of aircrafts using noise-distorted technical drawings of the same.

One of TRIPLE's limitations is that it does not handle scale, that is, it assumes that all images are of objects at the same scale. TRIPLE also does not address the issue of images that may require different segmentation procedures to adequately extract target regions and boundaries. The constrained domain of aircrafts allows the use of knowledge regarding feature interactions to ascertain whether an image of a target and the features extracted therefrom indicate the presence of a new object or the loss of significant detail or inadequacies in the knowledge base. This may be an over simplification for less constrained domains.

4.2 Learning Recognition Strategies

Systems that attempt to build efficient recognition strategies adapt the feature extraction process to the recognition goal. The strategies so generated are computationally efficient as a result of extracting and examining only features that are deemed necessary. The generated recognition strategies, i.e. sequences of features to be extracted along with their expected associated values, are compiled for speed.

These systems differ in whether they assume the existence of explicit or implicit object models for matching purposes. An explicit object model is a localized handcrafted description of the object of interest. Implicit models constitute object descriptions that are distributed in declarations, rules and/or procedures. Common to both kinds of object models is that they are accessible for interpretation and prediction purposes, i.e. *transparent.* The advantage of transparent representations is that they can be used to direct feature extraction for meeting the recognition goal.

4.2.1 Algorithm Generation Approaches

Predominantly for the visual domain, algorithm generation approaches [9, 17, 27, 30] have been explored. Their objective is to acquire recognition strategies given a high-level task description such as to assert the presence or absence of an object in an image or determine its pose. Such strategies are referred to as algorithms/programs since they specify not only the exact features to use, but also the order in which they are extracted. These systems adapt feature extraction to the recognition task in the sense that they appropriately select a subset from a large but fixed set of features.

The recognition strategies in many of these systems are basically search trees, constructed by applying a set of rules that determine relevant features of the unknown image in order to establish object identity. The SLS [17] uses decision trees [46] to represent recognition strategies. Common to all the systems is that they address issues of feature selection and ordering. In addition, they determine criteria for hypothesis pruning (terminating feature matching at some branch of the search tree), which corresponds to rejecting or accepting an object hypothesis represented by a branch.

For active recognition tasks where the user specifies a high-level goal, recognition strategies provide an efficient computational approach. Passive recognition tasks are defined as requiring to identify objects as and when they occur. Examples of passive recognition tasks are: identifying sound sources as they become active in an environment, and identifying all the objects in an image. In contrast to active recognition, for passive recognition, the ability to reason with the object models would be more advantageous. For instance, it would enable sharing the effort involved in confirming and eliminating various candidate hypotheses.

4.2.2 Recognition as a Planning Task

Mori et al [38] address the task of identifying speaking modes or styles within short time intervals. Information about speaking modes can be used to achieve better performance in connected speech recognition. They test their approach on a particularly difficult subset of the spoken alphabet, namely the E1 set: (B, C, D, E, G, K, P, T, V, 3). Knowledge about speaking modes provides clues about the sound class for different speakers. For example, in one speaking mode, formant transitions may provide high discrimination power for a class of sounds, whereas in another, broad energy transitions might provide better discriminating power for the same class of sounds. Different speakers tend to use different speaking modes based on their education, anatomy and mood.

Rules along with preconditions are learned to achieve the recognition task. Also learned is a table that captures statistical information about which sequence of feature extraction algorithms provides the best discrimination under various circumstances. Knowledge about speech recognition is distributed in procedures conceived as perceptual plans. Recognition strategies could use invariant properties, when known to be reliable, obtained using statistical and/or speaker normalization techniques. Plan actions may either result in the creation of object hypotheses or the extraction of additional features.

The model acquisition process consists of first extracting some default acoustic properties and generating a description, which is a sequence of sets of acoustic properties. Based on initial experimentation the description is generalized. Further experimentation is carried out to gather statistics about the classification ambiguities due to the generalization. Plans to extract alternate/additional acoustic properties (features) are proposed based on speech recognition knowledge, so as to refine the description. Descriptions that do not meet certain acceptability criteria are discarded. The above process is repeated until the induced description performs satisfactorily on a large population of speakers and several speaking modes.

As with all incremental learning systems, a description may be rendered inadequate with the arrival of new instances. Mori et al [38] address this issue by first attempting to specialize the description and if that fails to meet classification needs, then extract additional features from the speech signal. The implementation relies on significant user interaction, requiring user guidance about features to extract when the disambiguation goal is not met. Mori et al collect statistics during training to build a table of classification ambiguities and their disambiguating features to realize greater recognition efficiency. The acquired object models are implicit in the sense that they are distributed in disambiguation plans.

4.2.3 Recognition as an Interpretation Task

PREMIO [7] maintains explicit models of the objects and the world. The sensors (characterized in terms of position and resolution) and light sources are modeled as part of the world. In addition, PREMIO maintains models of physical processes that could cause errors in feature matching. An object model in PREMIO consists of a CAD wire-frame model along with information about surface characteristics (color, reflectance) for each face. Further, a hierarchy of six data abstraction levels is used in object representation. Relational information for the sub-parts of an abstraction unit are also maintained. World and object models are used in conjunction to predict object appearance under different viewing contexts. Acquiring a recognition strategy involves predicting and comparing object appearance against the input image.

Especially interesting is that PREMIO addresses reasoning about models (of objects and the world) to make predictions. This is very useful while addressing recognition for complex scenarios where multiple objects may co-occur. Signal understanding is addressed in a similar fashion in the SUT [33] framework.

4.3 Limitations of Existing Approaches

The existing systems fall broadly into two categories: those that learn object models and those that learn recognition strategies. However, for the purpose of discussing their limitations, we categorize them based on whether they deal with symbolic or subsymbolic object models. The main limitation of the recognition strategy learning approaches is that they assume symbolic object models are available. The approaches that seek to acquire symbolic object models for adaptive front-end recognition systems rely heavily on human guidance. The subsymbolic approaches have several limitations: fixed signal processing front-end, need for hand-selected features, high training data requirements, lack of representational accessibility for purposes of reasoning by the recognition engine, and the VQ front-end.

4.3.1 Limitations of Subsymbolic Approaches

In this section we discuss the limitations arising of the fixed signal processing front-end in systems that use ANNs, HMMs or hybrids thereof, to induce classifiers for object recognition. In particular, we shall take a more detailed look at the limiting factors of hand-selecting features, need for large amounts of training data, a sub-symbolic representation and the VQ front-end.

Fixed Front-end and Feature Selection: Systems that learn object classifiers for fixed front-end recognition systems are constrained to use a fixed set of features, that is a fixed set of SPAs with preset parameters. The learning algorithms partition the feature space into distinct regions corresponding to different classes of objects. Classification success is directly dependent on whether the selected features are sufficient to distinguish among the classes. Consequently, a lot of care goes into the selection of these features. For instance, Luse et al [36] investigated a whole class of signal processing algorithms: Gabor Wavelets, Generalized Time Frequency Representation(GTFR), Choi-Williams distribution, Fourier Power Spectra, Higher-Ordered Spectra, and the Wigner-Ville distribution, before concluding that the GTFR best suited their classification task. Though all the SPAs take the same input signal and measure the same physical quantities, their results vary due to their distinct mathematical formulations and parameter settings.

The issue of selecting adequate features is further complicated while monitoring environments with time varying characteristics, either due to unpredictable extrinsic or intrinsic object activity. This is because the unpredictable nature of activity provides little guidance in selecting SPAs/parameters. Consider two or more sound sources that are simultaneously active. Even if they display no inherent temporal variations, they may require very different SPAs and/or processing parameters to capture adequately their characteristics, which complicates signal processing. A simple example to illustrate this would be the superposition of two sound sources, one of which was sharply increasing in frequency with respect to time and the other was composed of closely spaced frequency components. Simultaneous good frequency and time resolution would be required, which is not afforded by using any single set of parameters for any of the available SPAs. To generate reliable object hypotheses it may be necessary to selectively combine, through the use of domain knowledge, SPA-correlates with distinct precisions for quantum mechanically related dimensions. This is being explored in the SUT [33].

Intrinsic unpredictability arises in objects that display time variant behavior (the ring of the telephone, sound of a vacuum cleaner in use). To increase the likelihood of obtaining a distinguishing set of features, some systems use a large set of SPAs and parameters. The drawbacks of such an approach are: a need for greater computational resources (to compute the additional features), redundant, possibly confusing features for the learning element and a need for more training examples to cover the expanded feature space. Others instead restrict their domains of applicability to tasks for which the hand-selected features are known to be sufficient.

To conclude, the fixed signal processing front-end assumption that forms the basis of subsymbolic object model acquisition systems does not meet the needs of recognition for complex environments. It suffers from a need for carefully hand-selected features.

Classifier based Training Data Requirements: The subsymbolic learning algorithms tend to have several degrees of freedom. In the case of ANNs the parameters that may be varied are: number of input, output and hidden nodes; number of hidden layers, pattern of node inter-connections, learning rate, weight training algorithm, and the initial thresholds and connection strengths. Likewise, HMMs are also characterized by several degrees of freedom, which include the training unit, the probability density functions for the output symbols along the transition arcs, the transition probabilities and their initial estimates.

The higher the degree of freedom, the greater the need for training data to establish them adequately [31, 49, 2]. When labeled training data is relatively sparse, symbolic approaches may be more appropriate.

Training Unit based Training Data Requirements: Where the basic training unit may be shared among the different object classes, training data requirements are reduced. For instance, in large vocabulary speech tasks that use words as the basic unit, several instances of each word have to be presented, placing greater demands on computational and storage requirements. The problem is further magnified by variations in pronunciation in speaker-independent speech recognition tasks. To circumvent these large training data requirements, the SPHINX system uses generalized-triphones for the basic training unit. Identifying such a basic unit for training requires deep domain knowledge in addition to being a trial-and-error process. For complex scenarios that call for adaptive feature extraction, the task is even harder.

Lack of Transparency: When addressing recognition for complex scenarios, it is highly likely that only partial clues are available to indicate the presence of some objects. Support evidence for the object hypothesis may be masked by the other active/present objects or rendered indistinguishable due to inappropriate SPA/parameter settings. Further processing may be necessary using alternate SPAs and parameters to extract additional information. For instance, to identify the characteristic rise/fall frequency region of "chirp" type sounds, the STFT algorithm may be used provided its window length and decimation parameters are kept small. Often data may need to be selectively reprocessed to establish the presence or absence of specific pieces of support evidence. Symbolically represented objects allow access to information that may be used to guide this manner of reprocessing. In particular, when objects may co-occur, they are useful in predicting evidence interaction.

In contrast, subsymbolic object models are relatively inaccessible to such interpretation and prediction tasks. Gallant [22] extracts rules from a connectionist-based expert system. Compared to the reasoning power that must be captured to address recognition for complex perceptual scenarios, the expert system is simple. Further advances are definitely required before the knowledge captured in subsymbolic classifiers can be exploited to guide interpretation.

Vector Quantized Data: Data intensive recognition systems commonly apply vector quantization(VQ) to achieve data compression. VQ is applied either directly to the signal or to the output of one or more SPAs The vector quantized data then forms the input to other stages of the recognition system. Such systems assume that the signal information being quantized originates from a single object of interest. When two or more objects may co-occur the signal data must first be partitioned into their respective contributions, which is a nontrivial in itself and constitutes a large part of the recognition effort. VQ techniques if applied directly to the combined signal, would entail in training data requirements that are combinatorial in the number of objects that may co-occur and in the possible spatio-temporal relationships they may display.

The issue of partitioning data into groups or regions is an important first step of any recognition system. It is known as the *segmentation problem* by vision researchers, *speaker segregation* by speech researchers and *component grouping* in the SUT [33]. Parson [43] uses harmonic selection to separate speech from the interfering speech of a second talker before applying VQ based speech recognition techniques. Weintraub [53] uses a computational model for separating speech based on the following: pitch, periodicity, possible rate of change within a frequency channel and over consecutive channels. Ming and Bhanu [37] present an adaptive approach to image segmentation. In the SUT, psycho-acoustic criteria [6] are used in grouping frequency components. The adaptive front-end seeks to expose and separate adequately the various components.

Where multiple objects may co-occur, a large portion of the recognition effort, in particular that involved in separating the signal components based on their source of origin, must first be completed before VQ techniques are applicable. Recognition systems with adaptive front-ends dynamically adapt their signal processing to meet the needs of the scenario. The feature vector descriptors that are variable both in their length and in the actual features themselves, do not fit the VQ paradigm. One possible way to work around this constraint would be to agree upon a maximum length feature vector and pad it as necessary for different objects. This would however dilute the representational power of the VQ technique and the benefits thereof. A second approach would be to use VQ techniques at higher levels of data abstraction, that is after some level of data interpretation has been attempted. If the interpreted data is chiefly non-numeric, a loss in the benefits could result. This is because the technique has demonstrated its power mainly for numeric data. VQ based models at higher levels of data abstraction would be beneficial only if they could also meet the interpretation and prediction needs of recognition systems for complex scenarios.

4.3.2 Limitations of Symbolic Approaches

Learning Recognition Strategies - Existence of Object Models: Systems that acquire recognition strategies assume the availability of either implicit or explicit object models. The subsymbolic object models that are currently acquirable do not lend themselves to interpretation and prediction, a pre-requisite for recognition strategy learning approaches. Consequently, such systems have relied on hand-crafted symbolic object models, and as a result they have only partially addressed the knowledge acquisition bottleneck.

Adaptive front-ends and Human Guidance: Vadala's [51] system for acquiring sound source models, as an initial first step in the modeling of each sound, seeks significant human guidance to adapt some key SPA parameters. In addition, a priori the number of distinct object classes have to be specified. The system built by Mori et al [38] for disambiguating elements of the "E" set, in the event of failure to disambiguate among possible object classes seeks human guidance in the form of further features to explore. These are then extracted and the process of acquiring recognition strategies continues. Adaptive front-ends are characterized by their potentially large feature space (a feature is identified not only by the physical quantity measured but also by the SPA and its parameters settings) and human guidance is sought to constrain the search. These systems essentially require a means to automate this search.

4.3.3 Brief Summary

Learning systems developed for perceptual tasks fall basically into two categories: those that learn object models and those that learn object recognition strategies.

The majority of the systems that learn object models use a fixed set of features in classifier induction. Such an approach is inadequate for complex scenarios that are characterized by varying signal-to-noise ratio, possible object masking/occlusion and unpredictable object activity. Such scenarios require the power of adaptive feature extraction to give improved recognition performance.

Exceptions to the above are the systems of Mori et al [38] and Vadala [51], which cope with adaptive feature extraction, learning recognition strategies and object models respectively. Both systems however require user interaction; the former requiring assistance when a classification ambiguity is detected and the latter requiring assistance in the initial selection of the SPAs and some of their parameters.

The systems that learn recognition strategies assume the availability of either implicit or explicit object models. Consequently, they only partially address the knowledge acquisition bottleneck. However, they adapt feature extraction to the disambiguation task (as specified by the object models), a desirable feature for addressing object recognition in complex scenarios.

From the above discussion, it is clear that we need a paradigm that incorporates the power of adaptive feature extraction into an object model acquisition system to cope with complex perceptual recognition tasks. We need a mechanism to automate the identification of features which facilitates the object modeling task.

5 Discrepancies and Diagnosis

We briefly discuss discrepancy detection and diagnosis [33], which form the backbone of our learning approach. Discrepancies fall into three categories: **data-data**, **data-expectation**, and **violation**. The key is to use these different discrepancies to control search in the SPA/parameter space to induce more efficiently appropriate symbolic descriptions of the objects.

When the signal processing results obtained from the application of two or more distinct SPAs from a family of functionally-similar SPAs are contradictory, we have a **data-data discrepancy**. It indicates a need for SPA/parameter adaptation. For instance, Bitar et al [5] describe the use of the Pseudo-Wigner Distribution(PWD) [10] in conjunction with the STFT [40] to detect time resolution inadequacies.

A **data-expectation discrepancy** is encountered when the signal processing results do not support our expectations. It indicates that either the expectations are invalid, or that the data processing (SPAs and/or parameters) is inappropriate. Expectations in the context of learning could be invalid either because the selected generic model (refer Section 8.3) used was inappropriate or the data was inappropriately processed. While the latter is correctable through data reprocessing, the former occurs when either the generic model database does not contain a single model or combination of models that fits the data. For instance, consider a database that contains a hair dryer generic model that gets retrieved based on certain clues present in the initial data processing. However, let us assume that the signal data being analyzed originates from a fan. Obviously expectations for noise components as a result of air being forced through a nozzle will not be detected in the fan data, giving rise to data-expectation discrepancies. Where the generic models are less specific, the actual data analysis may indicate pockets of data that do not fit the model but are nonetheless part of the signal. Generic models are not expected to account for all the data, just capture certain intrinsic characteristics of the data.

During learning, when additional instances of an object are encountered, expectations generated based on the currently available model may be invalid or imprecise, a consequence of inadequate training. The model earlier acquired may be from an instance that was incompletely processed or significantly different from the current instance. In such cases, however, the models improve with additional training.

Violation discrepancies by virtue of our interpretation can be of two types. During the learning phase, if a newly created object model encompasses one or more other models in the database, a violation discrepancy is said to occur. It indicates that one or more object models require refinement/specialization through data reprocessing. During the recognition phase, if despite the removal of all other detectable discrepancies, there exists pockets of data that do not support any known object model in the database, a violation discrepancy is declared. It flags the presence of a new object that has not been encountered earlier; an object that requires modeling. The mechanism in effect supports unsupervised learning. Alternately, a violation discrepancy could indicate that the data-data discrepancy detection mechanism is "incomplete", providing feedback regarding aspects of the domain knowledge that require strengthening.

Diagnosis involves seeking the cause of a discrepancy in order to propose an alternate, more appropriate processing context. A more goal directed version of diagnosis is **differential diagnosis** in which competing object models are examined to determine pieces of support evidence that must be sought in order to state categorically whether one or more of them are present.

6 Model Acquisition Algorithm

The model acquisition algorithm is presented in Figure 5. Each training instance consists of the object label and the signal file that contains the time-amplitude waveform data. The first step involves the initialization of the processing context using past experience where available when the learning is supervised.

The main modeling loop seeks to resolve processing and interpretation discrepancies at successively higher levels of data abstraction (most perceptual systems [18, 24, 33] maintain data at multiple levels of abstraction). This involves processing the data, checking for discrepancies, diagnosing the same and reprocessing the

```
For each training instance:

1. initialization:
 a. Set initial processing context using either
     default settings, past experience (generic
     models or earlier encountered instance models)
 b. Any expectations? Set them up.

2.  loop:
 a.  Progressively remove processing and
      interpretation discrepancies at higher
      and higher levels of data abstraction.
 b.  attempt model integration into database
       ambiguity -> seek new processing context
                         goto step 2a.

3.  next instance
```

Figure 5: Algorithm for acquiring Acoustic-Event models for the SUT.

data after adapting the SPAs and their parameters accordingly. Resolving a discrepancy at a given level in the data abstraction could entail data reprocessing at one or more lower levels of data abstraction. Further, discrepancies involving quantum mechanically related dimensions cannot be simultaneously removed [21], e.g., time and frequency. Under such circumstances it becomes necessary to resolve each individually and combine the information, through the use of domain knowledge, to generate a *composite model* that meets our interpretation needs. Processing assumptions are explicitly maintained in order to reason about the uncertainties they introduce.

When a discrepancy at the highest level of data abstraction occurs, it indicates that the object model generated either subsumes or is subsumed by one or more of the other models into the database. When no such discrepancies are detected, it implies that the new model may be included in the model database. Once this is accomplished the next training instance is addressed.

The algorithm is incremental in two respects, firstly object classes may be trained for as and when they are encountered and secondly, the system incrementally refines the object models when either additional training instances of an object class become available or instances of similar but distinct object classes are encountered. Before we discuss issues of search effort we present an example.

7 Example

The following example describes the sequence of events that would ensue when modeling two very similar acoustic-events that have identical representations with respect to the default processing context. The acoustic-events are modeled as a synchronized

[6] set of frequency components [11]. We shall discuss repercussions on search effort and model database consistency based on the availability of previously encountered training instances for purposes of re-use. However, through our choice of example, we avoid discussion of composite model generation.

7.1 Learning: Acquiring Object Models

Input: Two synthetically generated sounds: Simple-Steady-1 and Simple-Steady-2, each composed of a single frequency component, at 1010 and 1018 Hz respectively. Data sampled at a frequency of 10 KHz. See Figure 6.
Default Processing Context: STFT algorithm for spectral analysis with a window length of 512 data points. Given that the data is sampled at 10 KHz, a frequency resolution of 19 Hz is obtained.
Assumption: The model database is initially empty.

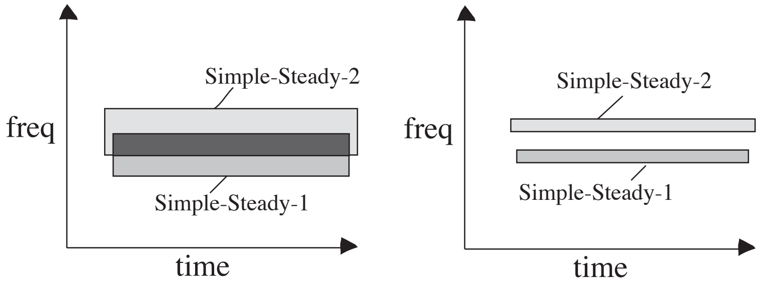

Figure 6: Model Acquisition through iterative signal reprocessing.

When the signal data for Simple-Steady-1 is analyzed using the default processing context, the object model generated indicates a single frequency component whose frequency lies in the range [1000, 1020] Hz. As is to be expected, no conflicts are encountered while integrating this model into the initially empty model database. When the training instance for Simple-Steady-2 is encountered, the initial model generated indicates a single frequency component in the range [1008, 1028] Hz. While integrating this second model into the database, a violation discrepancy is detected; the two models being virtually indistinguishable in the range [1008, 1020] Hz. Note that energy is normalized within a model in order to generalize with respect to "volume"; only the relative energies of the frequency components are preserved.

Based on the available domain knowledge, the discrepancy could be diagnosed either as: insufficient frequency resolution with consequent loss in frequency detail, or too high an energy threshold that resulted in undetected weak frequency components, or insufficient time resolution that lead to obscuring of distinguishing time behavior, or combinations of the same. Increased time or frequency resolution may be coupled with increased energy sensitivity (by reducing the energy threshold). Let us say that the control mechanism elects to pursue increased frequency and energy resolution for which the diagnosis mechanism recommends increasing the STFT window by a factor of two (to 1024 data points), and a lowering the energy threshold by 10%.

If training data is available for re-use, it is possible to reprocess the signal data corresponding to both Simple-Steady-1 and Simple-Steady-2 using the new processing context. In this case, however, the new frequency resolution obtained: 10 Hz, is insufficient for the given disambiguation task. Only on the second iteration, when the data is reprocessed with an STFT window of 2048 data points, is sufficient frequency resolution obtained. The models so obtained for Simple-Steady-1 [1008, 1013] and Simple-Steady-2 [1016, 1021] are unambiguous and may be included in the model database. Had there been a reprocessing limit such as not to exceed an STFT window length of 1024 data points, modeling failure would have occurred. If instead computational resources had been expended on obtaining increasingly better time resolution, the time frequency trajectory would have increased in width resulting in modeling failure.

If training instances were not available for re-use, the model disambiguation criterion used in guiding the search process would have been weaker. In particular, only Simple-Steady-2 would have been available for reprocessing. The reprocessing effort would have terminated once predetermined resource bounds were exhausted. The database would have remained inconsistent until such time as another instance of Simple-Steady-1 was encountered. To avoid the space requirements of saving all training instances and yet balance the time required to generate a consistent model database, a fixed number of "representative" instances of each object class could be maintained for purposes of re-use.

7.2 Another Scenario

From the above example one may wonder whether modeling would have been a one-shot process if instead the best possible frequency resolution had been used. In this section we illustrate a situation to show otherwise.

Input: Two synthetically generated sounds: Simple-Steady-1 and Simple-Chirp as in Figure 2a.
Default Processing Context: as before.
Assumption: The model database is initially empty.
The only difference with respect to the previous example is in the second training instance, in this case Simple-Chirp as opposed to Simple-Steady-2. The processing and as a result the model generated for Simple-Steady-1 would be just as indicated in the first example. However, the model generated for Simple-Chirp using the default

processing parameters would mimic that of Figure 2b. Though this model introduces no ambiguities in the database, such as on the lines of the initial models for Simple-Steady-1 and Simple-Steady-2 in the first example, the model for Simple-Chirp does not meet our *Simplicity* criterion. The discrepancies are resolvable only through reprocessing the signal data for Simple-Chirp with higher time resolution (obtained by lowering frequency resolution).

7.3 Recognition: Use of Object Models

Input: Waveform data comprising both Simple-Steady-1 and Simple-Steady-2 starting and ending at the same time.
Default Processing Context: same as before.
Assumption: Model database contains only the models for Simple-Steady-1 and Simple-Steady-2.

We examine the sequence of events that would ensue during a SUT recognition run operating in Configuration II [33], which is very similar to that illustrated in Figure 1. The spectral data obtained on STFT analysis is grouped over time into spectral-activity bands, which are used to index into the model database. For the given scenario, spectral activity is restricted to a single band that indexes both Simple-Steady-1 and Simple-Steady-2. To disambiguate among the alternatives, the differential-diagnosis component (refer to Section 5), based on an examination of the respective object models suggests data reprocessing using an STFT window of 2048 data points (to obtain the necessary frequency resolution). When the data is so reprocessed, the respective frequency components of Simple-Steady-1 and Simple-Steady-2 are identified, conclusively establishing the presence of each.

Alternately, if the object model for Simple-Steady-1 had not been revised due to a lack of availability of signal data, after accounting for the frequency component characteristic of Simple-Steady-2 there would still be data that was unexplained. However, the model for Simple-Steady-1 can be revised only when additional training instances of the same are encountered.

8 Modeling Effort

The space of possible SPA settings may be large even for small sets of SPAs where each is governed only by a few parameters. This is especially true when the parameters may take on a range of values (integral or real). Each point in the parameter space of an SPA constitutes a signal processing strategy, giving rise to a "view" of the signal, henceforth also called an SPA-correlate. As discussed earlier, object modeling involves the search for SPA parameterizations, which yield views of the signal that capture its intrinsic properties.

In this section we discuss the uncertainty associated with a model, search strategies that generate "stable" models, and the use of generic models to reduce search effort. We briefly take a look at model revision and the implications of providing less supervision to the learning element.

8.1 Model Uncertainty

Each SPA parameterization when used to process a signal, gives rise to a "view" of the signal at the corresponding data abstraction level. With any limited search of the space of SPAs and their parameterizations, only a finite number of views become available. As a result, an object model constructed on the basis of the views obtained is uncertain to the extent that all salient views may not have been exposed. Whereas it is impossible to explore the whole search space, it is however possible to annotate the models with sources of uncertainty (SOUs), each indicating a region of the search space unexplored.

For instance, consider the frequency resolution example of Figure 1. Let us assume that spectral analysis of the signal is carried out using the STFT SPA with a short analysis window. The resulting view would indicate a single frequency component with a rather wide frequency spread. The object model constructed solely on the basis of this view would be annotated with an SOU indicating that there may be insufficient frequency resolution at the spectrum level. Indeed, in the example of Figure 1, when the signal is processed using a larger STFT analysis window, two frequency components that are closely spaced are exposed.

Modeling effort, in the absence of other guidance such as discrepancies and/or generic models, would concentrate on the removal of SOUs in the evolving model. The SOUs would be used to direct search effort into unexplored regions. In addition, when new instances of a previously encountered object arrive, the initial model obtained could be used to suggest a starting point for the search effort and the SOUs used to guide further effort. This would enable better utilization of modeling resources.

By explicitly maintaining information regarding the uncertainty within a model, it is possible to reason about the models, predict possible model ambiguities and as described earlier, direct data reprocessing effort.

8.2 Stable Models

Our model search effort must not only seek out the intrinsic characteristics of the objects, but also generate robust models that do not require frequent revisions. By sampling the search space at distant points, a more complete view of the object is obtained. However, to acquire robust models, we must systematically explore the search space till the views obtained at neighboring points do not vary significantly or are *stable*. For instance, while selecting peaks from a spectrum based on an energy criterion, it is more likely that the energy threshold used is valid if on decreasing it by some factor, say two, no new peaks filter through. What constitutes a neighboring point, that is, what are acceptable step sizes for each parameter of an SPA? Signal processing domain theory and empirically gained knowledge will be used to determine these for use in the modeling effort.

8.3 Generic Models

For complex objects that exhibit many distinct high level features, the search effort involved may be significant. For example, a motor sound is characterized by harmonics which are tricky to identify without the use of a specialized SPA. Such SPAs are however not routinely used due to their associated cost. If the class of an object could be established using some preliminary analysis, the relevant specialized SPAs could be invoked to extract additional features. Such an approach would reduce search effort, and towards this, we define generic models.

What we seek to capture in a generic model is a unique feature or a set of features that frequently co-occur and are representative of a class of sounds. For example, motor sounds and ring/buzzer sounds. In fact, a hierarchy of such models may be defined. Leaf level models could include classes such as, tonal, complex tonal, impulsive, quasi-periodic impulsive, ringing and noise.

How do we represent these models? Given that we seek models to represent a class of sounds, a representation as specific (absolute ranges for frequency, energy and duration) as that in Figure 4 would be inappropriate. The model must capture only the intrinsic and not the incidental characteristics of the sound. To make this point clear: a motor sound should be recognized as such regardless of its intensity, the harmonics that may have been attenuated (a virtue of the physical casing of the motor), or the frequency of its fundamental (dependent on the power-line frequency). Likewise a "chirp", or linearly modulated sinusoid, is a chirp regardless of its duration or its shrillness. Our approach is to associate with each generic model a unique set of properties, whose existence is established through the use of specialized SPAs and/or specific SPA parameterizations. With a hierarchy of generic models, the properties of the leaf models would have to be satisfied in addition to properties that capture their inter-relationships.

To make this discussion more concrete, let us re-examine the physical nature of the sound originating from a hair dryer (refer Section 3 and Figure 4). The sound from a hair dryer is a result of the working of a motor and the forcing of air through a nozzle. A generic model, say **blower** defined in terms **motor** and **noise**, would cover both hair dryers and fans. The blower model would be defined as the time synchronized occurrence of motor and noise characteristics Motors generate a set of harmonically related time frequency trajectories (fundamental equal to the power-line frequency), which are synchronized in time. Analysis towards establishing the presence of a motor sound would be focused on determining the existence of harmonics using a specialized SPA for harmonic detection and enhancement. The detection of noise rests on finding broad regions of spectral activity with no apparent correlation. When indeed a blower is active, the noise and motor harmonics are present and synchronized in time.

It is true that generic models need to be defined, but, having a hierarchy enables the re-use of primitive models. We believe that the savings in search effort that will be realized justify their construction and the time spent to identify the applicable generic models during training. As a first step, we will experiment with models for ringing, motor (complex tonals) and impulsive sounds.

8.3.1 Indexing

The knowledge encapsulated in the generic models may be readily used if class information is provided along with the input signal. This, however, would be a step backward in the direction of automation. Instead, we favor using low level processing with default SPAs and parameters to drive the data up to the stream level of data abstraction. Then, we ascertain whether any of the generic models known to the system match the data. If yes, generic model hypotheses are created, annotated with SOUs to capture our belief or lack thereof as a result of missing pieces of support evidence. In particular, the initial data analysis is not targeted to detecting finer features that may involve the use of special purpose, more compute intensive analysis. The SOUs, based on their importance, form the seed for further processing of the signal. With a hierarchy of generic models, when leaf models are established as present, a recursive test for parent models is carried out.

Returning to our hair dryer example: let the default processing (high energy threshold with high frequency resolution for spectral analysis) of the acoustic signal result in the detection of three of its highest energy frequency components that are approximately harmonic. This would index into the generic motor sound, but such a hypothesis would be accompanied with uncertainty: no evidence for noise, harmonics not accurately detected, missing fundamental frequency, and time synchrony not clearly established. These are a direct result of the search space that was not explored: that is, the use of a high energy threshold that excludes the capture of weak frequency components, STFT analysis with only a long analysis window which would provide insufficient time resolution for the detection of rapid temporal variations and hence the presence or absence of time synchronization. Reprocessing to resolve/reduce the uncertainty in the motor hypothesis would entail the use of SPAs and parameters known to handle the above distortions. Signal reprocessing always reduces uncertainty in a model. When the expected characteristics are not found on signal reprocessing, our belief in the initial hypothesis: that the sound belonged to the generic model class being tested is reduced. Alternately, if the features are detected, the more complete model would have been acquired with considerably less search effort.

8.3.2 Expectations and Search

Generic models identified to be present provide expectations regarding the signal content and its future behavior. In the context of these expectations, the signal data is analyzed in a more focused manner. Continuing with the hair dryer example, this would translate to using the specialized SPA for the detection of harmonics (which in the course of blind search would not have been used due to its associated cost). In the SUT [33], Configuration II, the approximate knowledge available in the spectral bands is used to generate sound source hypotheses. These in turn generate expectations for specific support units which are tested for using specialized SPAs as necessary. In the model acquisition system, we will exploit generic models in a similar fashion. Towards this, the SUT blackboard database will be extended to include a space for generic models.

8.4 Model Revision

Model revision occurs either through model generalization or specialization. *Generalization* becomes necessary only when the model for a new instance of an object, *new-model* is not encompassed by the earlier acquired model, *old-model.* If the new-model subsumes the old-model and does not introduce conflicts in the database, it is used to replace the old-model after updating statistics used in belief calculation. When the old and new models are not related by the subsumes relation, generalizations that cover the old and new models are generated, *gen-models.* If there exists one such model that does not introduce conflicts into the database it is retained. If no such model exists and new-model is consistent with the database, a "disjunctive" model may be generated and incorporated in the database. We will adopt this approach before seeking to generalize object models to establish whether indeed the majority of the objects are similar and which are exceptions. Such a situation is likely to occur when the different instances semantically belong to the same class but are physically very distinct. In all other cases, the nature of conflict may be examined and the object model *specialized* through data reprocessing to extract additional or more refined characteristics. As indicated earlier in the example of Section 7, reprocessing effort directed at model specialization is more focused and has a better defined termination criterion when the training data are available for re-use.

8.5 Supervised versus Unsupervised Learning

Learning is considered supervised when the object label is provided along with the signal data. In the algorithm presented in Section 6, the object label serves four purposes, firstly, it associates with the object model constructed a label that is meaningful to the world. Secondly, when additional training instances of an object are encountered, the system compares the newly constructed object model to that previously acquired and seeks to build a model that is consistent; it seeks to generalize its description of the object. The degree of generality is however governed by the "supervisor" who labels the objects. The object models could become overly general and meaningless if very different instances of an object are labeled similarly. For example, consider grouping together all hair dryers sounds together (regardless of the physical structure of the sound producing appliance or the power-line frequency of operation) as opposed to only sounds originating from hair dryers of a single brand. One would expect the former to perform poorly for recognition purposes unless the modeling process could extract the intrinsic high level features (this will be explored to some extent in the course of this work). Thirdly, the purpose in providing labels is to trigger search in the SPA parameter space in order to capture views that disambiguate the signals and hence their models. Fourthly, if an instance of an object has been encountered and modeled earlier, expectations about its characteristics enable the selection of SPAs and their parameters. Labels provide more informed expectations about the signal being processed (akin to the use of class labels for generic models) and thus reduce search effort.

In the absence of class information, the modeling process would mainly be guided by the stability criterion and the use of generic models. When a newly generated model closely resembles an earlier acquired model - based on similarity metrics used in the recognition engine, the models could be combined to yield a more general model. In principal, the learning algorithm would perform satisfactorily without supervision.

9 Evaluation

For evaluation purposes in the domain of speech recognition, a rich variety of commercial and research systems are available; recognition performance and training effort are comparable. For the acoustic, non-speech signal domain, no such systems are readily available for comparison purposes. Consequently, our evaluation effort of the model acquisition system will be with respect to itself, studying the effects of termination criteria, training data re-usability, and generic models on modeling effort and recognition performance.

In particular the following comparisons will be made:

- Comparison of hand-crafted and learned models with regard to SUT recognition performance.

- The training effort entailed when the object class label is provided will be compared to that resulting when it is not. In addition, the actual models acquired will be compared, to study the degree of generalization achieved through the use of object class labels as opposed to using the similarity metric in the recognition engine. It is expected that the latter will give rise to a lower level of generalization when the recognition engine is more conservative.

- Modeling effort as a function of the search termination criterion will be studied. We seek to answer whether requiring stable models as opposed to just unambiguous models results in reduced modeling effort over a training sequence?

- Study the effect of different search heuristics on model acquisition efficiency.

- Study database consistency as a function of the number of training instances that will be maintained for re-use.

10 Status

Currently the system is capable of modeling synthetic sounds that exhibit steady frequency behavior such as those presented in the example of Section 7. Our immediate next step is to handle synthetic acoustic events that exhibit transient behavior: chirps and impulsive sounds. We will then proceed with the learning of real world sounds using generic models.

11 Conclusion

To address recognition for complex tasks that are characterized by a high degree of variability and non-stationarity, recognition systems that adapt their signal processing front-ends have merged. To meet the parallel needs of interpretation and object disambiguation, such systems employ symbolic object models that typically have been hand-crafted. We propose a supervised knowledge-based learning approach to automate object model acquisition for such systems that relies on the detection of signal processing discrepancies and their resolution. Success, determined by the acquisition of models that meet our recognition needs, would demonstrate one solution to the knowledge acquisition bottleneck of such systems.

Acknowledgements

This work was supported by the Rome Air Development Center of the Air Force Systems Command under contract F30602-91-C-0038. The content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

We would like to thank Paul Utgoff for his several helpful comments. The first author also thanks Carla Brodley for her comments on initial drafts, and Bruce Draper for being available to discuss the vision literature.

References

[1] D. Adams and Y. Pao, "High Order Neural Networks for Sonar Signal Discrimination" *DARPA Artificial Neural Network Technology, 1991 Program Review*, Vol II Proceedings, pp. 409-422

[2] T. Ash, "Dynamic Node Creation in Backpropagation Networks", (ICS Report 8901), San Diego, CA., Institute of Cognitive Science, University of California 1989.

[3] A. Averbach, L. Bahl, R. Bakis, P. Brown, A. Cole, G. Dagett, S. Das, K. Davies, S. de Gennaro, P. de Souza, E. Epstein, D. Fraleigh, F. Jelinek, S. Katz, B. Lewis, R. Mercer, A. Nadas, D. Nahamoo, M. Picheny, G. Schichman, P. Spinelli, "An IBM-PC Based Large Vocabulary Isolated Utterance Speech Recognizer", *Proc. IEEE ICASSP*, pp. 53-56, Tokyo, Japan, April 1986.

[4] B. Bhanu, S. Lee, and J. Ming, "Self-Optimizing Control System for Adaptive Image Segmentation", *Proc. of the DARPA Image Understanding Workshop*, Pittsburgh, PA., Sept. 1990. pp. 583-595.

[5] N. Bitar, E.Dorken, D. Paneras and H. Nawab, "Integration of STFT and Wigner Analysis in a Knowledge-Based Sound Understanding System", *IEEE ICASSP '92 Proceedings*, March 1992.

[6] A. Bregman, "Auditory Scene Analysis: The Perceptual Organization of Sound", MIT Press, 1990.

[7] O. Camps, L. Shapiro, and R. Haralick, "PREMIO: The Use of Prediction in CAD Model-Based Vision System", Technical Report no. EE-ISL-89-01, University of Washington, Seattle, July, 1989.

[8] N. Carver, *Sophisticated Control for Interpretation: Planning to Resolve Uncertainty*, Ph.D. Thesis, Dept. of Computer Science, Univ. of Massachusetts, 1990.

[9] C. Chen, P. Mulgoankar,"Automatic Vision Programming," *CVGIP: Image Understanding*, Vol. 55, No. 2, March, pp. 170-183, 1992.

[10] T.Claasen and W. Meclenbrauker, "The Wigner Distribution: A tool for Time-Frequency Signal Analysis", *Phillips J. Res.*, vol. 35, pp. 276-350, 1980.

[11] L. Cohen, "What is a Multicomponent Signal?", *IEEE*, 1992.

[12] M. Cohen, "Hybrid Neural Network/Hidden Markov Model Speech Recognition", *DARPA Artificial Neural Network Technology, 1991 Program Review*, Vol II Proceedings, pp. 254-273.

[13] B. Dawant and B. Jansen, "Coupling Numerical and Symbolic Methods for Signal Interpretation", IEEE transactions on Systems, Man Cybernetics, Vol. 21, No. 1, Jan/Feb 1991.

[14] G.R. Doddington, "Phonetically Sensitive Discriminants for Improved Speech Recognition", *Proceedings IEEE ICASSP*, pp. 556-559, Glasgow, Scotland, May 1989.

[15] E. Dorken, H. Nawab, and V. Lesser, "Extended Model Variety Analysis for Integrated Processing and Understanding of Signals", *IEEE ICASSP '92 Proceedings,* March 1992.

[16] E. Dorken, "Approximate Processing and Knowledge-Based Reasoning of Non-Stationary Signals", PhD Thesis, Dept. of Electrical Engineering, Boston University, 1993.

[17] B. Draper, "Learning Object Recognition Strategies", PhD Thesis, Dept. of Computer Science, University of Massachusetts, CMPSCI TR93-50, 1993.

[18] L. Erman, R. Hayes-Roth, V. Lesser, D. Reddy, "The Hearsay II Speech Understanding System: Integrating Knowledge to Resolve Uncertainty", Computing Surveys, Vol. 12, June 1980.

[19] U. Fayyad, Personal communication, Amherst, MA, June 1993.

[20] K. Fukunaga, "Introduction to Statistical Pattern Recognition", Academic Press, London, 1990.

[21] D. Gabor, "Theory of Communication", *Journal of the Institute of Electrical Engineers*, **93**, pp. 429-441, 1946.

[22] S.I. Gallant, "Connectionist Expert Systems", *CACM*, Vol .31, No. 2, February 1988.

[23] R. Gray, "Vector Quantization' *IEEE ASSP Magazine*, Vol. 1, No. 2, pp. 4-29, April 1984.

[24] A.R. Hanson and E.M. Riseman, "VISIONS: A Computer System for Interpreting Scenes", in *Computer Vision Systems*, Hanson and Riseman (eds.), N.Y.: Academic Press, 1978. pp 303-333.

[25] B. Hayes-Roth, R. Washington, R. Hewett, and A Seiver, "Intelligent Monitoring and Control", *IJCAI '89 Proceedings* pp. 243-249.

[26] S. Haykin and T.K. Bhattacharya, "Adaptive Radar Detection using Supervised Learning in Time-Frequency Domain", in ??.

[27] Ikeuchi, K. and Hong, K.S. "Determining Linear Shape Change: Toward Automatic Generation of Object Recognition Programs," CGVIP: Image Understanding, 53(2): 154-170 (1991).

[28] E. Karnin, " A Simple Procedure for Pruning back-propagation Trained Neural Networks", *IEEE Trans. on Neural Networks*, 1, pp. 239-242, 1990

[29] F. Klassner, "Data Reprocessing and Assumption Representation in Signal Understanding Systems", Technical Report 92-52, Computer Science Dept, University of Massachusetts, 1992.

[30] C. Kohl, A. Hanson, E. Riseman, "A Goal-Directed Intermediate Level Executive for Image Interpretation", *Proc. 1987 of the Joint International Conference on Artificial Intelligence.*

[31] K. Lee, "Large-Vocabulary Speaker Independent Continuous Speech Recognition: The SPINX System", PhD Thesis, Dept. of Computer Science, Carnegie Mellon University, 1988, Technical Report No. CMU-CS-88-148.

[32] V. Lesser, H. Nawab, M. Bhandaru, Z. Cvetanovic, E. Dorken, I. Gallastegi, and F. Klassner, *Integrated Signal Processing and Signal Understanding*, Technical Report 91-34, Computer Science Dept., University of Massachusetts, 1991. Also available as: H. Nawab and V. Lesser, "Integrated Processing and Understanding of Signals", Chapter 6, *Knowledge-Based Signal Processing*, A. Oppenheim and H. Nawab, editors, Prentice Hall, New Jersey, 1991.

[33] V. Lesser, H. Nawab, I. Gallastegi, F. Klassner, "IPUS: An Architecture for Integrated Signal Processing and Signal Interpretation in Complex Environments", *AAAI-93*, Washington, USA, 1993. A more detailed version will be appearing in *AIJ* in 1995.

[34] R.Lippmann, E. Martin, D. Paul, "Multi-style Training for Robust Isolated-word Speech Recognition", *Proceedings of IEEE ICASSP*, pp. 705-708, Dallas, TX, April 1987.

[35] R. Lippmann, "Review of Neural Networks for Speech Recognition", *Readings in Speech Recognition*, Edited by: Alex Waibel and Kai-Fu Lee, Morgan Kaufmann Publishers, Inc. San Mateo, CA, 1990.

[36] S. Luse, H. McBeth, "Passive Sonar Detection/Classification", *DARPA Artificial Neural Network Technology, 1991 Program Review*, Vol II Proceedings, pp. 450-479.

[37] J. Ming, B. Bhanu, "A Multistrategy Learning Approach for Target Model Recognition, Acquisition, and Refinement", *Proc. of the DARPA Image Understanding Workshop*, Pittsburgh, PA., Sept. 1990. pp. 742-756.

[38] R. De Mori, L. Lam and M. Gilloux, "Learning and Plan Refinement in a Knowledge-Based System for Automatic Speech Recognition", *IEEE* 1987, pp. 246-262.

[39] H. Murase and S. Nayar, "Learning Object Models from Appearance", *AAA93*, pp. 836-843, Washington, July 1993.

[40] H. Nawab and T. Quatieri, "Short-Time Fourier Transform", *Advanced Topics in Signal Processing*, Prentice Hall, New Jersey, 1988.

[41] J. Naylor, S. Boll, "Techniques for Suppression of Interfering Talker in Co-channel Speech", *Proceedings of IEEE ICASSP*, pp. 205-208, Dallas, TX, April 1987.

[42] H. Nii, E. Feigenbaum, J. Anton, A. Rockmore, "Signal-to-Symbol Transformation: HASP/SIAP Case Study", *AI Magazine.* Vol. **3** (2), pp. 23-35, Spring 1982.

[43] T.W. Parson's, "Separation of Speech from Interfering Speech by Means of Harmonic Selection", *Journal of Acoustic Society of America*, vol 60, no. 4, pp. 911-918, Oct. 1976.

[44] Y. Peng and J. Reggia, "Plausibility of Diagnostic Hypotheses: The Nature of Simplicity", *AAAI 86 Proceedings*, pp 140-145.

[45] J. Picone, "Continuous Speech Recognition Using Hidden Markov Models", *IEEE ASSP Magazine*, pp. 26-41, July 1990.

[46] J.R. Quinlan, "Induction of Decision Trees", *Machine Learning* Vol.**1**, No.1, pp. 81-106, 1986.

[47] L.R. Rabiner, J.G. Wilpon, F.K. Soong, "High Performance Digit Recognition Using Hidden Markov Models", *Proceedings IEEE ICASSP*, pp. 119-122, New York, April 1988.

[48] L. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", *The IEEE* , Vol. 77, No. 2, pp. 257-285, February 1989.

[49] D.E. Rumelhart, J.L.McClelland and the PDP Research Group, *Parallel Distributed Processing, Explorations in the Microstructure of Cognition, Vol. I: Foundations*, A Bradford Book, MIT Press, 1986.

[50] R. W. Schafer, L.R. Rabiner, "Digital Representation of Speech Signals", *Readings in Speech Recognition*, Edited by: Alex Waibel and Kai-Fu Lee, Morgan Kaufman Publishers, Inc. San Mateo, California, 1990.

[51] C. Vadala, "Gathering and Evaluating Evidence for Sound Producing Events", Masters Thesis, Dept of Biomedical Engineering, Boston University, January 1992.

[52] R.L. Watrous, L. Shastri, A.H. Waibel, "Learned Phonetic Discrimination Using Connectionist Networks", *European Conference on Speech Technology*, pp 377-380, Edinburgh 1987.

[53] M. Weintraub, "A Computational Model for Separating Two Simultaneous Talkers", *IEEE ICASSP 86, Tokyo*, pp. 81-84, 1986.