# DISTINCTIVE FEATURE DETECTION USING SUPPORT VECTOR MACHINES

*Partha Niyogi, Chris Burges, and Padma Ramesh*

Bell Labs, Lucent Technologies, USA.

## ABSTRACT

An important aspect of distinctive feature based approaches to automatic speech recognition is the formulation of a framework for robust detection of these features. We discuss the application of the support vector machines (SVM) that arise when the structural risk minimization principle is applied to such feature detection problems. In particular, we describe the problem of detecting stop consonants in continuous speech and discuss an SVM framework for detecting these sounds. In this paper we use both linear and nonlinear SVMs for stop detection and present experimental results to show that they perform better than a cepstral features based hidden Markov model (HMM) system, on the same task.

## 1. INTRODUCTION

We are pursuing an approach to speech recognition that develops detectors for various distinctive features from their acoustic correlates in the speech signal. Crucial to the success of such an approach are the following: (1) determination of the acoustic signatures for different sound classes and the development of signal representations in which that acoustic signature best expresses itself; (2) the construction of statistical learning paradigms that operate on the above representations and separate the positive instances of the distinctive feature from the negative instances of the same feature. Traditionally, (1) is regarded as the front-end and (2) as the back end of a speech recognition system. In most traditional speech recognition designs, the *same* representation is used for all sound classes (i.e., cepstra; filterbanks; computed with the same analysis window and stepping rate). The front-end is thus a vector time series. The back-end is typically an HMM of one form or another. In contrast, we consider using different representations for different sound classes. An important question that arises in this context is: given a particular representation, what sort of a statistical learning machine should be used to optimally separate the positive examples of each sound from the negative?

In this paper, we investigate the application of support vector machines (SVM) for the kinds of detection problems that would naturally arise in our feature based approach to speech recognition. Here, we address the issue of detecting stop consonants in continuous speech. We first provide a description of the detection problem and then discuss support vector machines.

An important issue that needs to be addressed is the ability of the machine to generalize from its training set to its test set. We use the Vapnik-Chervonenkis theory [7] that gives rise to SVMs to address this issue. We examine a variety of support vector machines with different architectures and capacity control mechanisms and show their effect on the successful utilization of SVMs for speech recognition. Finally, we present experimental results on using SVMs for the detection of stop consonants.

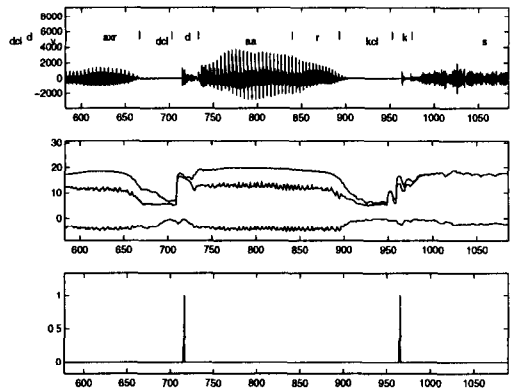

Figure 1: Portion of the speech waveform $s(n)$,(top panel), the associated three-dimensional feature vector, $\mathbf{x}(n)$ (middle panel), and the desired output $y(n)$ bottom panel marking the times of the closure-burst transition, x axis is time in msec.

## 2. STOP DETECTION

Stop consonants are produced by causing a complete closure of the vocal tract followed by a sudden release. Hence they are signalled in continuous speech by a period of extremely low energy (corresponding to the period of closure) followed by a sharp, broad band signal (corresponding to the release). As a result, stops consonants are highly transient (dynamic) sounds that have a varying duration lasting anywhere from 5 to 100 ms. In American English, the class of stops consists of the sounds { p,t,k,b,d,g}.

In order to build a detector for stop consonants in running speech, the speech signal, $s(t)$, is characterized by a vector time series with three dimensions: (i) log(total Energy) (ii) log(Energy above 3kHz) (iii) spectral flatness measure based on Wiener Entropy defined as $\int log(S(f,t))df - log(\int S(f,t)df)$ where $S(f,t)$ is spectral energy at frequency $f$ and time $t$. All quantities are computed using 5 ms windows moved every 1 ms. Thus, we have $\mathbf{x}(n) = [x_1(n)\ x_2(n)\ x_3(n)]'$ where $n$ represents time (discretized in units of milliseconds) and $x_1$ through $x_3$ are the three acoustic quantities that are measured every 1 ms. Energies at 1 ms intervals potentially allow us to track rapid transitions that would otherwise be smoothed out by a coarser temporal resolution. For more on stop consonants, their acoustic-phonetic features and common confusions, see [4].

We need to find an operator on the feature vector time series that will return a single dimensional time series that takes on large values around the times that stops occur and small values otherwise. The most natural points in time that mark the presence of stops are the transition from closure to burst release. Shown in fig. 1 is an example of a speech waveform $s(n)$, the associated feature vector

time series $\mathbf{x}(n)$ and a desired output $y(n)$.

The technical goal is to find an operator $g$ on the time series $\mathbf{x}(n)$ that produces an output $y_g(n) = g \circ \mathbf{x}(n)$ with values in {0,1}, such that $\| y - y_g \|$ is small in some sense (norm). Specifically, we choose the optimal operator (from some class $\mathcal{G}$ of operators) according to the criterion

$$g_{opt} = \arg\min_{g \in \mathcal{G}} R(g) = \arg\min_{g \in \mathcal{G}} E[(y - y_g)^2] \qquad (1)$$

Since both $y$ and $y_g$ have values in $\{0, 1\}$, this is a two class, pattern classification problem at each point in time $n$. Here we consider operators given by $y_h(n) = h(\mathbf{x}(n-W), \ldots, \mathbf{x}(n+W))$ where $h$ is a function from $R^{3(2W+1)} \longrightarrow \{0, 1\}$. In our experiments $W = 5$.

Finally, it is important to emphasize that the speech recognition problem can be decomposed into a collection of feature detection problems that have a structure very similar to that of the stop detection problem described above. For example, a vowel detector might be built with a representation $\mathbf{x}$ consisting of dimensions like degree of periodicity in the signal, ratio of high frequency to low frequency energy, and so on. The same is true of a nasal detector, a fricative detector and so on. In all of these detection problems, the support vector machine framework might provide the basis for constructing optimal detectors that are learned from the data.

## 3. SUPPORT VECTOR MACHINES

For a general introduction to using SVMs for the pattern recognition task, see [1]. Consider a two-class pattern recognition problem with labelled examples, i.e., $(\mathbf{x}_i, y_i)$ pairs drawn according to some unknown distribution $P(\mathbf{x}, y)$ on the space $X \times Y$. The goal is to construct a function $h$ (drawn from some class $\mathcal{H}$ : $X \longrightarrow Y$ ; let $Y = \{-1, 1\}$ without loss of generality) that is able to classify unknown patterns $\mathbf{x}$ into the appropriate class with minimum misclassification error. A suitable $h$ is usually picked by minimizing the empirical risk, i.e.,

$$\hat{h}_l = \arg\min_{f \in \mathcal{H}} R_{emp}(f) = \arg\min_{f \in \mathcal{H}} \frac{1}{l}(y_i - f(\mathbf{x}_i))^2$$

This is the straightforward approach commonly pursued in statistical pattern recognition and the hope is that the empirically chosen function $\hat{h}_l$ would generalize successfully to novel unlabelled examples.

### 3.1. Structural Risk Minimization

The straightforward approach of minimizing the empirical risk turns out not to guarantee a small actual risk on the test set, particularly, if the number $l$ of training examples are limited. As a result of the work of Vapnik and Chervonenkis [7] (see also [5, 6]), a new induction principle has emerged, the principle of structural risk minimization (SRM). This is based on the fact that the true goal of the learner should be to minimize the *expected risk*, i.e.,

$$h_{opt} = \arg\min_{f \in \mathcal{H}} R(f) = \arg\min_{f \in \mathcal{H}} E[(y - f(\mathbf{x}))^2]$$

where the expectation is taken according to the true distribution $P$. Since $P$ is unknown, one approximates the above functional $R(f)$ by the following large deviation bound (that holds with probability greater than $1 - \eta$)

$$R(f) \leq R_{emp}(f) + \Phi(\frac{d}{l}, \frac{\log(\eta)}{l}) \qquad (2)$$

where $\Phi$ is defined as $\Phi(\frac{d}{l}, \frac{\log(\eta)}{l}) = \sqrt{\frac{d\left(\log\frac{2l}{d}+1\right) - \log(\eta/4)}{l}}$ The parameter $d$ is called the VC–dimension of the set of functions, $\mathcal{H}$[7] and is a measure of its complexity. Some aspects of eq. 2 are worth highlighting. First, to guarantee minimization of the true risk, $R(f)$, one has to ensure that both terms, $R_{emp}(f)$ and $\Phi(\frac{d}{l}, \frac{\log(\eta)}{l})$ are made sufficiently small — significantly, it is noted that minimizing $R_{emp}$ alone is not enough. Second, for a fixed amount of training data $l$, the two components represent a complexity regularization trade-off. As the class $\mathcal{H}$ becomes larger, the minimum empirical risk becomes smaller but the VC term becomes larger. Generalization is controlled by controlling each of these two terms.

Conceptually this is done by imposing a structure $\mathcal{H}_1 \subset \mathcal{H}_2 \subset \mathcal{H}_3 \ldots \mathcal{H}$ of nested subsets of $\mathcal{H}$. One then searches within this hierarchy for the class $\mathcal{H}_j$ that minimizes the bound, i.e.,

$$\hat{h} = \arg\min_{\mathcal{H}_j \in \mathcal{H}} \left(R_{emp}(f) + \Phi()\right) \qquad (3)$$

### 3.2. Hyperplanes

Here we discuss the application of the SRM principle to the case where the class $\mathcal{H}$ is the class of linear hyperplanes in $X$, i.e., $\mathcal{H} = \{h : X \rightarrow \{-1, 1\} | h(\mathbf{x}) = \theta(\mathbf{w}.\mathbf{x} + b)\}$ where $\theta(z) = 1$ if $z \geq 0$, and $\theta(z) = -1$ otherwise. Thus each pair $(\mathbf{w}, b)$ corresponds to a unique hyperplane (after removing an arbitrary scale factor in $\mathbf{w}$ and $b$ by requiring that $\min_i |\mathbf{w}.\mathbf{x}_i + b| = 1$ where $(\mathbf{x}_i, y_i)$ are the set of training examples.) Now we use the following theorem [7]:

**Theorem 1** *Let $B_{\mathbf{x}_1,\ldots,\mathbf{x}_r} = \{\mathbf{x} \in \mathbf{X} : \|\mathbf{x} - \mathbf{a}\| < V\}$ $(\mathbf{a} \in \mathbf{X})$ be the smallest ball containing the points $\mathbf{x}_1, \ldots, \mathbf{x}_r$, and*

$$\mathcal{H}_A = \{f_{\mathbf{w},b} = \theta\left((\mathbf{w}.\mathbf{x}) + b\right) \mid \|\mathbf{w}\| \leq A\} \qquad (4)$$

*be a subclass of hyperplanes in canonical form with respect to $\{\mathbf{x}_1, \ldots, \mathbf{x}_r\}$. Then, $\mathcal{H}_A$ has a VC-dimension $d$ satisfying*

$$d \leq V^2 A^2. \qquad (5)$$

This theorem suggests a natural structure on the set of hyperplanes in canonical form. Clearly, $\mathcal{H} = \cup_{A>0} \mathcal{H}_A$. It is possible to show that utilizing such a structure transforms the problem posed in eq. 3 to

$$\hat{h} = \arg\min_{\mathbf{w},b} \left(R_{emp}(\mathbf{w}, b) + \lambda \mathbf{w}.\mathbf{w}\right)$$

$\lambda$ trades-off the fit to data with model complexity.

#### 3.2.1. *Separable Data*

For the case, where the data is separable by hyperplanes, the structural risk minimization principle attempts to minimize the VC-dimension of $\mathcal{H}_A$ while keeping the empirical risk $R_{emp}$ fixed at 0. This is equivalent to

$$\min \frac{1}{2}\mathbf{w}.\mathbf{w} \qquad (6)$$

subject to

$$y_i(\mathbf{w}.\mathbf{x}_i + b) \geq 1 \ \ \forall i \qquad (7)$$

The $i$th constraint is satisfied only if the $i$th data point is correctly classified by the hyperplane classifier. Introducing Lagarange multipliers for each of the constraints, the Lagrangian is formed as

$$\mathcal{L}(\mathbf{w}, b, \{\alpha_i\}) = \frac{1}{2}\mathbf{w}.\mathbf{w} - \sum_i \alpha_i(y_i(\mathbf{w}.\mathbf{x}_i + b) - 1)$$

The optimal solution lies at the saddle point of the Lagrangian and satisfies the following conditions [1]: (1) differentiating with respect to $\mathbf{w}$ and setting to 0 yields $\mathbf{w}^* = \sum_i y_i\alpha_i\mathbf{x}_i$, i.e., the optimal hyperplane is a linear combination of the training vectors, (2) differentiating with respect to $b$ and setting to 0 yields $\sum_i \alpha_i y_i = 0$, (3) first order Kuhn Tucker conditions yield $\alpha_i(y_i(\mathbf{w}.\mathbf{x}_i + b) - 1) = 0$. From this we see that all points $\mathbf{x}_i$ for which $\alpha_i$ is non-zero lie on the *margin hyperplanes* $\mathbf{w}^*.\mathbf{x} + b = 1$ or $\mathbf{w}^*.\mathbf{x} + b = -1$. Such points are called *support vectors*. All other points are exterior points and do not enter in the expansion of the optimal hyperplane $\mathbf{w}^*$ since they have $\alpha_i = 0$. As a result of these properties, the learning machine is called the *support vector machine*.

Substituting for $\mathbf{w}$ in the Lagrangian yields the quadratic programming problem, max $\sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i y_i \alpha_j y_j (\mathbf{x}_i.\mathbf{x}_j)$ subject to $\sum_i \alpha_i y_i = 0$.

### 3.2.2. *Non-separable Case*

In many practical applications, a perfectly separating hyperplane does not exist. To allow for the possibility of examples violating (7), [2] introduce slack variables

$$\xi_i \geq 0, \quad i = 1, \ldots, l, \tag{8}$$

to get

$$y_i((\mathbf{w}.\mathbf{x}_i) + b) \geq 1 - \xi_i, \quad i = 1, \ldots, l. \tag{9}$$

The structural risk minimization approach to minimizing the guaranteed risk bound (2) consists of the following:

$$\min \Phi(\mathbf{w}, \xi) = \mathbf{w}.\mathbf{w} + U\sum_{i=1}^{l}(\xi_i)^\epsilon \tag{10}$$

subject to the constraints (8) and (9).

In eq. 10, the $\mathbf{w}.\mathbf{w}$ corresponds to the VC-dimension of the learning machine as before. The term $\sum_{i=1}^{l}(\xi_i)^\epsilon$ (for small $\epsilon$) is equivalent to the number of misclassifications on the training set and therefore a measure of empirical risk. The constant $U$ therefore controls the trade-off between the empirical risk $\sum_{i=1}^{l}(\xi_i)^\epsilon$ and the VC-dimension of the learning machine. In actual practice, one has to make choices both for $U$ and $\epsilon$. For computational reasons, $\epsilon$ is chosen to be 1 because that translates the optimization problem of eq. 10 into a quadratic programming problem like that of the previous section (in fact, $\epsilon = 2$ does also). Introducing Lagrange multipliers for each of the constraints in eqs. 8 ($\lambda_i$'s) and 9 ($\alpha_i$'s), we form the Lagrangian $\mathcal{L}(\mathbf{w}, \{\xi_i\}, b, \{\alpha_i\}, \{\lambda_j\})$ as before

$$\mathcal{L} = \frac{1}{2}\mathbf{w}.\mathbf{w} + U\sum_{i=1}^{l}(\xi_i) - \sum_{j=1}^{l}\lambda_j\xi_j - \sum_{i=1}^{l}\alpha_i(y_i((\mathbf{w}.\mathbf{x}_i) + b) - 1)$$

First order conditions show that the optimal hyperplane is given by $\mathbf{w}^* = \sum_i y_i\alpha_i\mathbf{x}_i$. Additionally taking into account the Kuhn-Tucker conditions, eq. 10 is transformed into
min $\sum_i \alpha_i - \frac{1}{2}\sum_i\sum_j \alpha_i\alpha_j y_i y_j (\mathbf{x}_i.\mathbf{x}_j)$ subject to $\sum_i \alpha_i y_i = 0; 0 \leq \alpha_i \leq U$. Once the $\alpha_i$'s are obtained by solving the above problem, the optimal hyperplane is easily found by substitution.

### 3.3. Non-linear Extensions: Kernels

In order to get non-linear decision boundaries, we transform the data by a fixed non-linear transformation and use linear techniques in the transformed space. Consider a transformation $\psi : \mathbf{X} \longrightarrow \mathbf{Z}$ mapping points $\mathbf{x}_i \in \mathbf{X}$ to corresponding $\mathbf{z}_i \in \mathbf{Z}$. Constructing hyperplanes in $\mathbf{Z}$ according to the SRM principle ultimately reduces to solving optimization problems of the sort described earlier (eq. 10) with $\mathbf{x}_i$'s replaced by $\mathbf{z}_i$'s. Significantly, we note that the only form in which $\mathbf{z}_i$'s appear in the optimization problem is inner products, i.e., $(\mathbf{z}_i, \mathbf{z}_j)$. Therefore, it is enough to know the inner product between pairs of them. Consequently, we characterize the transformation $\psi$ by the inner product it imposes on the space $\mathbf{Z}$. Specifically, we consider mappings of the form $\psi_K : \mathbf{X} \longrightarrow \mathbf{Z}$ such that for any $\mathbf{z}_1, \mathbf{z}_2 \in \mathbf{Z}$, we have $(\mathbf{z}_1, \mathbf{z}_2) = K(\mathbf{x}_1, \mathbf{x}_2)$ where $K$ is the Kernel of a positive Hilbert-Schmidt operator and according to Mercer's theorem of functional analysis ([3]) corresponds to a dot product in some other space. Each different choice of the kernel $K$ defines a different choice of the transformation $\psi_K$. The dimensionality of $\mathbf{Z}$ depends upon the number of non-zero eigenvalues of the kernel $K$ and is potentially infinite.

If we set up the optimization problem appropriately, we see that the optimal hyperplane has the form $\mathbf{w}^* = \sum_i y_i\alpha_i^*\mathbf{z}_i$. The decision rule for this is: $h(\mathbf{x}) = \theta(\sum_i y_i\alpha_i^* K(\mathbf{x}, \mathbf{x}_i) + b)$. We see that the form of this decision rule is like a feed-forward neural network, or a kernel-based non-parametric scheme with two significant differences (a) the parameters are estimated by the principle of structural risk minimization (b) the number of "hidden nodes" or "basis functions" is chosen automatically by the procedure. Thus an important problem of model selection is resolved.

## 4. EXPERIMENTAL RESULTS

As formulated in section 2, stop detection is a two class pattern classification problem that can be solved using SVMs. We discuss in this section the experiments conducted with these SVMs. Detection experiments were conducted on dialect region 4 of the TIMIT test set that consists of 32 speakers, 16 male and 16 female uttering 10 sentences each making for a total of 320 sentences in all. Training was performed on 10 sentences each from 4 randomly chosen speakers from the TIMIT training set from different dialect regions yielding 133 positive examples and 10760 negative examples.

### 4.1. Linear Hyperplanes

In this section we consider results obtained when the class of linear hyperplanes is used as a decision boundary between stops and non-stops (non-separable formulation).

On 10893 training data points, with $U = 1$, 163 support vectors were generated (1.5% of training vectors). The errors on the training set consisted of 25 false positives and 37 false negatives. Shown in fig. 2 is a distribution (normalized histograms) of $d(\mathbf{x}) = \mathbf{w}.\mathbf{x} + b$ for the positive and negative training examples. As $\frac{d(\mathbf{x})}{||\mathbf{w}||}$ is the distance of each datapoint $\mathbf{x}$ to the separating hyperplane $d(\mathbf{x})$ is proportional to this distance. Positive examples that have $d < 0$ and negative examples that have $d > 0$ are misclassified. The region $-1 \leq d \leq 1$ corresponds to the *margin*, i.e., points that lie within the strip given by the hyperplanes $\mathbf{w}.\mathbf{x} + b = 1$ and $\mathbf{w}.\mathbf{x} + b = -1$.

As we discussed in section 2, the problem is really one of accurate *detection* of stops. Consequently, by changing the threshold

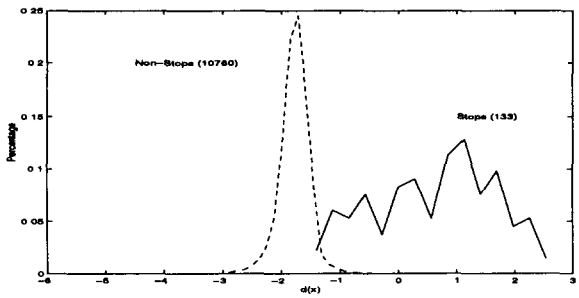


Figure 2: Histogram of $d(x)$ on the training set.

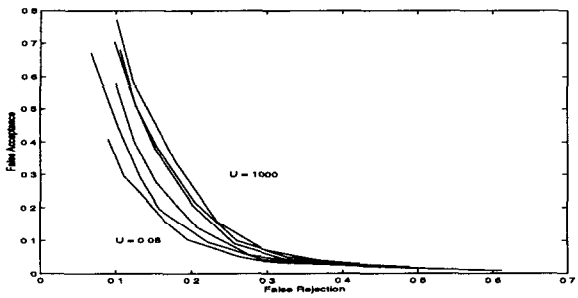


Figure 3: Performance of linear support vector machines.

(currently set at $d = 0$) one can obtain a trade-off between type I and type II errors for the stop detection problem. Shown in fig. 3 are the ROC curves generated by varying such a threshold of acceptance for values of $U = 1000, 100, 10, 1, 0.2, 0.05$. $U$ controls the trade-off between empirical fit to the data and capacity of the learning machine. Performance on the test set (a measure of generalization) gets progressively better as $U$ decreases over this range. The best performance was obtained for $U = 0.05$.

### 4.2. RBF Kernels

Another important choice in the adoption of the support vector framework for problems such as these involves the choice of kernel, $K$. In the experiments below, we considered Radial Basis Function kernels of the sort $K(\mathbf{x}, \mathbf{y}) = \exp - \frac{||\mathbf{x}-\mathbf{y}||^2}{\sigma^2}$. Experiments were conducted with values of $\sigma = 100, 250, 500$. Performance improved marginally from $sigma = 100$ to $\sigma = 500$. Shown in fig. 4 is the ROC curve for an RBF network with $\sigma = 500$ (labeled SVM). Also shown are ROC curves for the best linear hyperplane and one obtained using a derivative operator on total and high-frequency energies.

As a point of comparison, the performance of an HMM based system running with free grammar on the test sentences is shown by the '*' in fig. 4. The HMM based system consisted of 47 phones with 3 state left to right HMMs per phone. The output probabilites were mixtures of Gaussians (16 mixtures per state) and the front-end was a 39 dimensional vector time series obtained from the first 12 cepstral coefficients and total energy and their first and second differences (delta and delta-delta). A stop in a test sentence was considered to be correctly identified if the closure-burst transition was *anywhere* within the segment postulated as a stop by the HMM recognizer. If a particular segment postulated by the HMM recognizer as a stop did not contain a closure-burst

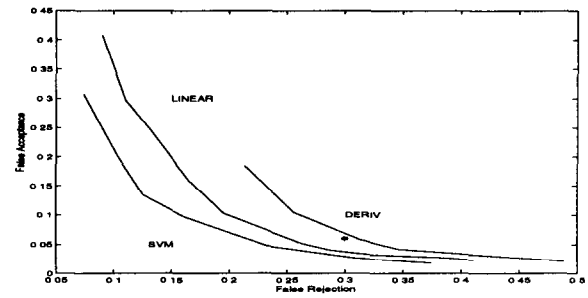


Figure 4: Performance of linear and non-linear support vector machines against derivative operators and HMMs.

transition, it was deemed a false accept.

## 5. CONCLUSIONS

Stop detection belongs to a class of feature detection problems that naturally arise in a feature based approach to speech recognition. It is particularly interesting as stops present a transient signal with a period of rapid change that is often poorly characterized by standard cepstral representations. We have discussed how the problem can be cast as trying to discriminate between positive and negative examples using functions drawn from the class $\mathcal{H}$ of the appropriate complexity. We have introduced the principle of structural risk minimization that provides a framework for doing this.

We have discussed the two major issues, the choice of an appropriate $U$ and kernel $K$ that affect the successful deployment of this technology for detection problems. We have also shown a steady improvement from linear to non-linear SVMs and shown that they perform better than an HMM using cepstral features.

## 6. REFERENCES

[1] Burges, C.J., "A Tutorial on Support Vector Machines for Pattern Recognition," Data Mining and Knowledge Discovery, Vol. 2, No. 2, pp. 1-47, 1998.

[2] Cortes, C. and Vapnik, V. N., "Support Vector Networks," Machine Learning Journal, Vol. 20, pp. 1-25.

[3] Courant, R. and Hilbert, D., *Methods of Mathematical Physics*, J. Wiley, New York, 1953.

[4] Niyogi, P., Mitra, P., and Sondhi, M., "A Detection Framework for Locating Phonetic Events," Proceedings of ICSLP-98, Sydney, Australia.

[5] Shawe-Taylor, J., Bartlett, P. L., Williamson, R. C., and Anthony, M., "A Framework for Structural Risk Minimization," Proceedings, 9th Annual Conference on Computational Learning Theory, pp. 68-76, 1996.

[6] Shawe-Taylor, J., Bartlett, P. L., Williamson, R. C., and Anthony, M., "Structural Risk Minimization over Data-Dependent Hierarchies," NeuroCOLT Technical Report NC-TR-96-053, 1996.

[7] Vapnik, V. N. *The Nature of Statistical Learning Theory*, Springer-Verlag, 1995.