A new class of Support Vector Machine (SVM) that is applicable to sequential-pattern recognition such as speech recognition is developed by incorporating an idea of non-linear time alignment into the kernel function. Since the time-alignment operation of sequential pattern is embedded in the new kernel function, standard SVM training and classification algorithms can be employed without further modifications. The proposed SVM (DTAK-SVM) is evaluated in speaker-dependent speech recognition experiments of hand-segmented phoneme recognition. Preliminary experimental results show comparable recognition performance with hidden Markov models (HMMs).
recognition that much. This is because SVM assumes that each sample is a vector
of fixed dimension, and hence it can not deal with the variable length sequences
directly. Because of this, most of the efforts that have been made so far to apply
SVM to speech recognition employ linear time normalization, where input feature
vector sequences with different lengths are aligned to same length [4]. A variant
of this approach is a hybrid of SVM and HMM (hidden Markov model), in which
HMM works as a pre-processor to feed time-aligned fixed-dimensional vectors to
SVM [5]. Another approach is to utilize probabilistic generative models as a SVM
kernel function. This includes the Fisher kernels [6, 7], and conditional symmetric
independence (CSI) kernels [8], both of which employ HMMs as the generative mod-
els. Since HMMs can treat sequential patterns, SVM that employs the generative
models based on HMMs can handle sequential patterns as well.

In contrast to those approaches, our approach is a direct extension of the original
SVM to the case of variable length sequence. The idea is to incorporate the op-
eration of dynamic time alignment into the kernel function itself. Because of this,
the proposed new SVM is called “Dynamic Time-Alignment Kernel SVM (DTAK-
SVM)”. Unlike the SVM with Fisher kernel that requires two training stages with
different training criteria, one is for training the generative models and the second
is for training the SVM, the DTAK-SVM uses one training criterion as well as the
original SVM.

2 Dynamic Time-Alignment Kernel

We consider a sequence of vectors $X = (x_1, x_2, \ldots, x_L)$, where $x_i \in \mathbb{R}^n$, $L$ is the
length of the sequence, and the notation $|X|$ is sometimes used to represent
the length of the sequence instead. For simplification, we at first assume the so-called
linear SVM that does not employ non-linear mapping function $\phi$. In such case, the
kernel operation in (1) is identical to the inner product operation.

2.1 Formulation for linear kernel

Assume that we have two vector sequences $X$ and $V$. If these two patterns are
equal in length, i.e. $|X| = |V| = L$, then the inner product between $X$ and $V$ can
be obtained easily as a summation of each inner product between $x_k$ and $v_k$ for
$k = 1, \ldots, L$:

$$X \cdot V = \sum_{k=1}^{L} x_k \cdot v_k,$$

and therefore an SVM classifier can be defined as given in (1). On the other hand
in case where the two sequences are different in length, the inner product can not
be calculated directly. Even in such case, however, some sort of inner product like
operation can be defined if we align the lengths of the patterns. To that end, let
$\psi(k), \theta(k)$ be the time-warping functions of normalized time frame $k$ for the pattern
$X$ and $V$, respectively, and let “$\circ$” be the new inner product operator instead of
the original inner product “.”. Then the new inner product between the two vector
sequences $X$ and $V$ can be given by

$$X \circ V = \frac{1}{L} \sum_{k=1}^{L} x_{\psi(k)} \cdot v_{\theta(k)},$$

where $L$ is a normalized length that can be either $|X|, |V|$ or arbitrary positive
integer.
There would be two possible types of time-warping functions. One is a linear time-warping function and the other is a non-linear time-warping function. The linear time-warping function takes the form as

\[ \psi(k) = \lceil (|X|/L)k \rceil, \quad \theta(k) = \lceil (|V|/L)k \rceil, \]

where \( \lceil x \rceil \) is the ceiling function which gives the smallest integer that is greater than or equal to \( x \). As it can be seen from the definition given above, the linear warping function is not suitable for continuous speech recognition, i.e. frame-synchronous processing, because the sequence lengths, \( |X| \) and \( |V| \), should be known beforehand. On the other hand, non-linear time warping, or dynamic time warping (DTW) [9] in other word, enables frame-synchronous processing. Furthermore, the past research on speech recognition has shown that the recognition performance by the non-linear time normalization outperforms the one by the linear time normalization. Because of these reasons, we focus on the non-linear time warping based on DTW.

Though the original DTW uses a distance/distortion measure and finds the optimal path that minimizes the accumulated distance/distortion, the DTW that is employed for SVM uses inner product or kernel function instead and finds the optimal path that maximizes the accumulated similarity:

\[
X \circ V = \max_{\psi, \theta} \frac{1}{M_{\psi, \theta}} \sum_{k=1}^{L} m(k) x_{\psi(k)} \cdot v_{\theta(k)},
\]

subject to

\[
1 \leq \psi(k) \leq \psi(k + 1) \leq |X|, \quad \psi(k + 1) - \psi(k) \leq Q,
\]

\[
1 \leq \theta(k) \leq \theta(k + 1) \leq |V|, \quad \theta(k + 1) - \theta(k) \leq Q,
\]

where \( m(k) \) is a nonnegative (path) weighting coefficient, \( M_{\psi, \theta} \) is a (path) normalizing factor, and \( Q \) is a constant constraining the local continuity. In the standard DTW, the normalizing factor \( M_{\psi, \theta} \) is given as \( \sum_{k=1}^{L} m(k) \), and the weighting coefficients \( m(k) \) are chosen so that \( M_{\psi, \theta} \) is independent of the warping functions.

The above optimization problem can be solved efficiently by dynamic programming. The recursive formula in the dynamic programming employed in the present study is as follows

\[
G(i, j) = \max \left\{ \begin{array}{l}
G(i - 1, j) + \text{Inp}(i, j), \\
G(i - 1, j - 1) + 2 \text{Inp}(i, j), \\
G(i, j - 1) + \text{Inp}(i, j),
\end{array} \right. \]

where \( \text{Inp}(i, j) \) is the standard inner product between the two vectors corresponding to point \( i \) and \( j \). As a result, we have

\[
X \circ V = G(|X|, |V|)/(|X| + |V|).
\]

### 2.2 Formulation for non-linear kernel

In the last subsection, a linear kernel, i.e. the inner product, for two vector sequences with different lengths has been formulated in the framework of dynamic time-warping. With a little constraint, similar formulation is possible for the case where SVM’s non-linear mapping function \( \Phi \) is applied to the vector sequences. To that end, \( \Phi \) is restricted to the one having the following form:

\[
\Phi(X) = (\phi(x_1), \phi(x_2), \ldots, \phi(x_L)),
\]

where \( \phi \) is a non-linear mapping function that is applied to each frame vector \( x_i \), as given in (1). It should be noted that under the above restriction \( \Phi \) preserves the original length of sequence at the cost of losing long-term correlations such as the
one between $x_1$ and $x_L$. As a result, a new class of kernel can be defined by using the extended inner product introduced in the previous section;

$$K_s(X, V) = \Phi(X) \circ \Phi(V)$$

$$= \max_{\psi, \theta} \frac{1}{M_{\psi, \theta}} \sum_{k=1}^{L} m(k) \phi(x_{\psi(k)}) \cdot \phi(v_{\theta(k)})$$

$$= \max_{\psi, \theta} \frac{1}{M_{\psi, \theta}} \sum_{k=1}^{L} m(k) K(x_{\psi(k)}, v_{\theta(k)}).$$

We call this new kernel “dynamic time-alignment kernel (DTAK)”.

### 2.3 Properties of the dynamic time-alignment kernel

It has not been proven that the proposed function $K_s(\cdot, \cdot)$ is really an SVM’s admissible kernel which guarantees the existence of a feature space. This is because that the mapping function to a feature space is not independent but dependent on the given vector sequences. Although a class of data-dependent asymmetric kernel for SVM has been developed in [10], our proposed function is more complicated and difficult to analyze because the input data is a vector sequence with variable length and non-linear time normalization is embedded in the function. Instead, what have been known about the proposed function so far are (1) $K_s$ is symmetric, (2) $K_s$ satisfies the Cauchy-Schwartz like inequality described bellow:

**Proposition 1**

$$K_s(X, V)^2 \leq K_s(X, X)K_s(V, V)$$

**Proof** For simplification, we assume that normalized length $L$ is fixed, and omit $m(k)$ and $M_{\psi, \theta}$ in (11). Using the standard Cauchy-Schwartz inequality, the following inequality holds:

$$K_s(X, V) = \max_{\psi, \theta} \sum_{k=1}^{L} \phi(x_{\psi(k)}) \cdot \phi(v_{\theta(k)}) = \sum_{k=1}^{L} \phi(x_{\psi^+(k)}) \cdot \phi(v_{\theta^+(k)})$$

$$\leq \sum_{k=1}^{L} \| \phi(x_{\psi^+(k)}) \| \| \phi(v_{\theta^+(k)}) \|,$n

where $\psi^+(k), \theta^+(k)$ represent the optimal warping functions that maximize the RHS of (13). On the other hand,

$$K_s(X, X) = \max_{\psi, \theta} \sum_{k=1}^{L} \phi(x_{\psi(k)}) \cdot \phi(x_{\theta(k)}) = \sum_{k=1}^{L} \phi(x_{\psi^+(k)}) \cdot \phi(x_{\theta^+(k)}).$$

Because here we assume that $\psi^+(k), \theta^+(k)$ are the optimal warping functions that maximize (15), for any warping functions including $\psi^+(k)$, the following inequality holds:

$$K_s(X, X) \geq \sum_{k=1}^{L} \phi(x_{\psi^+(k)}) \cdot \phi(x_{\psi^+(k)}) = \sum_{k=1}^{L} \| \phi(x_{\psi^+(k)}) \|^2.$$

In the same manner, the following holds:

$$K_s(V, V) \geq \sum_{k=1}^{L} \phi(v_{\theta^+(k)}) \cdot \phi(v_{\theta^+(k)}) = \sum_{k=1}^{L} \| \phi(v_{\theta^+(k)}) \|^2.$$
Therefore,

\[ K_s(X, X)K_s(V, V) - K_s(X, V)^2 \]

\[ \geq \left( \sum_{k=1}^{L} \| \phi(x_{\omega(k)}) \|^2 \right) \left( \sum_{k=1}^{L} \| \phi(v_{\omega(k)}) \|^2 \right) - \left( \sum_{k=1}^{L} \| \phi(x_{\omega(k)}) \| \| \phi(v_{\omega(k)}) \| \right)^2 \]

\[ = \sum_{i=1}^{L} \sum_{j=i+1}^{L} \left( \| \phi(x_{\omega(i)}) \| \| \phi(v_{\omega(j)}) \| - \| \phi(x_{\omega(i)}) \| \| \phi(v_{\omega(j)}) \| \right)^2 \geq 0 \quad (18) \]

3 DTAK-SVM

Using the dynamic time-alignment kernel (DTAK) introduced in the previous section, the discriminant function of SVM for a sequential pattern is expressed as

\[ g(X) = \sum_{i=1}^{N} \alpha_i y_i \Phi(X^{(i)}) \circ \Phi(X) + b \quad (19) \]

\[ = \sum_{i=1}^{N} \alpha_i y_i K_s(X^{(i)}, X) + b, \quad (20) \]

where \( X^{(i)} \) represents the \( i \)-th training pattern. As it can be seen from these expressions, the SVM discriminant function for time sequence has the same form with the original SVM except for the difference in kernels. It is straightforward to deduce the learning problem which is given as

\[ \min_{W, b, \xi_i} \quad \frac{1}{2} W \circ W + C \sum_{i=1}^{N} \xi_i, \quad (21) \]

subject to

\[ y_i (W \circ \Phi(X^{(i)}) + b) \geq 1 - \xi_i, \quad (22) \]

\[ \xi_i \geq 0, \quad i = 1, \cdots, N. \]

Again, since the formulation of learning problem defined above is almost the same with that for the original SVM, same training algorithms for the original SVM can be used to solve the problem.

4 Experiments

Speech recognition experiments were carried out to evaluated the classification performance of DTAK-SVM. As our objective is to evaluate the basic performance of the proposed method, very limited task, hand-segmented phoneme recognition task in which positions of target patterns in the utterance are known, was chosen. Continuous speech recognition task that does not require phoneme labeling would be our next step.

4.1 Experimental conditions

The details of the experimental conditions are given in Table 1. The training and evaluation samples were collected from the ATR speech database: A-set (5240
Japanese words in vocabulary). In consonant-recognition task (Experiment-1), only six voiced-consonants /b,d,g,m,n,N/ were used to save time. The classification task of those 6 phonemes without using contextual information is considered as a relatively difficult task, whereas the classification of 5 vowels /a,i,u,e,o/ (Experiment-2) is considered as an easier task.

To apply SVM that is basically formulated as a two-class classifier to the multi-class problem, “one against the others” type of strategy was chosen. The proposed DTAK-SVM has been implemented with the publicly available toolkit, SVMTorch [11].

4.2 Experimental results

Fig. 1 depicts the experimental results for Experiment-1, where average values over 5 speakers are shown. It can be seen in Fig. 1 that the best performance of 95.8% was achieved at $\gamma = 2.0$ and $C = 10$. Similar results were obtained for Experiment-2 as given in Fig. 2.
Figure 2: Experimental results for Experiment-2 (5 vowels recognition) showing (a) correct classification rate and (b) the number of SVs as a function of $\gamma$ (the parameter of RBF kernel).

Table 2: Recognition performance comparison of DTAK-SVM with HMM. Results of Experiment-1 for 1 male and 1 female speakers are shown. (numbers represent correct classification rate [%])

<table>
<thead>
<tr>
<th>Model</th>
<th># training samples/phoneme</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>male</td>
</tr>
<tr>
<td></td>
<td>50</td>
</tr>
<tr>
<td>HMM (1 mix.)</td>
<td>75.0</td>
</tr>
<tr>
<td>HMM (4 mix.)</td>
<td>83.3</td>
</tr>
<tr>
<td>HMM (8 mix.)</td>
<td>82.8</td>
</tr>
<tr>
<td>HMM (16 mix.)</td>
<td>79.9</td>
</tr>
<tr>
<td>DTAK-SVM</td>
<td>83.8</td>
</tr>
</tbody>
</table>

Next, the classification performance of DTAK-SVM was compared with that of the state-of-the-art HMM. In order to see the effect of generalization performance on the size of training data set and model complexity, experiments were carried out by varying the number of training samples (50, 100, 200), and mixtures (1,4,8,16) for each state of HMM. The HMM used in this experiment was a 3-states, continuous density, Gaussian-distribution mixtures with diagonal covariances, context-independent model. HTK [12] was employed for this purpose. The parameters of DTAK-SVM were fixed to $C = 10, \gamma = 2.0$. The results for Experiment-1 with respect to 1 male and 1 female speakers are given in Table 2.

It can be said from the experimental results that DTAK-SVM shows better classification performance when the number of training samples is 50, while comparable performance when the number of samples is 200. One might argue that the number of training samples used in this experiment is not enough at all for HMM to achieve best performance. But such shortage of training samples occurs often in HMM-based real-world speech recognition, especially when context-dependent models are employed, which prevents HMM from improving the generalization performance.
5 Conclusions

A novel approach to extend the SVM framework for the sequential-pattern classification problem has been proposed by embedding a dynamic time-alignment operation into the kernel. Though long-term correlations between the feature vectors are omitted at the cost of achieving frame-synchronous processing for speech recognition, the proposed DTAK-SVMs demonstrated comparable performance in hand-segmented phoneme recognition with HMMs. The DTAK-SVM is potentially applicable to continuous speech recognition with some extension of One-pass search algorithm [9].

References