JOINT DECODING OF MULTIPLE SPEECH PATTERNS FOR ROBUST SPEECH RECOGNITION

Nishanth Ulhas Nair and T.V. Sreenivas

Department of Electrical Communication Engineering, Indian Institute of Science, Bangalore, India

catchnishanth@gmail.com, tvsree@ece.iisc.ernet.in

ABSTRACT

We are addressing a new problem of improving automatic speech recognition performance, given multiple utterances of patterns from the same class. We have formulated the problem of jointly decoding K multiple patterns given a single Hidden Markov Model. It is shown that such a solution is possible by aligning the K patterns using the proposed Multi Pattern Dynamic Time Warping algorithm followed by the Constrained Multi Pattern Viterbi Algorithm. The new formulation is tested in the context of speaker independent isolated word recognition for both clean and noisy patterns. When 10 percent of speech is affected by a burst noise at -5 dB Signal to Noise Ratio (local), it is shown that joint decoding using only two noisy patterns reduces the noisy speech recognition error rate to about 51 percent, when compared to the single pattern decoding using the Viterbi Algorithm. In contrast a simple maximization of individual pattern likelihoods, provides only about 7 percent reduction in error rate.

Index Terms— Robust Speech Recognition, Viterbi Algorithm, Dynamic Time Warping, Burst Noise

1. INTRODUCTION

In our normal day to day telephone conversations, it is quite common to ask the person speaking to us, to repeat certain portions of their speech, because we don't understand it. This happens more often in the presence of background noise where the intelligibility of speech is affected significantly. Although exact nature of how humans decode multiple repetitions of speech is not known, it is quite possible that we use the combined knowledge of the multiple utterances and decode the unclear word or phrase. If humans, with exceptional speech recognition capabilities, require repetitions of spoken words, then it is more important that speech recognition machines to utilize such repeated information, especially in the presence of heavy/bursty background noise.

The problem that we are formulating is: how to increase automatic speech recognition (ASR) performance given multiple utterances (repetitions) of the same word? If we have K test utterances ($K \geq 2$) of a word, is it possible to improve the speech recognition accuracy over a single test utterance, for the case of both clean and noisy speech?

The two classical approaches to speech pattern matching are Dynamic Time Warping (the non-parametric approach) and by using Hidden Markov Models (the parametric approach). We have developed a novel formulation in which we use both the parametric and non parametric approaches for speech recognition - a hybrid approach - to solve the problem of jointly decoding multiple speech patterns. This is achieved by using the proposed Multi Pattern Dynamic Time Warping (MPDTW) algorithm followed by the new Constrained Multi Pattern Viterbi Algorithm (CMPVA). We experimented the new algorithms for both clean speech and speech with burst noise for Isolated Word Recognition (IWR). Similar extensions are possible for connected word recognition and continuous speech recognition cases also.

In speech recognition, researchers have proposed various methods to handle burst noises. A burst or impulse noise could be a door slam or a lip smack. Various versions of the Viterbi Algorithm (VA) [1] has also been proposed to handle burst noise, for example the recently proposed weighted VA [2]. A Robust VA to handle short impulsive noises with unknown characteristics by means of joint decoding and detection during Viterbi Search was proposed in [3]. In our paper, we remove the noise, which is impulsive with unknown characteristics, by choosing the clean speech portion from the other clean utterances.

A time-synchronous Viterbi-style beam search procedure called the N-best algorithm, has been proposed in literature [4] to find the N most likely whole sentence alternatives that are within a given "beam" of the most likely sequence. This N-best algorithm was used to simultaneously decode multiple utterances to derive one or more allophonic transcriptions for each word in [5]. Work has been done to extend HMMs to two-dimensions, to offer a more realistic approach to speech recognition. A 2D extension of Hidden Markov Model (HMM) was introduced in [6] to improve the modeling of speech signals. A 3D HMM search space and a Viterbi-like decoding algorithm was proposed for Utterance Verification [7]. In [7], the two axes in the trellis belonged to HMM states and one axis belongs to the observational sequence. In our paper, for the CMPVA, we have one axis for the HMM states, and K axes, one for each of the K observational sequences. We try to use multiple utterances of a word by the same speaker to improve speech recognition performance using this CMPVA whose time path is fixed by the MPDTW algorithm.

Dynamic Time Warping (DTW) has been used in areas other than speech recognition to deal with multidimensional data. For sequences which are multidimensional, like on-line signature sequences, 2D curves, etc., an Extended R-squared [8] is proposed as a similarity measure. It was used for multidimensional sequence matching and it was coupled with DTW to enhance robustness in signature verification. For large sequence databases, an effective processing of similarity search that supports time warping was introduced in [9]. Dynamic Space Warping, which is similar to DTW, was used to determine the pose angle of a face [10], even from a 2D face image. DTW was extended to deal with multi modal sequences [11] which consist of the data or feature sequences acquired from multiple heterogeneous sensors over a period of time.

However, none of these papers are addressing the problem of jointly decoding multiple utterances of spoken words to improve speech recognition performance. To the best of our knowledge, the

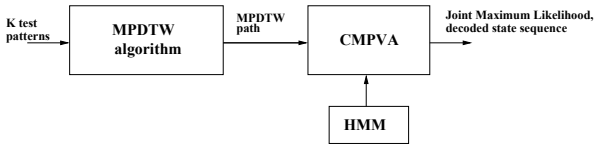

Fig. 1. *Joint decoding of K patterns*

problem of multi pattern joint decoding has not been addressed in the speech recognition literature. We illustrate that by using even two utterances, we can get a significant improvement in speech recognition accuracy for speech with burst noise.

2. JOINT DECODING OF MULTIPLE PATTERNS

We have K number of observational speech sequences (patterns) $O_1^{T_1,1}, O_1^{T_2,2}, \ldots, O_1^{T_K,K}$, of lengths $T_1, T_2, \ldots, T_K$, respectively, where $O_1^{T_i,i} = (O_1^i, O_2^i, \ldots, O_{T_i}^i)$ is the observational sequence of the i^{th} pattern and $O_{t_i}^i$ is the feature vector of the i^{th} pattern at time frame t_i. Let each of these K observation sequences belong to the same pattern class (spoken word). They are different utterances of the same word by the same speaker.

For the sake of simplicity, let us define the following terms: $\bar{T} = (T_1, T_2, \ldots, T_K)$, $\bar{1} = (1, 1, \ldots, 1)$, $\bar{t} = (t_1, t_2, \ldots, t_K)$. For joint decoding of multiple patterns using the HMM λ, the objective is to maximize P($O_1^{T_1,1}, O_1^{T_2,2}, \ldots, O_1^{T_K,K}/\lambda$) jointly. Since $O_1^{T_1,1}, O_1^{T_2,2}, \ldots, O_1^{T_K,K}$ are of different lengths and uttered independently, a new multi-dimensional trellis is constructed for joint decoding. In this case, state sequence $q_{\bar{1}}^{\bar{T}} = (q_{\bar{1}}, \ldots, q_{\bar{T}})$, where $q_{\bar{t}}$ is the HMM state at the joint time vector $(t_1, t_2, \ldots, t_K)$ and a coordinate in the K+1 dimensional space, where t_i is the time frame when $O_{t_i}^i$ has occurred. We are now looking at a multi-dimensional trellis having K+1 dimensions, where K time dimensions are for the K utterances (patterns) of the repeated words, and one dimension is for the HMM states (K+1 dimensional space). The most likely state sequence $\hat{q}_{\bar{1}}^{\bar{T}}$ is defined as

$$\hat{q}_{\bar{1}}^{\bar{T}} = \arg\max_{q_{\bar{1}}^{\bar{T}}} log\ P(q_{\bar{1}}^{\bar{T}}/O_1^{T_1,1}, \ldots, O_1^{T_K,K}, \lambda) \quad (1)$$

$$\hat{q}_{\bar{1}}^{\bar{T}} = \arg\max_{q_{\bar{1}}^{\bar{T}}} log\ P(q_{\bar{1}}^{\bar{T}}, O_1^{T_1,1}, \ldots, O_1^{T_K,K}/\lambda) \quad (2)$$

We can perform this joint decoding using the MPDTW algorithm followed by the CMPVA. Using the MPDTW algorithm we find the "least distortion joint warping path" (which is the "most similar path" or the "aligned path" or the "MPDTW path") between the K test patterns belonging to the same class. We then fit a layer of HMMs on this path $\{(K+1)^{th} axis\}$ and then apply the CMPVA for decoding the HMM states, as shown in Figure 1. Here, the CMPVA, and not MPDTW algorithm, is used for IWR. MPDTW is used only for finding the aligned path between the K patterns.

3. MULTI PATTERN DYNAMIC TIME WARPING (MPDTW)

The Dynamic Time Warping (DTW) [12] gives us the least distortion path between two given patterns. We extend this case to that of multi-pattern DTW wherein an optimum path in multi-dimensional space is determined to optimally warp all the K patterns jointly, leading to the minimum distortion, referred to as MPDTW path. As in standard DTW, all K patterns are warped with respect to each other and they all belong to the same pattern class. (The case of test pattern and reference pattern coming from different classes does not arise in our application. Here, all the patterns are test patterns of a word spoken by the same speaker.) To find the MPDTW path, we need to traverse through a multi-dimensional trellis which has K time axes.

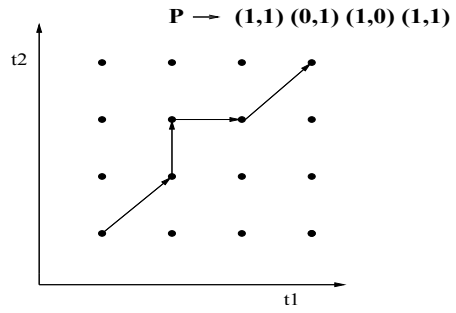

Fig. 2. *An example path P for K = 2*

We define a path P (see [12]) as a sequence (concatenation) of moves in the trellis diagram, each specified by a set of coordinate *increments*, i.e.,

$$P \rightarrow (p_1^1, p_1^2, \ldots, p_1^K)(p_2^1, p_2^2, \ldots, p_2^K) \ldots (p_T^1, p_T^2, \ldots, p_T^K) \quad (3)$$

where p_k^i is the increment at step k by utterance i (i^{th} dimension). An example of a path P through a trellis, when number of patterns K = 2, is shown in Figure 2.

Let the $k = 1$ step correspond to $(1, 1, \ldots, 1)$, where $(1, 1, \ldots, 1)$ is the staring point in the trellis where all the K utterances begin. Let us set $p_1^1 = p_1^2 = \ldots = p_1^K = 1$ (as if the path originates from $(0, 0, \ldots, 0)$). Let $k = T$ correspond to $(T_1, T_2, \ldots, T_K)$, which is the ending point in the trellis. $\phi_1(k)$, $\phi_2(k)$, ..., $\phi_K(k)$ are K warping functions such that $\phi_i(k) = t_i$ for the i^{th} utterance. Let us define:

$$\phi_l(k) = \sum_{i=1}^{k} p_i^l \quad (4)$$

$l = 1, 2, \ldots, K$
The coordinate *increments* satisfy the constraints:

$$\sum_{k=1}^{T} p_k^l = T_l \quad (5)$$

$l = 1, 2, \ldots, K$
Endpoint constraints are as follows:

$$\phi_1(1) = 1, \ldots, \phi_K(1) = 1 \quad (6)$$

$$\phi_1(T) = T_1, \ldots, \phi_K(T) = T_K \quad (7)$$

Relaxed end pointing can also be introduced. Various types of Local Continuity Constraints (LCCs) and Global Path Constraints as defined for DTW, are extended to K dimensional space. We define a multi-vector distance measure $d(t_1, t_2, \ldots, t_K)$ at time $\bar{t}$, between the K vectors $O_{t_1}^1, O_{t_2}^2, \ldots, O_{t_K}^K$, as follows:

$$d(t_1, \ldots, t_K) = \sum_{i=1}^{K} d(O_{t_i}^i, C_{\bar{t}}) \quad (8)$$

where $C_{\bar{t}}$ is the centroid of the K vectors $O_{t_1}^1, O_{t_2}^2, \ldots, O_{t_K}^K$ and $d(O_{t_i}^i, C_{\bar{t}})$ is the Euclidean distance between $O_{t_i}^i$ and $C_{\bar{t}}$.

m(k) is the slope weighting function which controls the contribution of $d(\phi_1(k), \ldots, \phi_K(k))$ and

$$M_\phi = \sum_{k=1}^{T} m(k) \tag{9}$$

where M_ϕ is the global normalization factor.

3.1. MPDTW Algorithm

Let $D(t_1, t_2, \ldots, t_K)$ be the accumulated cost function, which is to be minimized.
1. Initialization

$$D(1, \ldots, 1) = d(1, \ldots, 1)m(1) \tag{10}$$

2. Recursion

$$\begin{aligned} D(t_1, \ldots, t_K) \quad = \quad & \min_{(t_1^{'}, \ldots, t_K^{'})} [D(t_1^{'}, \ldots, t_K{}') + \\ & \zeta((t_1^{'}, \ldots, t_K^{'})(t_1{}^1, \ldots, t_K{}^K))] \end{aligned} \tag{11}$$

where $(t_1^{'}, \ldots, t_K{}')$ are the candidate values as given by the LCC and

$$\begin{aligned} & \zeta((t_1^{'}, \ldots, t_K^{'})(t_1^1, \ldots, t_K^K)) \quad = \\ & \sum_{l=0}^{L_s} d(\phi_1(T^{'} - l), \ldots, \phi_K(T^{'} - l))m(T^{'} - l) \end{aligned} \tag{12}$$

$\phi_1(T^{'}) = t_1$, …, $\phi_K(T^{'}) = t_K$ and $\phi_1(T^{'} - L_s) = t_1^{'}$, …, $\phi_K(T^{'} - L_s) = t_K^{'}$ where L_s being the number of moves in the path from $(t_1^{'}, \ldots, t_K{}')$ to $(t_1, \ldots, t_K)$ according to $\phi_1, \ldots, \phi_K$.
A backtracking pointer I is defined to remember the best path.

$$\begin{aligned} I(t_1, \ldots, t_K) \quad = \quad & \arg \min_{(t_1^{'}, \ldots, t_K^{'})} [D(t_1^{'}, \ldots, t_K{}') + \\ & \zeta((t_1^{'}, \ldots, t_K^{'})(t_1{}^1, \ldots, t_K{}^K))] \end{aligned} \tag{13}$$

3. Termination

$$d(O_1^{T_1,1}, \ldots, O_1^{T_K,K}) = D(T_1, \ldots, T_K)/M_\phi \tag{14}$$

where $d(O_1^{T_1,1}, \ldots, O_1^{T_K,K})$ is the total distortion between $O_1^{T_1,1}$, …, $O_1^{T_K,K}$.

4. Path Backtracking
Path backtracking is done using the path back tracking pointer I.

$$(t_1^*, \ldots, t_K^*) = I(t_1, \ldots, t_K) \tag{15}$$

$$(t_1, \ldots, t_K) = (t_1^*, \ldots, t_K^*) \tag{16}$$

where $(t_1, \ldots, t_K) = (T_1, \ldots, T_K), \ldots, (1, \ldots, 1)$

We now get the least distortion path (MPDTW path) for K patterns, which gives us the most similar non linear time warped path between them. An example of a MPDTW path for 3 utterances (P1, P2, P3) of the word "Voice Dialer" by one female speaker is shown in Figure 3.

Let ϕ be the MPDTW path for K patterns. Let $\phi(k) = (t_1, \ldots, t_K)$ where $(t_1, \ldots, t_K)$ is a point on the MPDTW path. $\phi(1) = (1, \ldots, 1)$ and $\phi(T) = (T_1, \ldots, T_K)$. $\phi = (\phi(1), \phi(2), \ldots, \phi(K))$.

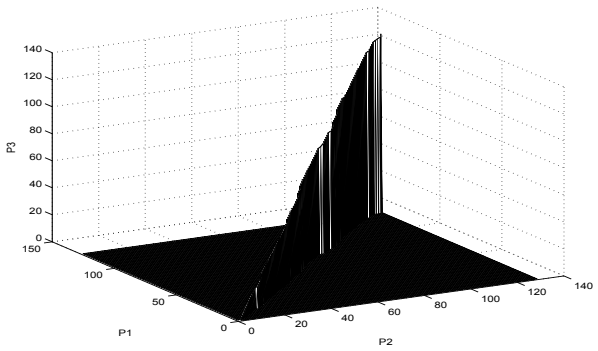

Fig. 3. *MPDTW path for 3 utterances of the word "Voice Dialer"*

4. CONSTRAINED MULTI PATTERN VITERBI ALGORITHM (CMPVA) USING VARIOUS DISCRIMINANT LIKELIHOODS

For a T length observational sequence, $O_1^T = (O_1, O_2, \ldots, O_T)$, denote the state sequence to be $q_1^T = (q_1, q_2, \ldots, q_T)$, where q_t is state index at time frame t. The standard Viterbi Algorithm (VA) [1] is used to search for the optimum state sequence $\hat{q}_1^T$ of a given HMM λ for the maximum likelihood of the given observation sequence. Now we have K number of observational sequences $O_1^{T_1,1}$, …, $O_1^{T_K,K}$, of lengths $T_1, \ldots, T_K$, respectively, where $O_1^{T_i,i} = (O_1^i, O_2^i, \ldots, O_{t_i}^i, \ldots, O_{T_i}^i)$ is the observation sequence of the i^{th} pattern and $O_{t_i}^i$ is the feature vector of the i^{th} pattern at time frame t_i. The path among the time axes is fixed according to the MPDTW (ϕ) path. The CMPVA is used to find the optimum state sequence and joint maximum likelihood of K patterns. We perform the CMPVA along the MPDTW path to search for the optimum state sequence. We call the Constrained Multi Pattern Viterbi Algorithm "constrained" because the path traversed in the time axes is fixed by the MPDTW algorithm. To solve equations (1) and (2), we use the CMPVA, to find the most likely state sequence $\hat{q}_{\phi(1)}^{\phi(T)} = (q_{\phi(1)}, \ldots, q_{\phi(T)})$, where $q_{\phi(k)}$ is the state index at time $(t_1, \ldots, t_K)$.

We define $\delta_{\phi(k)}(j)$ as the log likelihood of the first $\phi(k)$ observations of the K patterns through the most likely state sequence up to time $\phi(k-1)$ with $q_{\phi(k)} = j$. Mathematically, $\delta_{\phi(k)}(j)$ is defined as,

$$\begin{aligned} \delta_{\phi(k)}(j) \quad = \quad & \max_{q_{\phi(1)}^{\phi(k-1)}} log\ P(O_1^{t_1,1}, \ldots, O_1^{t_K,K}, q_{\phi(1)}^{\phi(k-1)}, \\ & q_{\phi(k)} = j/\lambda) \end{aligned} \tag{17}$$

A recursive equation can be derived as follows:

$$\delta_{\phi(k)}(j) = \max_i [\ \delta_{\phi(k-1)}(i) + log\ a_{ij}] + log\ b_j(O_{t_1}^1, \ldots, O_{t_K}^K) \tag{18}$$

$k = (2, 3, \ldots, T), j = 1, 2, \ldots, N$
Initialization is done as follows:

$$\delta_{\phi(1)}(j) = log\ \Pi_j + log\ b_j(O_1^1, \ldots, O_1^K), \tag{19}$$

$j = 1, 2, \ldots, N$
where N is the number of states in the HMM. a_{ij} is the state transition probability between two states $q_{\phi(k-1)} = i$ and $q_{\phi(k)} = j$, Π_j is the state initial probability at state j, $log\ b_j(O_{t_1}^1, \ldots, O_{t_K}^K)$ is

the joint likelihood of the observations $O^1_{t_1}, \ldots, O^K_{t_K}$ generated by state j. Equation (18) is an approximation of equation (17). This is because in equation (18), each state j is emitting a fixed number (K) of feature vectors at a each time instant. So it is possible that some of the feature vectors from some utterances are reused based on the MPDTW path. So we are basically creating a new virtual utterance which is some kind of a combination of all the K utterances (with repetitions of feature vectors possible) and which lies on the MPDTW path. We are decoding this virtual utterance.

$$\psi_{\phi(k)}(j) = \arg\max_i [\, \delta_{\phi(k-1)}(i) + log\ a_{ij}] \tag{20}$$

where $k = (2, 3, \ldots, T)$, $j = 1, 2, \ldots, N$, and $\psi_{\phi(k)}(j)$ is a back tracking pointer which stores the value of best previous state.
The algorithm is terminated as follows:

$$P^* = \max_{1 \le i \le N} \delta_{\phi(T)}(i) \tag{21}$$

P^* is the joint maximum likelihood or final CMPVA likelihood.

$$\hat{q}_{\phi(T)} = \arg \max_{1 \le i \le N} [\delta_{\phi(T)}(i)] \tag{22}$$

Path Backtracking is done as follows:

$$\hat{q}_{\phi(k)} = \hat{q}_{\phi(k+1)} \tag{23}$$

where k = T-1, . . . , 1.
For IWR, we use the CMPVA to calculate the probability P^* of the optimal sequence.

As said before, we are looking at a multi-dimensional trellis having K+1 dimensions, where K dimensions belong to time axes of K utterances and +1 dimension is the HMM states. To determine the joint likelihood $P(O^1_{t_1}, \ldots, O^K_{t_K}/j, \lambda)$ we can resort to various formulations. Since the path traversed in the time axes is already optimized, how we choose $b_j(O^1_{t_1}, \ldots, O^K_{t_K})$ affects only the final CMPVA likelihood P^* (equation (21)) and the state sequence. We define some criteria for calculating $b_j(O^1_{t_1}, \ldots, O^K_{t_K})$ as below:

4.1. Criteria 1

We know that $O_1^{T_1,1}$, $O_1^{T_2,2}$, . . ., $O_1^{T_K,K}$ are different speech patterns which come from the same class (word) and uttered by the same speaker. Given that they come from the same class these patterns are uttered independently. Even though the feature vectors $O^1_{t_1}$, $O^2_{t_2}$, . . ., $O^K_{t_K}$ come from the same class, we can assume that they are independent if it is given that they occur from the same state j, so as to compute the joint likelihood of the vectors being emitted from the HMM. So we get equation (24).

$$b_j(O^1_{t_1}, \ldots, O^K_{t_K}) = b_j(O^1_{t_1}).b_j(O^2_{t_2}) \ldots b_j(O^K_{t_K}) \tag{24}$$

where $b_j(O^i_{t_i})$ is likelihood of observation $O^i_{t_i}$ emitted by state j. The independence assumption is also valid because successive vectors in a pattern are only linked through the underlying Markov model and the emission densities act only one symbol at a time. If $O^i_{t_i}$ is emitted from its actual state j from the correct HMM model λ, we can expect that $b_j(O^i_{t_i})$ to have a higher value than that if $O^i_{t_i}$ is emitted from state j of the wrong model. And taking all the product of all the $b_j(O^i_{t_i})$ brings in a kind of "reinforcing effect". Therefore, while doing IWR, the values of final CMPVA likelihood P* using the correct model and the P^* when using the other mismatched models, is likely to widen. Therefore we can expect better speech recognition accuracy to improve. Even if some of the K vectors are noise, this reinforcing affect will improve speech recognition because the rest of the vectors are clean. In Criteria 1, we are not excluding any of the K patterns to calculate P^*.

4.2. Criteria 2

Let us consider the case when some (or all) of the K utterances is affected by burst noise somewhere randomly. It can also be that some parts of some (or all) utterances may be badly articulated. We have K feature vectors at each point in the trellis. Since we are considering distorted vectors for the point of increasing the likelihood, it would be better to choose only the "best" vector, for each state; i.e., we choose $b_j(O^1_{t_1}, \ldots, O^K_{t_K})$ as follows:

$$b_j(O^1_{t_1}, \ldots, O^K_{t_K}) = \max(b_j(O^1_{t_1}), b_j(O^2_{t_2}), \ldots, b_j(O^K_{t_K})) \tag{25}$$

If the speech patterns are affected by noise, we would expect Criteria 2 to give better speech recognition accuracy than Criteria 1 because we are leaving out the noisy vector/s, and choosing only the best one to calculate P^*. However, for the case of clean speech, it is possible that Criteria 2 can reduce speech recognition accuracy than Criteria 1 because the max operation will also increase the likelihood P^* for the mismatched model and bring it closer to the P^* of the correct model. Also, this reinforcing effect will be absent in Criteria 2. So we would prefer to use Criteria 2 when the speech patterns are noisy or badly spoken, and for the clean speech case we would prefer Criteria 1. Note that in Criteria 2 (unlike Criteria 1), at every point in the trellis, we are using only one feature vector among the K utterances to calculate $b_j(O^1_{t_1}, \ldots, O^K_{t_K})$. That is, to calculate P^* we are using only one feature vector at a time from one of the K patterns, at every time instant.

4.3. Criteria 3

In Criteria 1 we use all speech patterns at each point in the trellis, in Criteria 2 we use only one pattern at a time to calculate P^*. Now we look at the case when at some points in the path taken inside the trellis, we use all the K patterns and at some other points we use only one pattern at a time. Basically we use a combination of Criteria 1 and Criteria 2 switched based on a threshold. Equation (26) below gives us Criteria 3.

$$b_j(O^1_{t_1}, \ldots, O^K_{t_K}) = \begin{cases} \left[b_j(O^1_{t_1}).b_j(O^2_{t_2}) \ldots b_j(O^K_{t_K})\right]^{\frac{1}{K}} \\ \quad \text{if } d(t_1, t_2, \ldots, t_K) < \gamma \\ \\ \max(b_j(O^1_{t_1}), b_j(O^2_{t_2}), \ldots, b_j(O^K_{t_K})) \\ \quad \text{if } d(t_1, t_2, \ldots, t_K) \ge \gamma \end{cases} \tag{26}$$

where γ is a threshold and $d(t_1, t_2, \ldots, t_K)$ is the multi-vector distance between K vectors $O^1_{t_1}$, $O^2_{t_2}$, . . ., $O^K_{t_K}$ as defined in equation (8).

For bursty noise case (or even mispronunciation), it is likely that some vectors in one speech pattern (or more) are corrupted and the others are intact. The goal is to determine a robust likelihood estimation measure, in spite of the noise. The first option of $d(t_1, t_2, \ldots, t_K) < \gamma$ is provided to take care of the statistical variation among the patterns, even without noise. If the distortion is low (less than γ), it implies no noise and a proper alignment between patterns at that point in the trellis. However high distortion (greater than or equal to γ) could be due to misalignment as well as distortion in the patterns. So, we choose only one vector out of K vectors, corresponding to the pattern which gives the maximum probability of occurrence with respect to state j. Thus, only a clean vector is chosen to calculate joint maximum likelihood P^*. We should not do this max operation at all time instants, because it can reduce the gap between the likelihood P^* when we test the patterns with respect to the correct model and the likelihood P^* when we test the

patterns with other mismatched models. The increased likelihoods can reduce the gap between the likelihood of the correct model and the likelihood of the closest competitor of the test word, can lead to deteriorating the speech recognition performance.

In equation (26), if we choose $\gamma = \infty$ (infinity), then $b_j(O^1_{t_1}, \ldots, O^K_{t_K})$ is always equal to $[b_j(O^1_{t_1}).b_j(O^2_{t_2}) \ldots b_j(O^K_{t_K})]^{1/K}$ (product operation), and when $\gamma < 0$, then it is always equal to $\max\{b_j(O^1_{t_1}), b_j(O^2_{t_2}), \ldots, b_j(O^K_{t_K})\}$ (max operation), corresponding to Criteria 1 and 2 respectively.

The physical significance of γ is related to the statistical variance of the vectors emitted in state j. For simplicity we can choose a fixed γ for all j or we can vary it. Since for each j, there is a mixture Gaussian density, we can choose γ to be the average standard deviation between different mixtures. The value of γ can also be determined experimentally.

4.4. Criteria 4

In Criteria 3, while doing the max operation, we are taking only the best pattern. In practice a variable number of patterns could be noisy and we would like to use the max operation only to omit the noisy patterns and use the product operation for the rest of the patterns. So we choose only pairwise distortion between two vectors at a time and define a new criteria for the joint likelihood.

Let $1 \leq m, n \leq K$ be the indices of vectors belonging to the K patterns. Let us define the clean (undistorted) set of vectors be denoted as Z, such that $m, n \in Z$ iff $d(O^m_{t_m}, O^n_{t_n}) < \gamma$, where $d(O^m_{t_m}, O^n_{t_n})$ is the Euclidean distance between $O^m_{t_m}$ and $O^n_{t_n}$. Let $\overline{Z}$ be the set of remaining vector indices, such that $Z \cup \overline{Z} = \{1, 2, \ldots, K\}$. We can search all pairs of vectors among K exhaustively, i.e., $K(K-1)/2$ combinations, since K is usually small ($K \sim 2, 3$).

$$b_j(O^1_{t_1}, \ldots, O^K_{t_K}) = \begin{cases} \prod_{\{i|i\in Z\}} \left[b_j(O^i_{t_i})\right]^{\frac{1}{r}} \\ \text{if } Z \neq \phi \\ \max_{\{k|1\leq k\leq K\}} \left(b_j(O^k_{t_k})\right) \\ \text{if } Z = \phi \end{cases} \quad (27)$$

where r is the cardinality of set Z, and ϕ stands for null set.

Note that Criteria 4 becomes same as Criteria 3 if number of patterns (utterances) K is equal to 2.

5. EXPERIMENTAL RESULTS

Based on the formulations of sections 2, 3 and 4, we conducted two experiments - A1 and A2 for speaker independent IWR along with the base line system of standard VA for a single pattern, for the cases of both clean and noisy speech. Since the normal VA uses one utterance (pattern) to make a recognition decision and the proposed algorithms use K utterances to make a decision, the comparison of results may not be fair. For a fairer comparison we formulated the experiment A1, which also uses K utterances using the standard VA and the best likelihood of the K utterances is chosen. So we compare the new algorithms (experiment A2) with this experiment A1 also.

The experiment A1 is as described. Given $O^{T_1,1}_1$, $O^{T_2,2}_1$, ..., $O^{T_K,K}_1$ as the individual patterns, we can obtain the joint likelihood score as $\alpha_j = \max_{1\leq i\leq K} P(O^{T_i,i}_1/\lambda_j)$, where λ_j are the clean word models and the VA is used to calculate $P(O^{T_i,i}_1/\lambda_j)$. We select the word as $j^* = \arg\max_j \alpha_j$. We have restricted to two patterns. For each word of a test speaker, A1 is done for utterance 1 and utterance 2, utterance 2 and 3, utterance 3 and 1. Experiment A2 is the MPDTW algorithm followed by CMPVA described in this paper. In all joint decoding experiments (A2), we have restricted to two pattern joint decoding and compared the performance with respect to single pattern decoding (VA and A1). Thus, for each word of a test speaker, utterance 1 is jointly decoded with utterance 2, utterance 2 with 3, utterance 3 with 1. Please note that in the noisy case (burst noise), all the three utterances are noisy. As the number of test utterances $K = 2$, for the new experiments we chose LCCs for MPDTW as (1,0) or (0,1) or (1,1) and the slope weighting function $m(k) = 1$.

Table 1. *Comparison of ASR percentage accuracy (ASRA) for clean speech for VA, A1, and A2 using Criteria 3 and 4 (C34).*

Algorithm	ASRA
VA	89.70%
A1	89.87%
A2, $\gamma = \infty$	91.82%, C34
A2, $\gamma = 1$	91.43%, C34
A2, $\gamma = 0.5$	91.21%, C34
A2, $\gamma < 0$	91.17%, C34

Table 2. *Comparison of ASR percentage accuracy for noisy speech for VA, A1, and A2 using Criteria 3 and 4 (C34).*

Algorithm	-5 dB ASRA	0 dB ASRA	5 dB ASRA
VA	57.13%	61.49%	67.38%
A1	60.33%	64.29%	69.49%
A2, $\gamma = \infty$	61.16%, C34	66.27%, C34	72.40%, C34
A2, $\gamma = 2$	73.91%, C34	76.89%, C34	80.20%, C34
A2, $\gamma = 1$	77.96%, C34	80.38%, C34	83.33%, C34
A2, $\gamma = 0.5$	79.02%, C34	81.62%, C34	84.40%, C34
A2, $\gamma = 0.25$	78.98%, C34	81.67%, C34	84.36%, C34
A2, $\gamma < 0$	78.98%, C34	81.67%, C34	84.36%, C34

We carried out the experiments for IISc-BPL database[1] which contains 75 word vocabulary for 36 female and 34 male adult speakers, with three repetitions for each word by the same speaker, digitized at 8kHz sampling rate. The vocabulary consists of a good number of phonetically confusing words used in Voice Dialer application. Left to Right HMMs are trained using the Segmental K Means (SKM) algorithm. 25 male and 25 female speakers are used for training, with three repetitions of each word by each speaker. We tested the algorithm for 20 unseen speakers (11 female and 9 male) in both clean and noisy cases. Test words are three utterances for each word by each speaker, at each Signal to Noise Ratio (SNR). In the noisy case, burst noise was added to 10% of the frames of each word at -5 dB, 0 dB, 5 dB SNRs to all the three utterances. (The remaining 90% of the frames are clean; the range of -5dB to +5dB indicates severe to mild degradation of the 10% frames.) The burst noise can occur randomly anywhere in the spoken word with uniform probability distribution. MFCC, Δ MFCC, and Δ^2 MFCC (to-

[1] IISc-BPL database is an Indian accented English database used for Voice Dialer application. This database consists of English isolated words, English TIMIT sentences, Native language (different for different speakers) sentences, spoken by 36 female and 34 male adult speakers recorded in a laboratory environment using 5 different recording channels: PSTN-telephone (8 KHz sampling), Cordless local phone (16 KHz sampling), Direct microphone (16 KHz sampling), Ericsson (GSM) mobile phone (8 KHz sampling), Reverberant room telephone (Sony) (8 KHz sampling).

tal 39 dimension vector) is used. Energy components are neglected and Cepstral Mean Subtraction was included. Variable number of states are used for each word model; i.e. using the average duration of the training patterns, for each second of speech, 8 HMM states were assigned, with 3 Gaussian mixtures per state.

Since in the experimentation, the number of patterns K is equal to 2, Criteria 3 and 4 are same. We experimented for various values of the threshold γ. In Criteria 3, $\gamma < 0$ implies that only the max operation is used for the joint likelihood in equations (26), and Criteria 3 is same as Criteria 2. $\gamma = \infty$ implies that only the product operation is used, and Criteria 3 corresponds to Criteria 1. As discussed earlier, only for the noisy frames we would like to use the max operation. Hence we experimented with a range of values for γ and found that there is indeed an optimum value. For the noisy patterns with burst noises at -5 dB SNR, $\gamma = 0.5$ is found to be optimum. It is also clear that $\gamma = 0$ provides closer to optimum performance than $\gamma = \infty$, indicating that the max operation is more robust than the product operation.

The results for clean speech are summarized in Table 1. Table 2 gives the results for noisy speech, where 10% of the speech is added with burst noise at a particular SNR. In these tables, ASRA stands for ASR accuracy. In the tables, for experiment A2, in the ASRA column, the ASR accuracy and the symbol C34 is written. C34 stands for experiment A2 carried out using Criteria 3 and 4 (both are same as K = 2). In these tables -5dB ASRA stands for ASR Accuracy for noisy speech which has burst noise of 10% at SNR -5dB. It can be seen that the baseline performance of VA for clean speech is close to 90%. For example, for noisy case at -5 dB SNR burst noise it decreases to $\approx$ 58%. Interestingly, the experiment A1 provides a mild improvement of $\approx$ 0.2% and 2% for clean and noisy speech (at -5dB SNR burst noise) respectively, over the VA benchmark. This shows that use of multiple patterns is indeed beneficial, but just maximization of likelihoods is weak. The proposed new algorithm of joint decoding provides dramatic improvement for the noisy case, w.r.t. the VA performance. For example at -5 dB SNR burst noise the proposed algorithms (experiment A2) using Criteria 3 and 4 at threshold $\gamma = 0.5$, gave an improvement of about 22% speech recognition accuracy compared to VA performance and about 19% improvement compared to experiment A1. We also see that as the SNR improves, the gap in the speech recognition accuracy between Criteria 1 ($\gamma = \infty$) and Criteria 2 ($\gamma < 0$) reduces. In fact as SNR approaches to that of clean speech, Criteria 1 is better than Criteria 2. We see that for clean speech, the speech recognition accuracy improved by more than 2%. As expected, the product operation worked better than max operation for clean speech (see section 4).

In a real time spoken dialogue ASR system, the system uses the standard VA for decoding speech. If the ASR system has a confusion whether it has succeeded in recognizing a spoken word in a sentence, then it asks the user to repeat that word. The ASR system can decide whether or not it has recognized the word correctly based on some confidence measure. If the final probability of word match is too low, or if the difference in the final probabilities of the best and the second best matched word is too low, then the system can ask the user to repeat the word. And that is where the algorithms we proposed can be used to improve speech recognition performance. In a real working ASR system, if threshold γ cannot be found, we can go for Criteria 2 as it gives near optimal performance.

An example from the experiments is given as follows. In the experiments for noisy speech at -5 dB SNR (10% burst noise), the word "Oh" was commonly mismatched with words "Home" and "Four", for many speakers, when we used the standard VA. But by using the proposed algorithms with Criteria 2, 3 and 4, for most of the cases, this mismatch was removed and the word "Oh" was correctly recognized.

6. CONCLUSIONS

The problem of jointly decoding multiple speech patterns was addressed. We proposed a hybrid approach comprising of both the non parametric and parametric approaches to speech recognition to solve this problem. We proposed two novel algorithms - the MPDTW and CMPVA, and applied them sequentially, to jointly decode multiple patterns of speech. We got a huge improvement in speech recognition accuracy for noisy speech. We also got an improvement in speech recognition accuracy for clean speech.

7. REFERENCES

[1] Viterbi, A., "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", IEEE Trans. Inf.Theory,vol.IT-13,no.2,pp. 260-269, April 1967.

[2] A Bernard, A Alwan, "Joint channel decoding-Viterbi recognition for wireless applications - all 5 versions", Proc. Eur. Conf. Speech comm. Tech., pp. 2703-2216, 2001.

[3] Siu, Manhung and Chan, Arthur, "A Robust Viterbi Algorithm Against Impulsive Noise With Application to Speech Recognition", IEEE Trans. Audio, Speech and Language Proc., pp. 2122- 2133, Nov. 2006.

[4] Schwartz, R, Chow, Y.-L, "The N-best algorithms: an efficient and exact procedure for finding the N most likely sentence hypotheses", ICASSP-90., pp. 81 - 84 vol.1, April 1990.

[5] Wu,J, Gupta,V, "Application of simultaneous decoding algorithms to automatic transcription of known and unknown words", ICASSP '99. Proceedings., pp. 589 - 592, Vol. 2, March 1999.

[6] Jiayu Li, Murua, A.,"A 2D extended HMM for speech recognition", ICASSP '99. pp: 349 - 352, vol.1, March 1999.

[7] Lleida,E., Rose R.C., "Utterance verification in continuous speech recognition: decoding and training procedures", IEEE Trans. on Speech and Audio Proc., Vol. 8, Issue: 2, pp: 126-139, Mar 2000.

[8] Hansheng Lei; Palla, S.; Govindaraju, V., "ER^2: an intuitive similarity measure for on-line signature verification", IWFHR-9 2004, pp:191 - 195, Oct. 2004.

[9] Sang-Wook Kim, Sanghyun Park and Wesley W. Chu, "Efficient processing of similarity search under time warping in sequence databases: an index-based approach",Information Systems, Vol. 29, Issue 5, pp: 405-420, July 2004.

[10] Yegnanarayana, B.; Anil Kumar sao; Kumar, B.V.K.V.; Savvides, M., "Determination of pose angle of face using dynamic space warping", ITCC 2004. International Conference on, pp:661 - 664 Vol.1, 2004.

[11] M.H. Ko, G. West, S. Venkatesh, M. Kumar, "Temporal data fusion in multisensor systems using dynamic time warping", Workshop on Information Fusion and Dissemination in Wireless Sensor Networks 2005.

[12] Rabiner, L.R. and Juang, B-H., "Fundamentals of Speech Recognition", Pearson Education Inc, pp. 240 - 262, 1993.