Automatic Labeling and Digesting for Lecture Speech Utilizing Repeated Speech by Shift CDP

Yoshiaki Itoh[†] *and Kazuyo Tanaka*[‡]

[†] Iwate Prefectural University
[‡] Electrotechnical Laboratory
y-itoh@iwate-pu.ac.jp

Abstract

This paper proposes an automatic labeling and digesting method for lecture speech. The method utilizes same sections, such as same words or same phrases that are thought to be important and are repeated in the speech. To extract the same sections, we have proposed a new efficient algorithm, called Shift Continuous DP, because it is an extension of Continuous DP and realizes fast matching between arbitrary sections in two speech data sets frame-synchronously. Shift CDP is extended to extract same sections in single long speech data in this paper. This paper describes ways to apply the algorithm to labeling and digesting for a lecture speech. We conduct some preliminary experiments to show the method can extract same sections and a sequence of extracted sections can be regarded as a digest of the speech.

1. Introduction

With the development of the multi-media environment and the Internet, and the increase of speech and motion images data, a method to handle such data is needed more and more. There are, however, usually no text labels adhering to such data. For this purpose, an approach for recognizing time sequence data is thought to be one representative way. This approach gives labels or text to the data by recognizing all the data using models prepared in advance. Performance depends on the speech recognizer and miss recognition is inevitable because of the difficulty of correctly recognizing, especially real-world speech data, such as lectures or conversational speech. Thus, a new approach for labeling or digesting speech or motion images is proposed in this paper.

Generally speaking, there is redundancy in a speech, such as in a presentation or lecture, and the important words and phrases are, therefore, assumed to be repeated in the speech. This paper describes the methodology for extracting such repeated speech sections in the whole speech. Labeling and digesting the speech is also described by adhering some labels to those sections and offering an intensive sequence of labeled sections. We have proposed a new matching algorithm called Shift Continuous DP (Shift CDP) [1], [2], [3] that efficiently extracts the same sections between two time sequence data sets by performing fast spotting between arbitrary sections of time sequence data and arbitrary sections of another time sequence data. In this paper, Shift CDP is extended to extract the same sections from the whole time sequence data. The algorithm can be regarded as extending Incremental RIFCDP [4] that extracts the same sections in a limited time interval, such as in the most recent 30 seconds.

The concept and the algorithm of Shift CDP are explained briefly at first, and then the methodologies of extending Shift CDP to extract same sections from single long time sequence data. We applied the method to presentation speech data. The conditions and the results of the preliminary experiments for extracting same sections and digesting of the speech are shown in the third section.

2. Extension of Shift CDP

Shift Continuous DP (Shift CDP) is an improved version of Reference Interval Free Continuous DP (RIFCDP) [2] that performs matching between arbitrary sections of the database and arbitrary sections of query input. A brief description of the concept and the algorithm are described first, and then the method to extend the algorithm to extract same sections in single long time sequence data.

2.1. The Concept of Shift Continuous DP (Shift CDP)

The reference pattern R, that is considered as a database, and input pattern sequence I are expressed by Eq. (1) below, where $R\tau$ and I_t both indicate a member of a feature parameter series at the frame τ and time t respectively.

$$R = \{ R_1, \cdots, R\tau, \cdots, R_N \}$$
$$I = \{ I_1, \cdots, I_t, \cdots, I_\infty \} \qquad (1)$$

Shift CDP is an algorithm that spots similar sections between reference pattern R and the input pattern sequence I synchronously with input frames. The input pattern sequence is assumed to continue infinitely, as indicated in the above equation. Here, a similar section (Rs, Is) would lie between the two coordinate points (τ_1, t_1) and (τ_2, t_2).

In the algorithm to solve the above problem, all the matching should be done between (τ_1, t_1) and (τ_2, t_{now}) at each input, where $1 \le \tau_1 \le \tau_2 \le N$, $1 \le t_1 \le t_{now}$. Let the minimum and maximum length for Rs be N_{min} and N_{max} respectively, so $N_{min} \le \tau_2 - \tau_1 \le N_{max}$. These constraints give the desired length to be detected and also reduce the calculation burden. To perform optimal matching for the length from N_{min} to N_{max} at frame τ_2, this frame is assumed to be the end frame of CDP and CDP is performed for the N_{max}-N_{min} pattern whose mean length is $(N_{min}+N_{max})/2$. Thus, C

DP has to be done about $(N_{max}-N_{min}) \times N$ times for the reference pattern and its calculation burden is expected to be heavy even if such constraints for matching length are given.

Shift CDP can reduce the calculation burden described above. The concept behind this algorithm is shown in Figure 1. First, unit reference patterns are taken from reference pattern R. A unit reference pattern (URP), has a constant frame length of N_{URP}. The first URP is composed of frames from the first frame to the N_{URP}-th frame in R. The starting frame of the second URP is shifted by N_{shift} frames and the second URP is composed of the same number of N_{URP} frames, from the N_{shift} +1-th frame. In the same way, the k-th URP is composed of N_{URP} frames from the k×N_{Shift}+1-th frame. The last URP is composed of N_{URP} frames from the last frame of R toward the head of R. The number of URPs becomes N_{PAT}=[N/ N_{Shift}] +1 where [] indicates the integer that does not exceed the value.

For each URP, CDP is performed. It is not necessary to normalize each cumulative distance at the end frame of a URP because the length of all the URPs is the same. As described above, Shift CDP is a very simple and flat algorithm that just performs CDP for each URP and integrates the results.

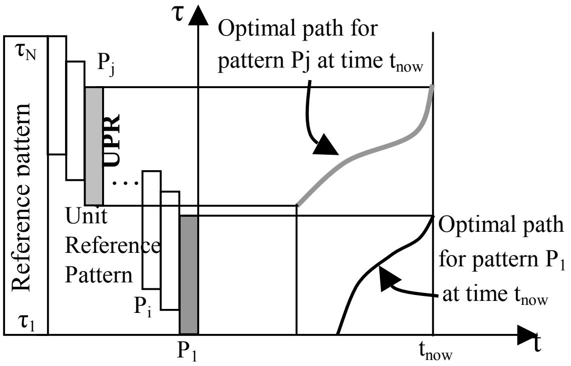
τ
Optimal path for pattern Pj at time t_{now}
$τ_N$
P_j
Reference pattern
UPR
. . .
Unit Reference Pattern
P_i
$τ_1$
P_1
t_{now}
t
Optimal path for pattern P_1 at time t_{now}

Figure 1: The concept of Shift CDP algorithm.

2.2. Shift CDP Algorithm Outline

We let the vertical axis represent the reference pattern frame τ(1≤τ≤N), and the horizontal axis as input time t. The local distance between the input t and frame τ is denoted as $D_t(τ)$. Here, asymmetric local restrictions are used used as the DP path, as shown in Figure 2.

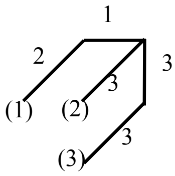

Figure 2. DP local restrictions

Let $G_t(i, j)$, $G_{t-1}(i, j)$ and $G_{t-2}(i, j)$ denote the cumulative distance up to frame j in the i-th URP at input time t, t-1 and t-2 respectively. Input time t is the current time and t-1 is the previous time. In the same way, $S_t(i, j)$, $S_{t-1}(i, j)$ and $S_{t-2}(i, j)$ denote the starting time. Let $τ_S(i)$ and $τ_E(i)$ be the frames of R that correspond to the starting and ending frames of the i-th URP. These locations should be calculated before input. To reduce the total calculation, $D3_t(τ)$ and $D2_{t-1}(τ)$ are calculated and saved apart from $D_t(τ)$ according to the DP path restrictions, as shown in Figure 2. $D3_t(τ)$ and $D2_{t-1}(τ)$ denote three times for $D_t(τ)$ and two times for $D_{t-1}(τ)$ respectively.

LOOP i (1≤ i ≤N_{PAT}): for each URP i,

LOOP j (1≤ j ≤N_{URP}): for each frame j of URP i,

$$\text{at } j=1,\quad \begin{cases} G_t(i, 1) = D3_t(τ_S(i)\,) \\ S_t(i, 1) = t \end{cases} \tag{2}$$

$$\text{at } j\geq 2,\quad \begin{cases} P(1) = G_{t-2}(i, j\text{-}1) + D2_{t-1}(τ_S(i) +j\text{-}1) \\ \qquad\qquad + D_t(τ_S(i) +j\text{-}1) \\ P(2) = G_{t-1}(i, j\text{-}1) + D3_{t-1}(τ_S(i) +j\text{-}1) \\ P(3) = G_{t-1}(i, j\text{-}2) + D3_t(τ_S(i) +j\text{-}2) \\ \qquad\qquad + D3_t(τ_S(i) +j\text{-}1) \\ \text{but at } τ=2, \\ \quad P(2) = D3_t(τ_S(i)) + D3_t(τ_S(i)+1) \end{cases} \tag{3}$$

$$α^* = arg\ min_{(α=1,\ 2,\ 3)}\ P(α) \tag{4}$$

$$G_t(i, j) = P(α^*) \tag{5}$$

$$S_t(i, j) = \begin{cases} S_{t-2}(i, j\text{-}1) & (α^*=1) \\ S_t(i, j\text{-}1) & (α^*=2) \\ S_{t-1}(i, j\text{-}2) & (α^*=3) \end{cases} \tag{6}$$
$$\text{but at } j=2 \text{ and } α^*=3,\ \ S_t(i, 2) = t$$

End LOOP j, if j=N_{CDP}. : end CDP for the i-th URP.

End LOOP i, if i=N_{PAT}.: end the process for the current time t

The three terms of P in Eq. (3) represent the three start points of the path restrictions shown in Figure 2. An optimal path is determined according to Eq. (4). The cumulative distance and the starting point are updated by Eqs. (5) and (6) using $α^*$. The adjustment degree A(t, i) of the i-th URP at time t is given by the cumulative distance at the last frame without normalization by the length of reference patterns because the length of all URPs is equal.

$$A(t, i) = G_t\ (i, N_{URP}) \tag{7}$$

After determining similar sections at time t, the cumulative distances, starting points and local distances are renewed[1]. This procedure is just renewing the index of arrangements in the actual program that produce no calculation burden.

There are ways to determine similar sections according to the application purpose [2]. For example,

1. Detect the most similar section,
2. Detect any similar sections,

in the reference and the given input. In this paper, a threshold value is set and all the sections are detected when the adjustment degree exceeds the threshold value because there might be plural similar sections in the reference pattern for input query.

Experiments were performed to evaluate the performance of the algorithm in detecting similar sections, using conversational speech data taken from the speech database of the Acoustical Society of Japan. The performance was compared to RIFCDP mentioned above, and the performance of Shift CDP is comparable with RIFCDP [2]. On the other hand, Shift CDP could reduce calculation resources remarkably. For example, Shift CDP requires 1/15 the processing time and 1/75 the memory use of RIFCDP at 25 frame shifts.

2.3. Same Speech Extraction from Lecture Speech by Extending Shift CDP

Shift CDP is a frame-synchronous algorithm. This section describes a method for extending Shift CDP to extract same sections from a long speech maintaining this characteristic. This image is illustrated in Figure 3. URPs are constructed according to the progress of input. After N_{URP} frames input, the first URP is composed of the first N_{URP} frames. With the progress on N_{shift} frames input furthermore, the second URP is composed of the same number of N_{URP} frames, from the N_{shift} +1-th frame. In the same way, the k-th URP is composed of N_{URP} frames after the {N_{URP} + (k-1)×N_{Shift} } frames input. The last URP is composed of N_{URP} frames from the last frame of R toward the head of R. As soon as a URP is composed, CDP is performed for the URP, shown in the left figure of Figure 4

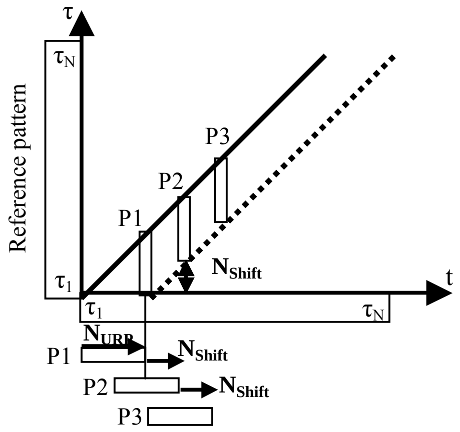

Figure 3 : Image for extracting same sections from a lecture.

and the search area becomes the bottom triangle that is the gray area in the right figure of Figure 4.

The following formalizes the algorithm, mentioned above, for extracting same sections from a long speech, using Shift CDP.

[Algorithm for extracting same sections in a long speech using Shift CDP]

```
Initial Conditions
  URP_NUM = 0
  Waiting_Frame = $N_{Shift}$- $N_{URP}$
LOOP1:  for each Input  $I_k(1 \le k \le N)$
  $R_k = I_k$
  Waiting_Frame++
  If (Waiting_Frame = $N_{Shift}$)
    URP_NUM++,  Waiting_Frame = 0
  LOOP2: for each frame $R_j$ $(1 \le j \le k$- Waiting_Frame)
    Local Distance Calculation D(Rk, Rj)
  LOOP2 End
  LOOP3: for each URP$_i$ $(1 \le i \le$ URP_NUM)
    Perform CDP and get G(i,k): Distance of URP$_i$ at input k
    If (URP$_i$ detects a same section) Output
      Distance of URP$_i$: G(i,k)
   Same sections: (S(i,k), k) and (S$_i$, E$_i$)
   Integrates the result until input k-1
  LOOP3 End
LOOP1 End
```

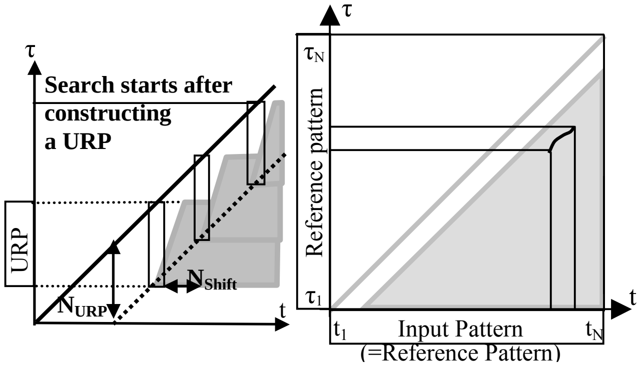

Figure 4: Search Area

In the algorithm, a Waiting_frame is introduced to judge when to construct a new URP. The timing is described in the explanation of Figure 3. In LOOP3, there is a judgment whether URP$_i$ detects a same section at input time k, described in section 2.2. When URP$_i$ detects a same section, the corresponding two sections are output. S$_i$ and E$_i$ in the algorithm indicate the start and the end frame of URP$_i$ and those newly detected sections are integrated with the same sections that have been detected until the previous input time.

3. Experiments Using Lecture Speech

3.1. Experimental Conditions

To evaluate the algorithm mentioned above, some preliminary experiments were conducted by applying it to a long speech. Here, we used a speech for an explanation of a Shift CDP algorithm that was made for a presentation exercise. The presentation was recorded with a head set microphone in a laboratory and took ten minutes (It is not a read text speech). The condition of speech analysis is shown in table 1. The amount of output can be controlled according to the threshold valued described in section 2.2.

Table 1 Conditions of Experiments

Sampling frequency	16 kHz
Frame interval	8 msec.
Feature vector	a 36-dimentional graduated spectrum field analyzed with a 20-channel filter bank [5]
N_{URP}	90 frames (720 ms)
N_{Shift}	30 frames (240 ms)
N	75000 frames(10minutes)
Language	Japanese

3.2. Results and Discussion

First, the calculation time was actually measured when extracting same sections from a ten minute speech. It took about 150 minutes with a SUN Microsystems' workstation (Ultra 5, 360 MHz). It is thought to be impossible for RIFCDP or IRIFCDP to deal with such a long speech because it requires too much memory use.

We used the threshold processing of matching distance G, described in section 2.2, to judge same sections. For a ten minute speech, the threshold value was controlled so that two one minute same sections are output. The details of the output are shown below.

- composed of 69 same sections pair
- totally 121 seconds long
- 59 seconds on the reference side, 62 seconds on the input side

When we listened to the output pairs, all the pairs showed almost the same phoneme sequence and there were no pairs that showed distinctly different phoneme sequences. This tendency was same even when twice the output was produced by loosening the threshold value.

Figure 5 shows a sample of the output in the order the system produced. The first two numbers indicate the start and the end frame number in the reference pattern and the length usually show the URP length, here 90 frames. The next two

Reference			Input		Speech
630	719	::	5029	5122	(RI)FCDP
5040	5129	::	5751	5833	(RI) FCDP
5700	5789	::	14655	14763	Well, RIFCD(P)
5010	5129	::	14674	14828	RIFCDP
14730	14849	::	15420	15510	CDP method
5070	5189	::	15431	15539	called Shift CD(P)
14730	14849	::	15941	16047	CDP method
15420	15509	::	15941	16036	CDP method
18840	18929	::	19196	19293	Well…
18780	18869	::	23434	23528	and well…
16650	16739	::	23590	23695	reference pattern
20580	20669	::	23607	23713	reference pattern
17880	17969	::	24994	25096	against the
15900	15989	::	27242	27334	Shift CDP
23160	23279	::	27548	27663	unit reference pa(ttern)
23430	23519	::	28417	28530	and well…
23130	23249	::	28701	28802	unit reference
27570	27689	::	28751	28848	reference pattern
12570	12659	::	29212	29294	(pause)
0	89	::	30005	30093	(pause)
21360	21449	::	40182	40268	Shift CD
30270	30359	::	40210	40285	(Shi)ft CDP

•
•
• 1 frame = 8mm second

Figure 5. Detected same sections

numbers indicate the start and end frame number in the input and show each detected input section has a different length. The column "Speech" was manually labeled after listening to the output and translated from the Japanese. The words in a parenthesis are for reference and are not detected.

From the 121 seconds output, 51 seconds output were pause sections or unnecessary words, such as "well," "and", "according to" and so on. The other 70 seconds includes all or a part of a "content word" such as "RIFCDP, Shift CDP, 15 times faster..."

When the two sections are extracted as the same speech, as shown in Figure 5, a label that indicates this is attached to those sections. Some labels are shared with many sections where there is a high possibility of important speech.

No.	Pair	Start	End	Length	Dup.	Speech
1	6	14655	14849	195	18	RIFCDP method
2	11	18780	18929	150	8	and well…
3	18	23590	23712	123	2	reference pattern
4	19	24994	25094	101	1	against
5	30	41040	41279	240	16	of a unit reference pattern
6	46	60630	60749	120	1	high detection perfor(mance)
7	50	62512	62674	163	11	well, if we
8	55	65550	65639	90	1	(Euc)rid distance calcu(lation)
9	60	68391	68513	123	1	Shift number 25
10	62	68807	68911	105	1	1.6 times
11	66	71004	71157	154	2	well…
12	69	73374	73486	113	2	speech data(base)

Figure 6. A sample digest of a presentation
(Dup. means the number of related sections)

Figure 6 shows a sample result of integrating the result shown in Figure 5 by just putting together chaining pairs to a representative single section.

Although there are many unnecessary speech sections, shown in Figure 5, it is possible to understand what kind of speech, at what time and how often they were made in the presentation by just listening to the digested output in Figure 6. The sequence of extracted pairs can be regarded as a digest of the presentation.

4. Conclusions

This paper proposes a new approach for labeling and digesting non-segmented and recognized speech data. Shift CDP, which was developed to extract same sections efficiently between two time sequence data, was extended to enable extracting repeated words or phrases in a long single speech, such as a lecture. The algorithm was applied to a presentation speech, and the result of the experiment showed the algorithm was able to extract the same speech sections correctly and to give those extracted sections the same labels that indicate repeated words or phrases. The sequence of extracted pairs, therefore, could be regarded as a digest of the presentation. Currently, the method is limited to single speaker's speech, such as a lecture, and we are extending it to a "speaker and language independent method" that allows for discussions by utilizing acoustic models of a language-independent phonetic code [5]. As there are still many unnecessary speech sections, a method to select good pairs that characterize speech is an important task for the future.

References

[1] Y. Itoh, "A Matching Algorithm Between Arbitrary Sections of Two Speech Data for Speech Retrieval by Speech," ICASSP, May 2001.

[2] Y. Itoh, J. Kiyama and R. Oka, "A Proposal for a New Algorithm of Reference Interval-free Continuous DP for Real-time Speech or Text Retrieval," ICASSP, vol.1, pp.486-489, Oct. 1996.

[3] H. Kojima, Y. Itoh and R.Oka, "Location Identification of a Mobile Robot by Applying Reference Interval-free Continuous Dynamic Programming to Time-varying Images," Third International Symposium on Intelligent Robotics Systems, Nov. 1995.

[4] J.Kiyama, Y.Itoh and R.Oka, "Automatic Detection of Topic Boundaries and Keywords in Arbitrary Speech Using Incremental Reference Interval-free Continuous DP," ICSLP, vol.3, pp.1946-1949, Oct 1996.

[5] K. Tanaka, H. Kojima, "Speech recognition method with a language-independent intermediate phonetic code," ICSLP2000, Vo.IV, pp.191-194, 2000.

[6] G.A.Smith, H.Murase, K.Kashino, "Quick Audio Retrieval using Active search," ICASSP, vol.6, pp.3777-3780, May 1998.

[7] R.C.Rose, E.I.Cang and R.P.Lippmann: "Techniques for Information Retrieval from Voice Messages," ICASSP, Vol.I, pp.317-320, Apr.1991.

[8] F.Chen and M. Withgott, "The use of emphasis to automatically summarize a spoken discourse," ICASSP, vol.I, pp.229-232,May 1992.