Application of Recognition Techniques for Mandarin Syllables to German Alphabet Recognition

[♦]Hansjörg Mixdorff, [♣]Yuan-Fu Liao and [♣]Sin-Horng Chen

[♦]Institut für Technische Akustik und Sprachkommunikation, TU Dresden,
Mommsenstraße 13, 01062 Dresden
E-mail: h.mixdorff@teles.de

[♣]Department of Communication Engineering, National Chiao Tung University,
1001 Ta Hsueh Rd., Hsinchu 300, Taiwan, Republic of China
E-mail: u8213803@cc.nctu.edu.tw

Abstract: Robust speaker-independent alphabet recognition over a telephone line is a task yet unsolved. The current study examines the feasibility of Modular Recurrent Neural Networks (MRNNs) successfully applied to Mandarin syllable recognition to the recognition of German letters. Letters were divided into sub-word units for each of which specialized RNNs were trained. The results presented in this paper show that, within the framework of MRNNs, the highest recognition rates can be achieved by segmenting letters into right context-dependent initials and context-independent finals. However, these rates are lower than those achieved with comparable HMMs, primarily due to the higher robustness against segmentation inaccuracies of the latter approach.

1. Introduction

In quite a number of applications of speech recognition, such as automated telephone directories, alphabet recognition is required. This task, however, is far from being solved due to the similarities between the phonetic representations of German letters. These fall into a few sets:

(1) the [a:]-set: A [?a:], H [ha:], K [ka:]
(2) the [e:]-set: B [be:], C [tse:], D [de:], E [?e:], G [ge:], P [pe:], T[te:], W [ve:]
(3) the [u:]-set: Q [ku:], U [?u:]
(4) the [E]-set: F [?Ef], L [?El], M [?Em], N [?En], R [?ER], S [?Es]
(5) Others: I [?i:], J [jOt], O [?o:], V [faU], X [Iks], Y [?YpsIlOn], Z [tsEt], Ä [?E:], Ö,[?2:] Ü,[?y:], ß [?EstsEt]

Considering the statistical distribution of German letters, over 80% of letters in an average text fall into the highly confusable groups (1), (2) and (4). The five sets can be further grouped into three broad classes, i.e. the (C)+V (set 1, 2 and 3), V+(C) (set 4) and others (set 5).
Structurally, the letter recognition task resembles the Mandarin syllable recognition task. Phonetically, many of the syllables of Mandarin exhibit (C)+V or V+(C) structures, which are, except for lexical tone, only distinguished by the initial or final consonant. Hence, the current study examines whether recognition strategies for the highly confusable Mandarin Chinese syllables can be successfully applied to letter recognition, with special emphasis on the application of Modular Recurrent Neural Networks (MRNN)[1]. It has been shown in earlier works on German that Modular Neural Networks can be useful for subdividing specific recognition tasks [2].

2. Database, Recognition Features and Modeling Units

The database used in the current study is the German Telekom “Zifkom” corpus containing recordings of all German letters uttered in isolation once by 100 male and 100 female speakers. The data was down-sampled to 8 kHz to simulate telephone conditions. 12 melcepstral coefficients (MFCCs), 12 delta-MFCCs and 12 delta-delta-MFCCs, one delta-energy and one delta-delta-energy factor were determined for frames of 32 ms at a delay of 10 ms. 80% (160 speakers, 4800 utterances) and 20% (40 speakers, 1200 utterances) of the data was used for training and testing, respectively.

Table 1. Table showing the sub-word unit coding for (b) 12 context-independent (CI) -initial and 20 CI-final sub-word models, (c) 22 right-final-dependent (RCD) initial and 20 CI final sub-word models and (d) 12 CI-initial, 13 CI-vowel and 11 CI-final sub-word models, and broad class-assignment.

letter	SAMPA	sub-word unit codes (b)		sub-word unit codes (c)		sub-word unit codes (d)			broad class
		initial	final	initial	final	initial	vowel	final	
A	[?a:]	1	1	1	1	1	1	1	1
B	[be:]	4	2	2	2	4	2	2	1
C	[tse:]	3	2	3	2	3	2	1	1
D	[de:]	5	2	4	2	5	2	1	1
E	[?e:]	1	2	5	2	1	2	1	1
F	[?Ef]	1	4	6	4	1	4	3	2
G	[ge:]	6	2	7	2	6	2	1	1
H	[ha:]	7	1	8	1	7	1	1	1
I	[?i:]	1	5	9	5	1	5	1	3
J	[yOt]	8	6	10	6	8	6	2	3
K	[ka:]	2	1	11	1	2	1	1	1
L	[?El]	1	7	6	7	1	4	4	2
M	[?Em]	1	8	6	8	1	4	5	2
N	[?En]	1	9	6	9	1	4	6	2
O	[?o:]	11	10	12	10	1	7	1	3
P	[pe:]	9	2	13	2	9	2	1	1
Q	[ku:]	2	3	14	3	2	3	1	1
R	[?ER]	1	11	6	11	1	4	7	2
S	[?Es]	1	12	6	12	1	4	8	2
T	[te:]	12	2	15	2	12	2	1	1
U	[?u:]	1	3	16	3	1	3	1	1
V	[faU]	10	13	17	13	10	8	1	3
W	[ve:]	11	2	18	2	11	2	1	1
X	[?Iks]	1	14	19	14	1	9	9	3
Y	[?IpsilOn]	1	15	19	15	1	10	10	3
Z	[tsEt]	3	16	20	16	3	4	2	3
Ä	[?E:]	1	17	6	17	1	11	1	3
Ö	[?2:]	1	18	21	18	1	12	1	3
Ü	[?y:]	1	19	22	19	1	13	1	3
ß	[?EstsEt]	1	20	6	20	1	4	11	3

Four sets of modeling units were tested. They include (a) 30 full-word-models, (b) 12 context-independent (CI)-initial and 20 CI-final sub-word models, (c) 22 right-final-dependent (RCD)-initial and 20 CI-final sub-word models and (d) 12 CI-initial, 13 CI-vowel and 11 CI-final sub-word models. The inventories of the latter three sets are listed in Table 1.

3. Simulations

3.1 Single RNN

As a baseline system, a single recurrent neural network (RNN) was trained with full-word samples. The RNN consists of three layers, an input layer, a hidden layer and an output layer. The outputs from the hidden layer are fed back to the input layer as additional inputs (see Figure 1). Different numbers of hidden layer nodes were tested. A number of 90 hidden nodes proved to be optimal (see Table 2).

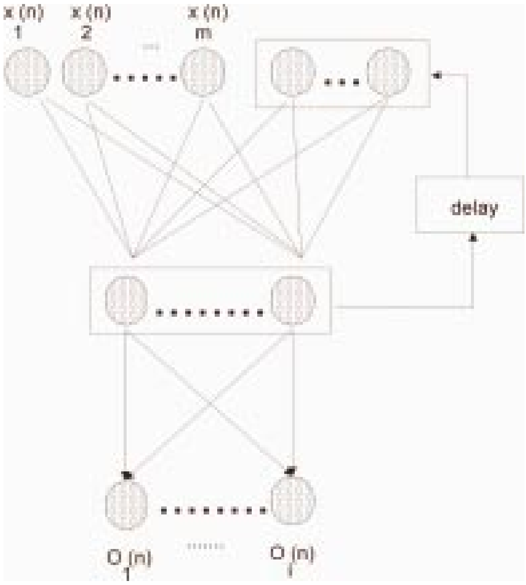

Figure 1. The structure of a Recurrent Neural Network (RNN).

Table 2. Overall recognition rates, single RNN.

Number of hidden nodes	30	90
Inside recognition rate (%)	96.16	98.18
Outside recognition rate (%)	86.78	91.38

Table 3. Letter-dependent outside recognition rate for a single RNN (90 hidden nodes), relative letter frequency and confused counterparts.

Letter	Rate (%)	Relative frequency (%)	mostly confused with
A	85.00	6.94	H
B	82.50	1.61	D
C	92.50	2.98	D
D	90.00	6.38	B
E	80.00	16.83	G, P
F	82.50	0.91	S
G	77.50	2.76	D, T, E
H	92.50	4.22	K
I	92.31	7.34	
J	97.50	0.33	
K	95.00	0.83	H, A
L	97.50	4.03	Ö, R
M	85.00	2.01	N, L
N	85.00	10.11	M, L
O	92.50	2.47	
P	95.00	0.78	B
Q	100.00	0.06	
R	90.00	7.27	L
S	77.50	7.55	F, L
T	92.50	6.26	C, P
U	95.00	4.93	Q
V	95.00	0.92	
W	90.00	1.32	B, D
X	100.00	0.09	
Y	100.00	<0.01	
Z	95.00	1.08	
Ä	92.50	<0.01	R, F
Ö	97.44	<0.01	
Ü	94.88	<0.01	Ö
ß	100.00	<0.01	

One must, however, consider, that recognition performance varies depending on the particular letter. Table 3 displays the outside recognition rate for all letters for the case of 90 hidden nodes. The second column from the right lists the relative frequency of a particular letter taken from [3].

The right column of Table 3 lists counterparts with which a particular letter is likely to be confused. It becomes obvious, that letters belonging to the same set are most prone to confusion. Confusion concerns place of articulation (labial voiced stop B and alveolar voiced stop D, for instance), manner of articulation (voiceless fricative H and voiceless stop K) and presence or absence of voicing (voiceless labial stop P and voiced labial stop B). These cases of confusion roughly correspond to those observed with humans listeners.

3.2 HMM-based approaches

The application of Modular Neural Networks (MRNNs) requires the segmentation of every letter into sub-word units, i.e. initial and final parts, for instance. This segmentation is achieved by training HMMs with topologies corresponding to the sub-word-modeling units chosen. Besides, the HMMs created in this process can be used as a reference for the performance of the resulting MRNNs.

According to the four sets of modeling units, four HMM topologies using simple left-to-right state diagrams are tested. They include (a) 30 8-state word models, (b) 12 3-state-initial and 20 5-state-final sub-word models, (c) 22 3-state-initial and 20 5-state-final sub-word models, and (d) 12 2-state-initial, 13 4-state-vowel and 11 2-state-final sub-word models, respectively. In addition, two silence states are used to model and segment the silence parts before and after each letter.

In order to determine sub-word boundaries, minimum distortion segmentation [4] is used to guess possible segmentation positions for each training syllable. A vector quantization procedure is applied to generate the first sets of initial and final HMM models according to the segmentation information. Then the HMM models are concatenated to form the first set of 30 syllable- (8 state-) HMM models. They are further fine-tuned by the segmental K-mean (re-segment-then-re-estimate) algorithms based on the maximum likelihood (ML) criterion.

Finally, those well-trained syllable HMM models are used to give more accurate segmentation positions for each training syllable, especially the segment boundary location between initial and final part. Using this segmentation information separate RNNs for initial and final part of the letter can be trained.

Following the results of a preliminary simulation, a mixture Gaussian distribution function with 15 Gaussian components was chosen for all HMM models. The simulation results are listed in Table 4. The best Top-1 and Top-5 recognition rates achieved are 94.2% and 100.0% using 22 RCD initial and 20 CI final HMMs. In a second step, the MCE/GPD algorithms were used to fine-tuned the above ML-trained HMMs. The simulation results are listed in Table 5. Although the Top-1 recognition rates are almost the same as with ML-trained HMMs, the Top-N recognition rates are improved.

Table 4. The alphabet recognition results of the four HMM-based approaches using the ML training algorithms.

HMM topology	Recognition rate (%)				
	Top-1	Top-2	Top-3	Top-4	Top-5
12 initials/13 vowel/11 finals	86.7	94.8	99.0	99.6	99.6
12 initials/20 finals	91.8	97.8	99.6	99.7	99.8
22 initials/20 finals	94.2	99.2	99.9	99.9	100.0
30 words	92.5	97.9	99.4	99.5	99.7

Table 5. The alphabet recognition results of the three best HMM-based approaches using the MCE/GPD training algorithms.

HMM topology	Recognition rate (%)				
	Top-1	Top-2	Top-3	Top-4	Top-5
12 initials/20 finals	90.8	98.1	99.7	99.7	99.9
22 initials/20 finals	94.2	99.2	100.0	100.0	100.0
30 words	92.6	98.0	99.4	99.6	99.8

3.3 MRNN-Based Approaches

3.3.1 MRNN with 22 RCD Initials and 20 CI Finals

According to the results of the HMM method, an MRNN-structure using 22 RCD-initials and 20 CI-finals like the best HMM topology was created. Figure 2 shows the block diagram of the MRNN. The MRNN has three constitute RNNs, i.e. the initial, final and weighting RNN. The initial and final RNNs are used to classify initials, finals and generate 42 partial discriminant functions. The weighting RNN is used to generate three dynamic weighting functions for initial, final and silence segments. The function of the latter is to suppress the silence parts before and after each utterance. In the Discriminant Function Accumulator the 42 partial discriminant functions are integrated into 30 letter discriminant functions.

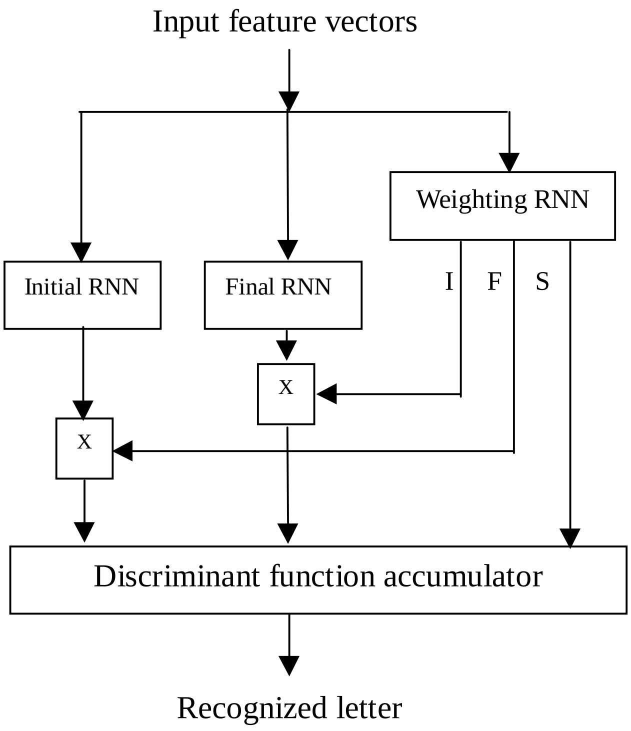

Figure 2: Block diagram of MRNN with 22 RCD Initials and 20 CI Finals.

In the testing phase, a discriminant function is defined for each letter. For the *p*-th letter, which is composed of the *i*-th RCD-initial, and the *j*-th final, the discriminant function can be expressed as

$$g_p(X_0^{L-1}) = \frac{1}{L}\sum_{n=0}^{L-1}\begin{Bmatrix} O_{I_C}^{W}(X_n)\cdot O_i^{I}(X_n) \\ +O_{F_C}^{W}(X_n)\cdot O_j^{F}(X_n) \end{Bmatrix}, \quad p = 0 \sim 29 \qquad \text{Equation (1)}$$

where X_0^{L-1} is the feature-vector sequence of a test utterance of length *L*; $O_i^I(X_n), O_j^F(X_n)$ are, respectively, the *i*-th output of the initial RNN, and the *j*-th output of the final RNN; $O_{I_C}^W(X_n), O_{F_C}^W(X_n)$ are the corresponding initial and final dynamic weighting functions produced by the weighting RNN, where *c* (*c*=1 or *c*=1~3) is the broad class of the letter. The final decision rule then chooses the best candidate letter according to the maximum discriminant function.

In the training phase, each input utterance is first segmented into initial/final/silence parts using the HMM method. The initial and final RNNs are thus independently trained using the sub-word-level

MCE/GPD algorithms according to the segmentation positions given by the HMM method. The weighting RNN is also independently trained by the conventional back-propagation through time (BPTT) algorithms using the same segmentation positions and the “0-1” target function. After the three RNNs are well trained, they are combined into the MRNN alphabet recognizer and further fine-tuned by the word-level MCE/GPD algorithms according to the discrimant function defined in Equation 1.

Several sets of the segmentation positions generated by the different HMM topologies are tested. The purpose is to find a more consistent and suitable initial/final segmentation scheme, as many HMMs, though consisting of one initial and one final model, do not share their sub-word models. They are essentially word models. The HMMs thus tend to find an optimal word model but not optimal initial/final segmentation. In all tests, 90 hidden neurons are used in initial and final RNNs, and 30 hidden neurons in the weighting RNN. According to different segmentation schemes, several sets of the initial/final and letter recognition rates achieved are listed in Table 6. As can be seen the results are poorer than those from the HMM methods.

The behavior of the MRNN is illustrated in Figure 4 on the example of the letter W [ve:]. The figure displays from top to bottom: The speech wave form, the spectrum, frame energy and zero crossing rate, the output of the weighting RNN, initial and final RNNs. It can be seen that the response of the initial RNN reaches its maximum before the inter-phone boundary, whereas the final RNN is activated shortly after.

Table 6. The partial recognition results of the initial and final RNNs, and the letter recognition result of the MRNN.

Segmentation schemes	Recognition rates (%)		
	22 initials	20 finals	30 letters
12 initials/20 finals	92.1	93.7	91.0
22 initials/20 finals	95.1	93.2	92.5

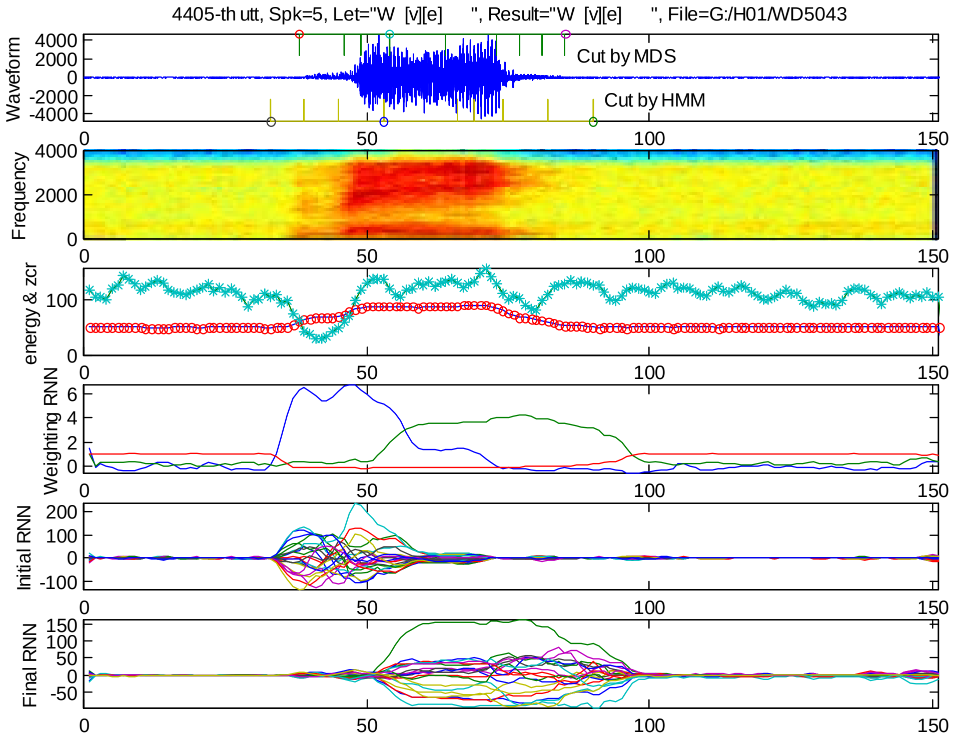

Figure 3. A recognition example of the letter W [ve:]. From top to bottom: The speech wave form, the spectrum, energy and zero crossing rate, output of the weighting RNN, initial and final RNNs.

3.3.2 MRNN with 12 CI-initials, 13 CI-vowels and 11 CI-finals

Although the performance of the HMM scheme using 12 initial, 13 vowel and 11 final sub-word models was relatively poorly, we are still interested in this scheme, since we have so far not used an RNN for segmenting a syllable into initial/vowel/final parts. Furthermore, all HMMs in this scheme have been forced to share their sub-word models. They are therefore not independent and may produce better segmentation positions for MRNN-based approaches. The block diagram of the resulting MRNN is shown in Figure 4. Preliminary results for the initial/vowel and final RNNs are listed in Table 9.

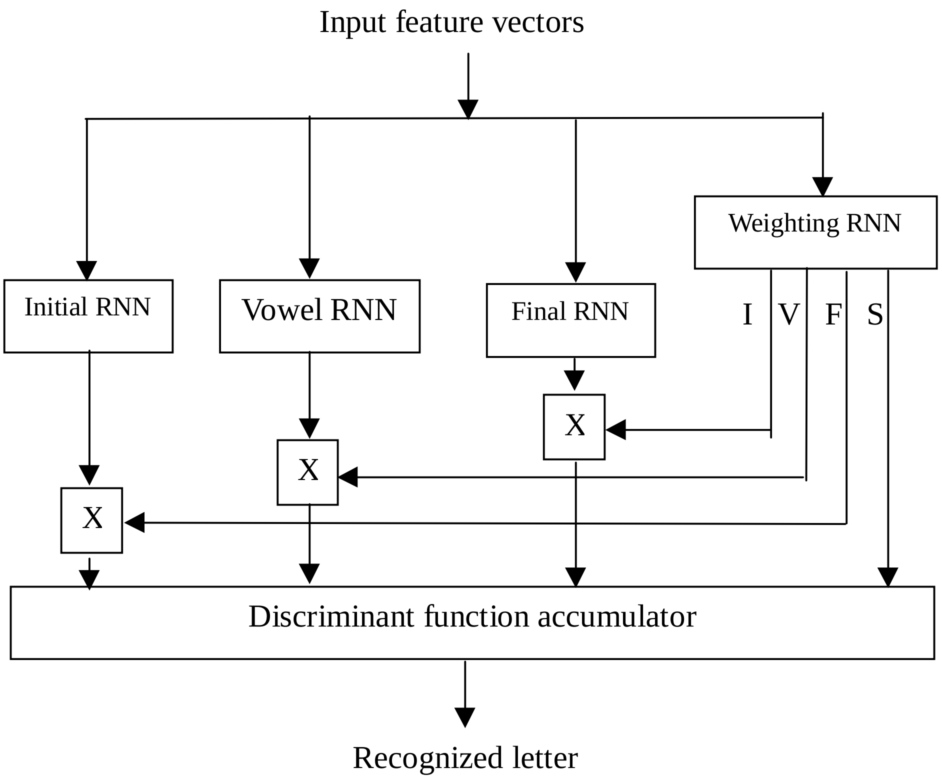

Figure 4. Block diagram of MRNN with 12 CI-initials, 13 CI-vowels and 11 CI-finals.

Table 9. The recognition results for the initial, vowel, final discrimination RNNs.

Segmentation schemes	Recognition rates (%)		
	12 initials	13 vowels	11 finals
12 initials/13 vowels/ 11 finals	93.2	96.7	85.5

4. Discussion and Conclusions

All structures of MRNNs examined in the current study are outperformed by the corresponding HMMs. Overall recognition rates compare to those achieved in an earlier study [5].

Case analysis reveals that most errors are due to failures in the endpoint detection, which tends to leave small segments of silence within the word boundaries. Therefore, the first and last states of the HMMs are polluted with silence and sometimes act as additional silence states, yielding widened word boundaries. The initial/final/silence states of the HMMs are therefore often mal-aligned, as the HMMs are trained using the ML algorithms, which only considers the within-class training. And many word models used, although made up of two sub-word models, do not share their sub-word models. As a consequence, the models may not optimally segment an input utterance into the initial/final/silence parts.

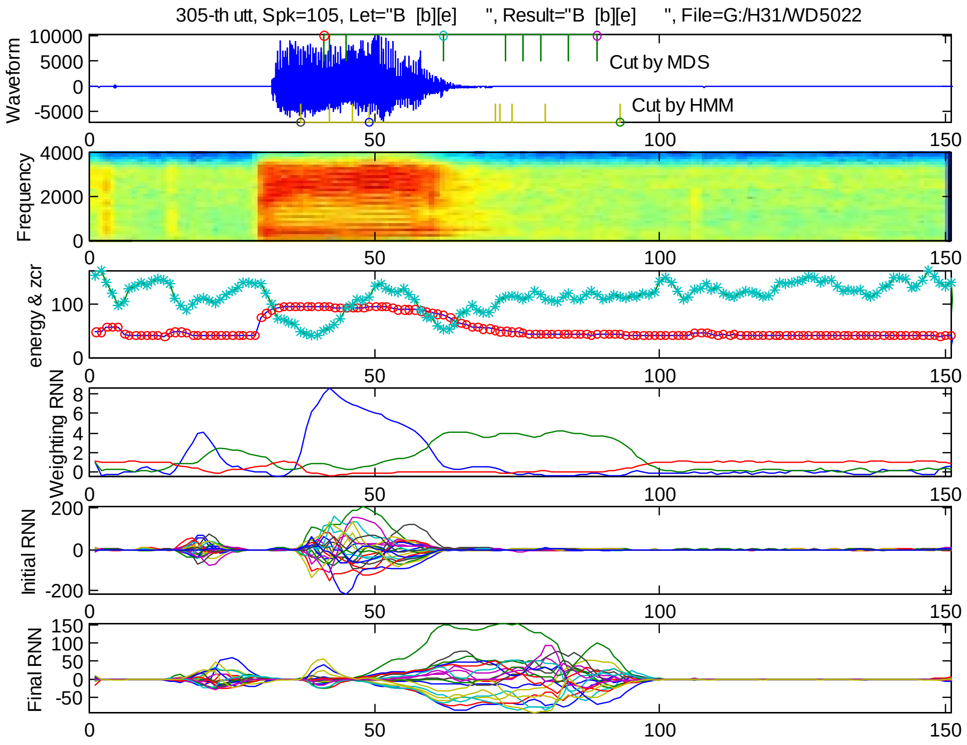

Figure 5. An example of erroneous endpoint detection (letter B [be:]). From top to bottom: The speech wave form, the spectrum, energy and zero crossing rate, output of the weighting RNN, initial and final RNNs. The final RNN reaches its output maximum late during the vowel [e:], the final-RNN output is active until well after the offset of the speech signal.

Since the MRNNs rely on the boundary positions produced by the HMMS, as a starting point and are trained with the competitive training algorithm, they are very sensitive to those segmentation errors. Hence mal-aligned boundaries will misguide and confuse the weighting, initial and final RNNs in the parallel training phase. The MRNN then are not able to recover from those errors in the following fine-tuning phase.

In the case of isolated Mandarin syllable recognition, the problem of segmentation errors does not occur, because the endpoint detection is more reliable for the (C)+V structure. Independent word models do not exist, as the 411 syllables are recognized using shared initial and final models which are forced to align with the true initial/final segments. Thus a better segmentation algorithm is needed for more successfully implementing the MRNN for German alphabet recognition.

This work was supported by DFG (Deutsche Forschungsgemeinschaft), grant no. MI 625.

5. References

[1] Chen, W.Y., Y.F. Liao and S.H. Chen. Speech recognition with hierarchical recurrent neural networks, *Pattern Recognition*, Vol.28, No.6, pp.795-805, 1995.
[2] Glaeser, A. *Modulare neuronale Netze zur aufwandsreduzierten Spracherkennung*. PhD thesis, Technische Hochschule Darmstadt. 1996.
[3] Tscheschner, W.. *Kommunikation und Kommunikationsgeräte*. 2. Lehrbrief. TU Dresden 1982.
[4] Y.F. Liao and S.H. Chen. An MRNN-Based Method For Continuous Mandarin Speech Recognition. *Proceedings ICASSP 98*, Vol.2, pp. 1121-1124, 1998.
[5] H. Hild and A. Waibel. Recognition of Spelled Names over the Telephone. *Proceedings of ICSLP 96*, vol.1., pp. 346-349.