QUANTILE BASED HISTOGRAM EQUALIZATION FOR ONLINE APPLICATIONS

Florian Hilger, Sirko Molau, and Hermann Ney

Lehrstuhl für Informatik VI
RWTH Aachen – University of Technology
Ahornstr. 55
D – 52056 Aachen, Germany
{hilger, molau, ney}@informatik.rwth-aachen.de

ABSTRACT

The noise robustness of automatic speech recognition systems can be increased by transforming the signal to make the cumulative density functions of the signal's values in recognition match the ones that where estimated on the training data. This paper describes a real–time online algorithm to approximate the cumulative density functions, after Mel scaled filtering, using a small number of quantiles. Recognition tests where carried out on the Aurora noisy TI digit strings and SpeechDat–Car databases. The average relative reduction of the word error rates was 32% on the noisy TI digit strings and 29% on SpeechDat–Car.

1. INTRODUCTION

Background noises or distortions caused by the transmission usually lead to mismatch between the test conditions and the training data of automatic speech recognition systems. A mismatch can severely deteriorate the recognition performance. To improve the performance, the mismatch should be reduced by adaptation of the recognizers' references to the noise and/or a feature extraction that reduces the influence of the noise to keep the mismatch small [1].

Quantile based histogram equalization ("quantile equalization") as it was introduced in [2] is an approach to keep the mismatch small by transforming the signals during the MFCC feature extraction. The idea is to make the cumulative density functions of the signal's values in testing match the ones observed on the training data. The cumulative density functions are roughly approximated using a small number (here four) of quantiles (Figure 1). Using this approximation the approach is suitable for real–time online applications where only a short delay is allowed and the noise conditions can change rapidly.

This work was partially funded by the DFG (Deutsche Forschungsgemeinschaft) under contract NE 572/4-1.

2. QUANTILE EQUALIZATION

Quantile based histogram equalization can in principle be applied at any stage of the feature extraction [3]. Depending on the position of the quantile equalization an adequate transformation function T_k has to be chosen.

$$Y_k^{eq}[t] = T_k(Y_k[t]) \tag{1}$$

In this paper $Y_k[t]$ denotes the output of the kth Mel scaled filter after applying a 10th root compression (Figure 2) at time frame t. The 10th root compression gave lower baseline error rates than the usual logarithm on the databases it was tested on.

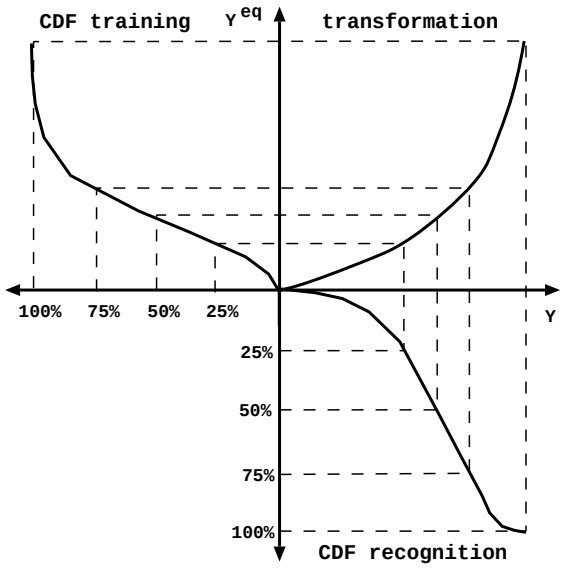

Fig. 1. Applying a transformation function to make the four training and recognition quantiles match.

The transformation function T_k that is actually used here is a power function [2]. The symbols used in the following equations are: N_Q the number of quantiles. Q_i^{train} the ith

quantile on the training data, these are estimated globally not dependent on the filter channel k. $Q_{k,i}$ the ith quantile estimated on the test utterance for filter channel k. To avoid scaling up noises which are lower than in training, lower bounds for $Q_{k,i}$ are defined:

$$\text{if} \quad Q_{k,i} < Q_i^{train} \quad \text{then} \quad Q_{k,i} = Q_i^{train} \tag{2}$$

Before actually applying the power function transformation the filter output values $Y_k[t]$ are scaled to the interval $[0,1]$ by dividing them through the maximal value Q_{k,N_Q} (on some databases the recognition performance can be improved by using an overestimation factor i.e. $o \cdot Q_{k,N_Q}$ instead of the original value). Then the transformation is applied and the resulting values are scaled back to the original range:

$$T_k(Y_k[t]) = \\ Q_{k,N_Q} \left(\alpha_k \left(\frac{Y_k[t]}{Q_{k,N_Q}} \right)^{\gamma_k} + (1-\alpha_k) \frac{Y_k[t]}{Q_{k,N_Q}} \right) \tag{3}$$

The transformation parameters α_k and γ_k are chosen to minimize the squared distance between the current quantiles $Q_{k,i}$ and the training quantiles Q_i^{train}:

$$\{\gamma_k, \alpha_k\} = \underset{\{\gamma_k, \alpha_k\}}{\operatorname{argmin}} \left(\sum_{i=1}^{N_Q-1} \left(T_k(Q_{k,i}) - Q_i^{train} \right)^2 \right) \tag{4}$$

A grid search is used to find the optimal transformation parameters in the ranges $\alpha_k \in [0,1]$ and $\gamma_k \in [1, max]$. By limiting the maximal value of γ_k to e.g. $max = 3$, the maximal amount of transformation can be restricted which generally leads to better recognition results.

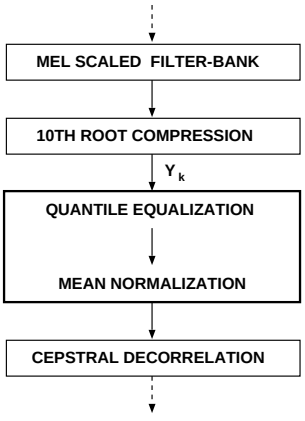

Fig. 2. Position of the quantile equalization and mean normalization modules after the Mel scaled filter bank and 10th root compression.

In previous work [2] it was shown that utterance wise quantile equalization can successfully be combined with an additional utterance wise (cepstral) mean normalization. The following section will describe how to combine quantile equalization and mean normalization in a way that can be used in real–time online applications.

3. ONLINE IMPLEMENTATION

Quantile equalization and mean normalization can both be implemented using a sliding window instead of the whole utterance. When simply applying the two techniques successively their individual delays would add up. To reduce the delay, the combined normalization scheme illustrated in Figure 3 can be applied.

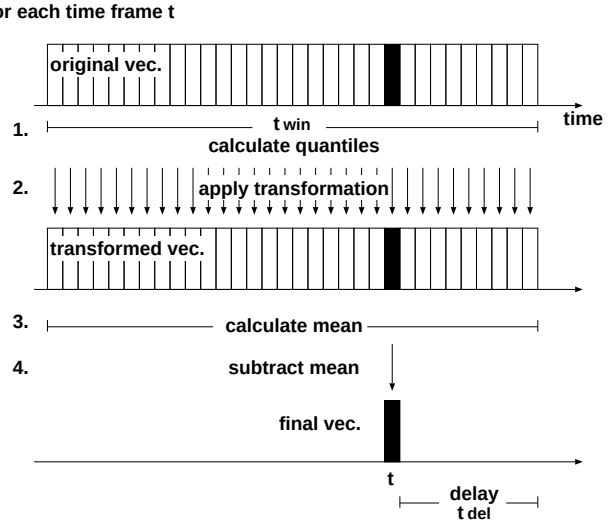

Fig. 3. Online normalizing scheme for the feature vectors after the Mel filter bank and 10th root compression.

For each time frame t:

1. Calculate the signal's quantiles $Q_{k,i}$ for each filter channel within a window around the current time frame. The window length t_{win} and the delay t_{del} should be chosen depending on the average utterance length and the delay that is allowed for the application, for example t_{win} = 1s to 5s and t_{del} = 10ms to 500ms
2. Determine the optimal transformation parameters α_k and γ_k (equation 4) and apply the transformation to all the vectors in the window. Some additional remarks on how to initialize and update α_k and γ_k can be found in the paragraph below.
3. Calculate the mean values of the transformed vectors within the window.
4. Subtract the mean to get the resulting vector.

After that step the feature extraction continues as usual with the calculation of the cepstral coefficients and derivatives.

The way of updating the transformation parameters α_k and γ_k in the online version has a strong influence on the recognition performance. When using a full grid search as described in Section 2 in every time frame, the transformation parameters tend to change significantly from one time frame to the next. This leads to a large amount of insertion errors and error rates higher than baseline. To counteract that effect, the updated values are only searched in the neighborhood of the old ones $\alpha_k[t-1] \pm \delta$ and $\gamma_k[t-1] \pm \delta$, with a value of δ in the range of 0.01. Thus there are no sudden changes of the transformation function and the number of insertion errors is reduced. As positive side effect the computational load is reduced significantly. The initial values used in the first time frame are $\alpha_k = 0$ and $\gamma_k = 1$ which corresponds to no transformation.

4. RECOGNITION RESULTS

Database definitions: The recognition tests where carried out on the Aurora databases distributed by ELRA: TI digit strings with added noises [4] and the digit string subsets of SpeechDat–Car in Danish, Finnish, German, and Spanish. The sampling rate of the recordings is 8kHz. Since it was not the intention of this work to evaluate voice activity detection algorithms the test data was segmented corresponding to the new official baselines results (200ms of silence left before and after each utterance) before further processing.

Recognizer setup: For training and all recognition tests the HTK speech recognition toolkit was used in the original setup defined for the ETSI Aurora evaluations [4] [5].

- HTK speech recognition toolkit (Aurora evaluation settings [4])
- Word models of fixed length (16 states) for the digits
- Gender independent models
- Gaussian mixtures

The front end is a modified version of the original Aurora WI007 MFCC feature extraction [4].

- Aurora WI007 MFCC feature extraction front end [4]
- logarithm replaced by 10th root
- Quantile equalization and mean normalization module added between 10th root and the calculation of the cepstral coefficients (Figure 2)
- 0th cepstral coefficient used instead of log energy

For the following tests a delay of only one time frame i.e. $t_{del} = 10$ms and a window length of $t_{win} = 5$s was used. The overestimation factor for Q_{k,N_Q} was 1.25 for the TI digit strings and 1.5 for the SpeechDat data. The training quantiles Q_i^{train} where always estimated on the corresponding training data sets. When carrying out the feature extraction on the training data only the mean normalization was applied. Quantile equalization was switched off.

Results SpeechDat–Car: Table 2 shows the recognition results on the SpeechDat–Car database with these settings. The overall average recognition performance improvement compared to the baseline feature extraction is 29%. The results clearly show that the relative improvement increases as expected with a growing amount of mismatch between training and testing conditions. The largest relative reductions were obtained on the high mismatch data. Looking at the performance for the different languages, the proposed algorithm apparently works best on the Finnish data. While the average improvements on the other languages are in the order of 20%–25% the result for Finnish is 45%.

Results Noisy TI digit strings: The overall average improvement on this database is 32% (Table 3). Using clean training data the mismatch between training and test is high, quantile equalization then leads to a large relative improvement of 50%. With multi condition training the average improvement is 15%. On test set C which has mismatched channel characteristics the improvement is higher than on A and B.

Additional tests: The approach was also tested on a database containing isolated German words [1], with training data collected in a quiet office and mismatched testing data collected in cars (city and highway traffic, microphone on the visor). The recognizer vocabulary consists of 2100 equally probable words. For the tests on this database the RWTH feature extraction and speech recognition system [2] was used. Here, the delay was $t_{del} = 500$ms and the window length $t_{win} = 1$s. The results are shown in Table 1

Table 1. *Recognition results on the isolated word car navigation database. CMN: baseline MFCC front end with log compression and cepstral mean normalization, MN: 10th root compression and mean normalization, QE + MN: combined quantile equalization and mean normalization.*

Isolated Word Car Navigation Database			
SNR [dB]	Word Error Rate [%]		
	CMN	MN	QE + MN
office 21	2.9	2.8	3.2
city 9	31.6	19.9	11.7
highway 6	74.2	40.1	20.1

Compared to the MFCC baseline with logarithm and cepstral mean normalization the setup using 10th root compression followed by mean normalization already gave significant error rate reductions on this database. Applying quantile equalization lead to further considerable error rate reductions on the noisy test sets.

Table 2. *Recognition results for the Aurora 3 SpeechDat–Car databases. WM: well matched, MM: medium mismatch, HM: high mismatch, Avg: weighted average (0.4WM+0.35MM+0.25HM)*

Aurora 3 Reference Word Error Rates [%]					
	Finnish	Spanish	German	Danish	Average
WM	7.26	7.06	8.80	12.72	8.96
MM	19.49	16.69	18.96	32.68	21.96
HM	59.47	48.45	26.83	60.63	48.85
Avg	24.59	20.78	16.86	31.68	23.48

Aurora 3 Word Error Rates [%]					
	Finnish	Spanish	German	Danish	Average
WM	4.52	7.83	7.53	12.43	8.08
MM	12.11	10.10	16.47	23.48	15.54
HM	20.14	16.54	16.51	26.58	19.94
Avg	11.08	10.80	12.90	19.84	13.66

Aurora 3 Relative Improvements [%]					
	Finnish	Spanish	German	Danish	Average
WM	37.74	-10.91	14.43	2.28	10.89
MM	37.87	39.48	13.13	28.15	29.66
HM	66.13	65.86	38.46	56.16	56.66
Avg	44.88	25.92	19.99	24.81	28.90

5. CONCLUSIONS

This paper has described quantile based histogram equalization for real–time online applications. It was shown that a small amount of adaptation data is sufficient to approximate a speech signal's cumulative density functions at the filter bank outputs by using quantiles. These quantiles where used to calculate transformation functions which reduced an eventual mismatch between training and test conditions of a speech recognition system. Previous experiments had shown that an additional mean normalization step can improve the overall performance. Here, the quantile based histogram equalization and mean normalization where combined in a way that keeps the resulting total delay small.

The experiments on the Aurora databases have shown that significant error rate reductions can be obtained even when the delay is reduced to one time frame. As expected the relative error rate reductions where largest in high mismatch conditions.

An important experimental result was that the way of updating the transformation function's parameters from one time frame to the next had a strong influence on the error rates. Further work will have to investigate an optimized updating scheme to replace empirical parameter optimization.

Table 3. *Recognition results for the Aurora 2 noisy TI digit strings. Multi: training with noise added at different SNRs. Clean: training without additional noise. Set A–C: different noise conditions for testing [4]*

Aurora 2 Reference Word Error Rates [%]				
	Set A	Set B	Set C	Overall
Multi	11.93	12.78	15.44	12.97
Clean	41.26	46.60	34.00	41.94
Average	26.59	29.69	24.72	27.46

Aurora 2 Word Error Rates [%]				
	Set A	Set B	Set C	Overall
Multi	10.20	10.75	10.76	10.53
Clean	23.53	21.90	22.36	22.64
Average	16.86	16.32	16.56	16.59

Aurora 2 Relative Improvements [%]				
	Set A	Set B	Set C	Overall
Multi	10.57	14.80	23.58	14.87
Clean	43.35	59.90	42.03	49.71
Average	26.96	37.35	32.80	32.29

6. REFERENCES

[1] F. Hilger and H. Ney, "Noise Level Normalization And Reference Adaptation For Robust Speech Recognition," in *ASR2000 – International Workshop on Automatic Speech Recognition*, pp. 64–68, Paris, France, Sept. 2000.

[2] F. Hilger and H. Ney, "Quantile Based Histogram Equalization for Noise Robust Speech Recognition," in *Proc. of the 7th European Conference on Speech Communication and Technology*, vol. 2, pp. 1135–1138, Aalborg, Denmark, Sept. 2001.

[3] S. Molau, M. Pitz, and H. Ney, "Histogram Based Normalization In The Acoustic Feature Space," in *ASRU 2001 – Automatic Speech Recognition and Understanding Workshop*, Madonna di Campiglio, Trento, Italy, Dec. 2001.

[4] H.-G. Hirsch and D. Pearce, "The Aurora Experimental Framework for the Performance Evaluation of Speech Recognition Systems Under Noisy Conditions," in *ASR2000 – International Workshop on Automatic Speech Recognition*, pp. 181–188, Paris, France, Sept. 2000.

[5] D. Pearce, "Enabling New Speech Driven Services for Mobile Devices: An overview of the ETSI standards activities for Distributed Speech Recognition Frontends," in *Applied Voice Input/Output Society Conference (AVIOS2000)*, San Jose, CA, May 2000.