# COMBINING NEIGHBORING FILTER CHANNELS TO IMPROVE QUANTILE BASED HISTOGRAM EQUALIZATION

*Florian Hilger, Hermann Ney*

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

*Olivier Siohan, Frank K. Soong*

Multimedia Communications Research Lab
Bell Laboratories, Lucent Technologies
600 Mountain Avenue
Murray Hill, NJ, 07974, USA
{siohan, fks}@research.bell-labs.com

## ABSTRACT

A mismatch between the training data and the test condition of an automatic speech recognition system usually deteriorates the recognition performance. Quantile based histogram equalization can increase the system's robustness by approximating the cumulative density function of the current signal and then reducing an eventual mismatch based on this estimate. In a first step each output of the mel scaled filter bank can be transformed independent from the others. This paper will describe an improved version of the algorithm that combines neighboring filter channels. On several databases recorded in real car environment the recognition error rates could be significantly reduced with this new approach.

## 1. INTRODUCTION

Histogram based methods that remove an eventual mismatch between the current distribution of the test data and the distribution of the system's training data have been successfully used to increase the robustness of speech recognition systems [1] [2].

If some minutes of test data are available to estimate the histograms a non parametric transformation [1] to reduce the mismatch can be calculated and applied. An alternative which can be used for online systems that only allow short delays is quantile based histogram equalization [3] [4]. This method approximates the cumulative density functions of the signals using a few quantiles and then optimizes the parameters of a transformation function based on these values.

In previous work [4] the transformation was a function that transformed each filter channel of the mel scaled filter bank individually. This paper will show how the approach can be improved by introducing a second transformation step that linearly combines neighboring filter channels.

This work was partially funded by the DFG (Deutsche Forschungsgemeinschaft) under contract NE 572/4-1 and by DARPA under grant N66001-00-8013

## 2. QUANTILE EQUALIZATION

The general algorithm for the online version of quantile based histogram equalization with mean normalization [4] is depicted in Figure 2. Within a window around the current time frame the some (typically $N_Q = 4$) quantiles of the signal are calculated for each filter bank output $Y_k$. A transformation that minimizes the squared distance between the current quantiles $Q_{k,i}$ and the average training quantiles $Q_i^{train}$ is determined and applied to all vectors within the window. The mean of the transformed values is calculated and subtracted from the current vector.

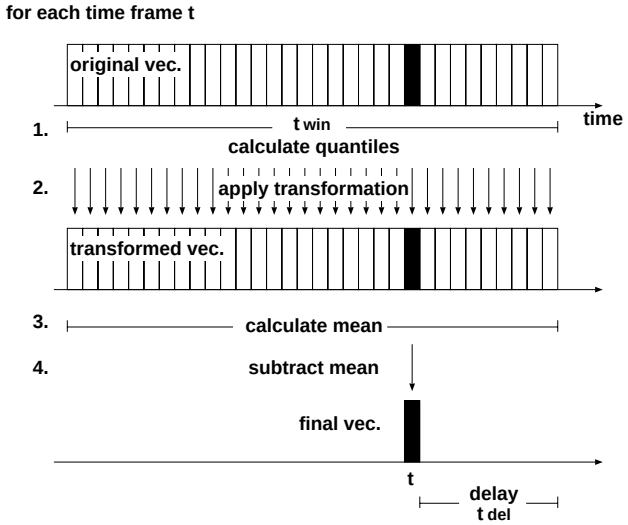


**Fig. 1**. Online normalizing scheme for the feature vectors after the Mel filter bank and 10th root compression.

The concept of finding an appropriate transformation and applying it is very general. Any function that reduces the mismatch between training and test could possibly be used and might improve the recognition performance. In

following power function transformation combined with an additional linear term is used. The filter output values $Y_k[t]$ are scaled to the interval $[0, 1]$, transformed, and then scaled back to the original range:

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

The scaling before the transformation ensures that values close to zero and high values are not transformed. Figure 2 shows that the non–speech background level has the largest mismatch. The high speech peaks still stick out of the background and are not that mismatched, the cumulative density functions (Figure 3) also show this clearly.

The transformation parameters are initialized with the values $\alpha_k = 0$ and $\gamma_k = 1$. This corresponds to no transformation. In the following time frames a grid search in a small range around the previous values is carried out [4], and the new values that minimize the the squared distance between the current quantiles $Q_{k,i}$ and the training quantiles $Q_i^{train}$ are chosen:

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

In the example (Figure 2) the signal with quantile equalization is identical to the noisy signal in the first time frames and then converges towards the clean signal. The match still is not perfect but plotting the cumulative density function (Figure 3) shows that the mismatch between the transformed signal and the clean one is considerably smaller.

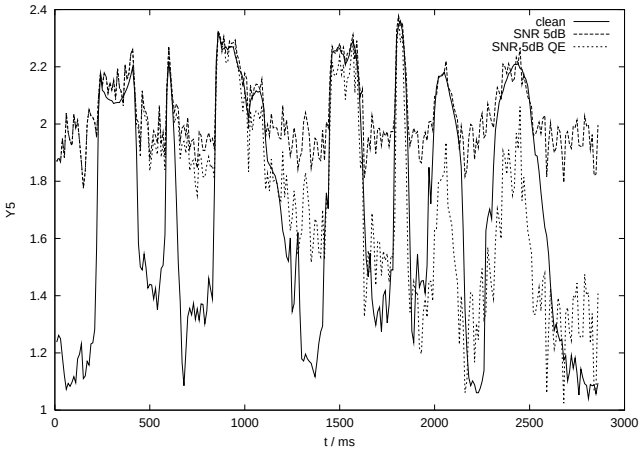


**Fig. 2**. Output of the fifth mel scaled filter for the spoken digit string "5674683". Clean, with car noise added at 5dB, and after applying online quantile equalization.

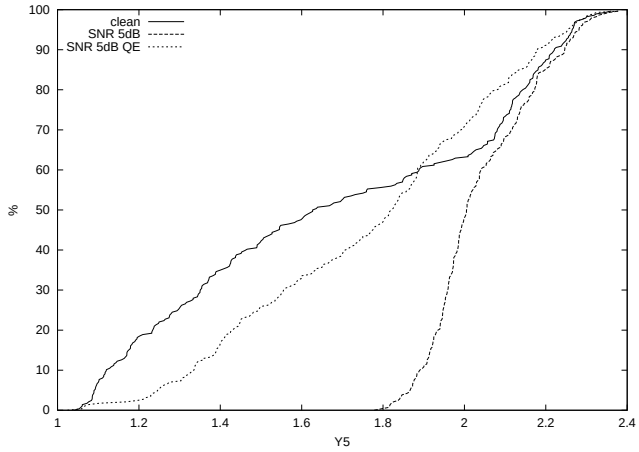


**Fig. 3**. Cumulative density functions of the signals shown in Figure 2

So far each filter bank output is transformed individually and possible interdependencies between neighboring filters are not taken into consideration. But it is likely the filters are not completely independent and combining them can improve the recognition performance. In [1] it was shown that a feature space rotation after histogram equalization can significantly improve the recognition performance if enough data is available to reliably estimate a scatter matrix and its principal component. The moving window of a few seconds that is used in the online approach will not be sufficient to estimate a reliable rotation matrix, but a very simple alternative method of combining the filters can already improve the recognition performance.

## 3. COMBINING NEIGHBORING FILTER CHANNELS

In the following equations $\tilde{Q}_{k,i}$ are the recognition quantiles after the power function transformation (Equation 1). The parametric power function transformation can not guarantee that training and recognition quantiles match perfectly, so in a second step a linear combination of a filter with its left and right neighbor can be used to further reduce the remaining difference.

$$\tilde{T}_k(\tilde{Q}_k) = (1-\lambda_k-\rho_k)\tilde{Q}_k + \lambda_k\tilde{Q}_{k-1} + \rho_k\tilde{Q}_{k+1} \quad (3)$$

Like the power function transformation factors $\alpha_k$ and $\gamma_k$ the linear combination factors $\lambda_k$ and $\rho_k$ are chosen to minimize the squared distance between the training and the recognition quantiles.

$$\{\lambda_k, \rho_k\} = \underset{\{\lambda_k,\rho_k\}}{\operatorname{argmin}} \left( \sum_{i=1}^{N_Q-1} \left( \tilde{T}_k(\tilde{Q}_{k,i}) - Q_i^{train} \right)^2 + \beta \left( \lambda_k^2 + \rho_k^2 \right) \right) \quad (4)$$

If the possible values of the combination factors $\lambda_k$ and $\rho_k$ are not restricted, the recognition performance will deteriorate. They can be limited to a fixed small range e.g. 0 to 0.1 but in general better results are obtained when a penalty factor like $\beta \left( \lambda_k^2 + \rho_k^2 \right)$ is used. A factor of typically $\beta = 1/50$ will usually limit $\lambda_k$ and $\rho_k$ to values smaller than 0.1, but will also allow higher values if the difference between the test quantile and the training quantile is still large.

When updating $\lambda_k$ and $\rho_k$ in the online implementation, the new values are also only searched in a small range around the previous value to avoid sudden changes.

After finding the optimal transformation parameters the transformation is applied to all feature vectors in the window around the current time frame (Figure 2). The mean of these transformed vectors is then calculated and subtracted, before the calculation of the cepstral coefficients, derivatives and an eventual linear discriminant analysis.

## 4. RECOGNITION RESULTS

Several databases recorded in real car environment and two speech recognition systems were used to test the quantile equalization with filter combination algorithm.

**Isolated Word Car Navigation Database:** the training data for this German database was collected in a quiet office, the testing data was collected in cars (city and highway traffic, microphone on the visor). The RWTH feature extraction and speech recognition system with the setup described in [3] was used for the tests. The recognizer's vocabulary consisted of 2100 equally probable isolated words. The delay for the online quantile equalization was set to 500ms with a window length of 1000ms. The filter combination used a penalty factor to restrict the values of the combination factors (Equation 4).

Simply replacing the logarithm in the feature extraction by a 10th root compression reduced the error rates on the noisy test sets significantly (Table 1). The normal quantile equalization leads to a further relative improvement in the order of 50%. The filter combination gives an other improvement, compared to the normal version of quantile equalization the error rates are reduced from 11.7% to 10.3% on the city test set and from 20.1% to 17.1% for highway, but there is some loss in performance on the clean data.

**Table 1**. *Recognition results on the isolated word car navigation database. baseline: MFCC front end with log compression and cepstral mean normalization, 10th: 10th root compression and mean normalization [4], QE: quantile equalization [4], QEF: quantile equalization with filter combination.*

| Isolated Word Car Navigation Database | | | | |
|---|---|---|---|---|
| SNR [dB] | Word Error Rate [%] | | | |
| | baseline | 10th | 10th+QE | 10th+QEF |
| office 21 | 2.9 | 2.8 | 3.2 | 3.6 |
| city 9 | 31.6 | 19.9 | 11.7 | 10.3 |
| highway 6 | 74.2 | 40.1 | 20.1 | 17.1 |

On this database the recognition results on the noisy test sets could be further improved by applying quantile equalization during training too (Table 2). In a first pass over the training data the reference quantiles of the training data were estimated. These were then used as target values for the transformation of the training as well as the test data.

**Table 2**. *Recognition results on the isolated word car navigation database. Quantile equalization (QE) resp. quantile equalization with filter combination (QEF) applied during training too.*

| Isolated Word Car Navigation Database | | |
|---|---|---|
| SNR [dB] | Word Error Rate [%] | |
| | 10th + QE training | 10th + QEF training |
| office 21 | 3.3 | 3.8 |
| city 9 | 10.1 | 9.0 |
| highway 6 | 16.7 | 15.4 |

Applying quantile equalization in training distorts the clean training data in a way that improves the recognition on the mismatched noisy data (Table 2) – but the price for these improvements is once again a deterioration on the clean testing data. Experiments on databases, with noisy training data like the ones described below, showed a general deterioration of the recognition performance when applying quantile equalization in training too. Probably applying quantile equalization in these cases reduces the diversity of the noisy training data and thus reduces the recognition performance.

**Car VUI Database:** the 8h 53min of training data were recorded in cars using a close talking microphone (different driving conditions, some recordings with background music), the test data (30min, digit strings and command phrases) with a microphone mounted on the visor. The Lucent Bell Labs speech recognition system with the setup described in [5] was used for the following test. The digits and 85 different command words were modeled with triphones. A finite state grammar determines the allowed command phrases. Here the quantile equalization window was set to 10ms delay and a length of 5 seconds. The filter combination used a penalty factor again.

Like on the previous database the 10th root compression alone leads to a significant improvement over the baseline (Table 3). The further improvement of the quantile equalization is not as large here, because even though a close talking microphone was used for the recordings, the training data is somewhat noisy so the overall mismatch between the training and test data is not as large as it was on the previous database. However the combination of the filters can still reduce the error rate to 7.6% which is a 20% improvement over the best result with normal quantile equalization.

**Table 3**. *Recognition results on the digit string and command phrase Car VUI database. visor: microphone mounted on the visor. QE: quantile equalization, QEF: quantile equalization with filter combination.*

| Car VUI Database | | | | |
|---|---|---|---|---|
| | WER [%] | | | |
| | baseline | 10th | 10th+QE | 10th+QEF |
| visor | 20.2 | 10.2 | 9.5 | 7.6 |

**IH Digit String Database:** a total of 88h 16min training data (8kHz sampling rate, recorded in cars with various telephone handsets) was used for the following tests. The different digit string test sets were also recorded in cars. Handset (24min), tele_tsclr (68min), sdn10 (5h 18min) are different telephone handsets. Lapel (58 min) was recorded using a lapel microphone. Lucent Bell Labs' speech recognition system, with context dependent head–body–tail digit models [6], was used for the recognition tests. For these tests an utterance wise quantile equalization was used. The filter combination factors were restricted to a fixed range of $[0, 0.1]$ without using a penalty factor, which in this case gave better results.

**Table 4**. *Recognition results on the IH digit string car database. handset, tele_tsclr, sdn10: different handheld telephones, lapel: lapel microphone. QE: quantile equalization, QEF: quantile equalization with filter combination.*

| IH Car Database | | | | |
|---|---|---|---|---|
| | WER [%] | | | |
| | baseline | 10th | 10th+QE | 10th+QEF |
| handset | 0.5 | 0.5 | 0.4 | 0.3 |
| tele_tsclr | 0.8 | 0.8 | 0.8 | 0.8 |
| sdn10 | 2.7 | 2.5 | 2.5 | 2.5 |
| lapel | 3.2 | 3.3 | 2.8 | 2.6 |

The baseline error rates on this database are much lower than on the previous databases and mismatch between the training and test conditions is small, so only small improvements can be expected from the quantile equalization. On two of the telephone handsets small improvements can be observed (Table 4). Only the mismatched lapel test set shows a significant improvement from 3.2% word error rate to 2.8% using normal quantile equalization and 2.6% when combining the filters.

## 5. CONCLUSIONS

In this paper an improved version of quantile equalization for online applications was presented. After transforming each filter bank output individually, neighboring filters were combined linearly to further reduce the mismatch between training and recognition quantiles. On several databases with a large mismatch between the training and the test conditions this new approach leads to relative improvements of 10% to 20% compared to the previous version. On a database with clean training data, the recognition of noisy test data could be further improved by applying quantile equalization in training too.

Future work will have to show if a more sophisticated way of combining the filter channels, e.g. by using approximations to rotation matrices, can perform better than this very simple linear combination of neighboring filters.

## 6. REFERENCES


[1] S. Molau, F. Hilger, and H. Ney, "Enhanced histogram normalization in the acoustic feature space," in *Proc. of the 7th International Conference on Spoken Language Processing*, vol. 1, pp. 1421–1424, Denver, CO, USA, Sept. 2002.

[2] J. C. Segura, M. C. Benitez, A. de la Torre, and A. J. Rubio, "Feature extraction combining spectral subtraction and cepstral histogram equalization," in *Proc. of the 7th International Conference on Spoken Language Processing*, vol. 1, pp. 225–228, Denver, CO, USA, Sept. 2002.

[3] 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.

[4] F. Hilger, S. Molau, and H. Ney, "Quantile based histogram equalization for online applications," in *Proc. of the 7th International Conference on Spoken Language Processing*, vol. 1, pp. 237–240, Denver, CO, USA, Sept. 2002.

[5] M. Afify and O. Siohan, "Sequential noise estimation with optimal forgetting for robust speech recognition," in *Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, vol. I, pp. 229–232, Salt Lake City, UT, USA, May 2001.

[6] M. Afify, H. Jiang, F. Korkmazskiy, C.-H. Lee, Q. Li, O. Siohan, F. K. Soong, and A. C. Surendran, "Evaluating the aurora connected digit recognition task – A Bell Labs approach," in *Proc. of the 7th European Conference on Speech Communication and Technology*, vol. 1, pp. 633–637, Aalborg, Denmark, Sept. 2001.