Additive Noise and Channel Distortion-Robust Parametrization Tool - Performance Evaluation on Aurora 2 & 3

Petr Fousek, Petr Pollák

Czech Technical University in Prague
CTU FEL K331, Technická 2, 166 27 Praha 6, Czech Republic
{fousekp.pollak}@feld.cvut.cz

Abstract

In this paper a HTK-compatible robust speech parametrization tool CtuCopy is presented. This tool allows for the usage of several additive noise suppression preprocessing techniques, non-linear spectrum transformation, RASTA-like filtration, and direct final feature computation. The tool is general, it is easily extendable, and it may be also used for speech enhancement purposes. In the second part, parametrizations combining the extended spectral subtraction for additive noise suppression and LDA RASTA-like filtration for channel-distortion elimination with final computation of PLP cepstral coefficients are examined and evaluated on Aurora 2 & 3 and Czech SpeechDat corpora. This comparison shows specific algorithm features and the differences in their behavior on above mentioned databases. PLP cepstral coefficients with both extended spectral subtraction and LDA RASTA-like filtration seem to be a good choice for noise robust parametrization.

1. Introduction

Since the speech recognition systems began to integrate to an ordinary life worldwide, the robustness is nowadays the prior requirement of ASR system. It leads to an extensive research in the field of robust parametrization. There are two approaches for dealing with the problem. Firstly, it is the looking for a parametrization technique which is in principle robust to environmental speech background. Secondly, disturbing background environmental noise may be removed before the parametrization. The experience of former research shows that it seems to be good compromise to combine noise removal technique and suitable parametrization in front-end. The development of such a front-end often represents combining methods for the noise and distortion suppression and their repetitive evaluation. Consequently, the computation costs are high and it is likely to encapsulate the whole processing into one flexible front-end tool which is able to perform effectively.

2. CtuCopy - extension to HTK

When combining several methods in front-end processing, the algorithm steps often overlap (e.g. FFT when operating in spectral domain) so there is a lot of redundancy in computation and there is also the loss in accuracy of the numbers representation, especially when using various software tools which implement the algorithms differently. Furthermore, the data is being transferred among the programs and it leads to a higher system resources and storage space consumption. When enclosed in one program the algorithm steps can be shared, the processing advance is consistent and effective.

2.1. Front-end structure

The CtuCopy is a widely configurable object-oriented C++ based tool ready for real-time applications which is an alternative to HCopy from HTK Toolkit [2]. For the purposes of extensibility with new preprocessing and parametrization methods it is implemented as a modular structure (see Fig. 1).

The input signal is preprocessed (preemphasis, segmentation, windowing, offset removal, dither adding) and transformed to the spectrum by FFT. At this point the current version of program offers a number of enhancement techniques.


For the purpose of further processing the amplitude spectrum can be transformed to an auditory-like domain (Bark spectrum with equal-loudness function or Mel-spectrum). As former research announced the benefit of performing the additive noise removal in such a domain[12] bringing computation savings, the sequence of spectrum warping and additive noise removal blocks can be changed.

To deal with a convolutionary distortion, general RASTA-like filtration method is implemented, offering band-specific processing. As the input to the procedure can be used either Mel-spectrum or Bark-spectrum with EQ-LD. The actual filter bank must be specified using external file. Note that this processing method brings an indispensable latency dependent upon
the length of the filter and it can in consequence disable the real-
time processing.

The enhanced spectrum is further passed to the parametriza-
tion block where the following features are extracted: LPC co-
cefficients – $a_k$, Mel-scale cepstral coefficients or LPC cepstral
coefficients or PLP cepstral coefficients – $c_0$, or Mel-spectrum
or PLP-spectrum – $S_k$. Finally, the postprocessing can be per-
domed denoting cepstrum liftering, cepstral normalization and
frame-dropping.

As an alternative to the front-end, the CtuCopy can also act
as a speech signal enhancer. The techniques described above
which operate on non-warped spectrum are now employed for
noise-removal. In this case the enhanced spectrum of segments
is transformed back to time-domain using inverse FFT with
OLA method requiring the phase information extracted from the
original signal. The output enhanced signal can be used for
purposes of mobile communications, noise suppression meth-
ods evaluation, SNR measurement, reference listernings etc.

2.1.1. Input & Output
The program is supposed to be run as a filter in command pipe.
Therefore, it expects data to come from standard input device
and the output is sent to standard output device. While used in
the pipe, the input data are supposed to be continuous and no
header is added to the output. Otherwise, when input or output
streams are redirected using program options, the tool supposes
to be run in batch mode, though the HTK header [2] is added to
the output features.

3. Used parametrization technique
The parametrization technique explored in this work is based
on a well-known PLP parametrization published by Herman-
sky [3]. This approach was chosen against MFCC features for
the result of our previous experiments performed on a clean
speech database SpeechDat-E. The experiments showed a bet-
ner performance of PLP features in noise-free conditions. In ad-
dition, the comparison of MFCC and PLPC performance pub-
lished by Putska [6] result in the same conclusion. Since the
PLP auditory-like spectrum is in closer agreement with a hu-
man perception system than the Mel-spectrum, PLP cepstral
coefficients are believed to perform better even in noisy envi-
ronment. When accompanied by noise suppressive algorithms,
the known robustness handicap of LPC modeling should be al-
leviated enough.

Two the most problematic types of disturbance that cause
loss of recognition accuracy are the additive and convolutionary
noises. The examined front-end combines the Extended spec-
tral subtraction presented by Sovka et al. [1] for additive noise
removal with the LDA-based RASTA-like filtration [5] for con-
volutionary noise removal, both applied prior to PLP modeling.

3.1. LDA filters
For all the experiments the RASTA-like filters were designed
using LDA method [4] on Czech SpeechDat corpus since the
discriminability between classes is known to be rather language-independent [4]. As the LDA requires a phoneti-
cally labeled database, forced alignment was performed using
P+L parametrization-based triphone HMMs on SpeechDat (the
parametrization will be described further). The LDA-filters
were obtained from training database represented by PLP spec-
tra of utterances computed using CtuCopy tool.

3.2. The method
The processing scheme can be seen in Fig. 2.

![Figure 2: Used front-end block scheme.](image)

The input signal is segmented with overlapping, then the
Hamming window and FFT is performed. After this the ex-
tended spectral subtraction algorithm is used to partially re-
move quasi-stationary additive noise. This method does not re-
quire a voice-activity detector, however, several first segments
of the utterance have to contain no speech for the noise esti-
mation purposes. The enhanced spectrum is then converted to
an auditory-like domain which is appropriate for the RASTA-
like filtration. Since the task for the filtration is to deal with
a channel distortion, the logarithmic Bark spectrum domain is
used. For the filtration purposes the data-driven LDA-based fil-
ter design method known from van Vuuren and Hermansky[4] is
employed. The 100 segments long window of a time trajectory
of each band is filtered by the relevant impulse response
and then the inverse logarithm is applied. At the point the spectrum
is supposed to be noise and channel-robust. Finally the rest of
PLP method yields LPC cepstral coefficients as the features for
the recognizer.

No post-processing algorithms are used in this work by rea-
son that the liftering is useless in the Baum-Welch training, and
the other two methods have not been completely imple-
mented yet. The results with post-processing will be available
at the poster. Since no end-pointing was used throughout this
work, for the comparability purposes the WI007 baseline sys-
tem without end-pointing was used.

4. Experiments
Aurora 2 & 3 corpora are suitable platforms for thorough testing
of the performance of noise-robust algorithms with the possibili-
ty to do SNR-dependent evaluation and to expose the methods
to the real-life environment. Moreover, the results are compar-
able with other approaches.

4.1. Parametrizations
Five front-ends were compared:

- **base** – in case of Aurora corpora the WI007 system, in case of
  SpeechDat corpora MFCC coeffs., no preemphasis, 20
  bands, 12 coeffs. + c0 + $\Delta + \Delta \Delta$ coeffs.

- **P** – PLP cepstral coeffs., no preemphasis, 8'th order LPC, 8
  cepstral coeffs. + c0 + $\Delta + \Delta \Delta$ coeffs.

- **P+E** – Extended Spectral Subtraction, PLP cepstral coeffs.,
  no preemphasis, 8'th order LPC, 8 cepstral coeffs. + c0 +
  $\Delta + \Delta \Delta$ coeffs.

- **P+L** – LDA-filter RASTA-like filtration, PLP cepstral coeffs.,
  no preemphasis, 8'th order LPC, 8 cepstral coeffs. + c0 + $\Delta + \Delta \Delta$ coeffs.

1The own vectors obtained from LDA are time-flipped to yield im-
pulse responses.
### 3. Czech SpeechDat-E database

The experiments achieved on Czech databases intent to propose an optimal training process for the front-end while preserving the comparison objectivity.

#### 4.3.1. Experiment setup

Since the SpeechDat-E corpus provides amount of continuous speech, triphone-based HMMs were used for the recognition with a benefit of flexible recognition vocabulary. There were used 46 Czech phonemes + SP and SIL pauses. All of them were modeled by 5-state HMMs with three emitting states. The training involved monophone training with flat-start, 7 retrainings with SP adding and realignment, followed by 9 triphone retrainings with state tying.

As a result of our previous experiments, the optimal segmentation was found to be 32/16 ms over commonly used 25/10 ms (the recognition scores were 94.7% for 25/10 ms in contrast to 97.5% for 25/16 ms). The accuracy fall for segmentation 25/10 ms is caused by the increasing number of insertions. The fixing of the number of insertions can be reached by setting word-end penalization in Viterbi. Actually, it is just a poor substitution for improper exponential state-duration modeling and missing phone and word duration modeling in HMMs [10].

The performance was evaluated by means of recognizing ten Czech isolated and connected digits with a vocabulary of sixteen pronunciation variants.

#### 4.3.2. Results

The SpeechDat-E database is almost noise-free and the only channel mismatch is represented by variances in telephone channels. The experiment should show the influence of applying EXTEN and LDA-RASTA methods and state a comparative results for Aurora databases.

As can be observed from Tab. 1, the differences between PLP-based methods are minimal. All these methods substantially outperform MFCC baseline. The P+E brought no noticeable change from P and the P+L slightly increased the score.

### 4. Results on Aurora 2 clean

When both methods were applied, the accuracy decreased. The result show only a slight improvement of performance when either method is applied but their combination could cause the drop of performance on clean database.

### 4.4. Results on Aurora 2

The task for the experiment on Aurora 2 was to compare the performances between PLP-based parametrizations in effort to explore the SNR-specific features. The segmentation 25/10 ms was used for consistency with baseline system and no word-penalization was used for the purpose of comparability. The clean training data were used for comparison in Tab. 2. The observations are following. The P parametrization performed worse than base in all cases. On the other hand, using P+E the recognition accuracy outperformed both the P and baseline. The P+L filtration performs comparably to P+E even in the c subset where channel distortion is present. When combining both methods P+E+L, the results outperform substantially P+L and P+E, showing that the sphere of action of both methods is relatively disjoint. The results show that when training on clean data, the P+E+L parametrization can form a desirable robust front-end. The aim of the next task was to compare the performance of the P+E+L to the baseline on both clean and multicondition training sets. The results are summarized in Tab. 4 and 5.

It can be seen that in clean conditions the benefit of P+E+L method is significant for medium-range SNR’s. In the channel-mismatched case the improvement is maximal, verifying the assumptions of normalizing features of LDA filtration. In the case of multicondition training the improvement is reasonable only for high-noise data. In the case of channel-mismatch the method still outperforms the baseline, and approves the contribution of LDA-filtration.

### 4.5. Results on Aurora 3

The Aurora 3 databases represent real environment for recognition. The experiment with P+E+L front-end was supposed to show the differences in behavior on various languages and conditions. The overall results are presented in Tab. 3.

A high improvement of accuracy can be noted in high-mismatch conditions. In well-matched and medium-mismatched conditions the differences between corpora are very high, and no systematic improvement can be seen. Gen-

<table>
<thead>
<tr>
<th>Set</th>
<th>SNR (dB)</th>
<th>Accuracy [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>P</td>
</tr>
<tr>
<td>base</td>
<td></td>
<td>P</td>
</tr>
</tbody>
</table>

Table 2: Results on Aurora 2 clean.

Table 1: Czech SpeechDat results

<table>
<thead>
<tr>
<th>Param.</th>
<th>P</th>
<th>P+E</th>
<th>P+L</th>
<th>P+E+L</th>
<th>base</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acc [%]</td>
<td>97.97</td>
<td>97.54</td>
<td>97.76</td>
<td>97.35</td>
<td>94.67</td>
</tr>
</tbody>
</table>

P+E+L – Extended Spectral Subtraction, LDA-based RASTA-like filtration, PLP cepstral coeffs., no preemphasis, 8th order LPC, 8 cepstral coeffs + c0 + Δ + ΔΔ coefs.

Zeroth cepstral coefficient was used instead of segmental energy due to slightly better results. The rest of settings for MFCC is consistent with the Aurora baseline. The number of coefficients and LPC order for PLP analysis was found to be optimal in [6]. The derivatives were used for L instead of LDA2 and LDA3 filters, because they can be computed “on the fly” saving up disc space and performing only a little worse than LDA2 & 3.

#### 4. Intentsions

The main goals of our experiments were to extend the former testing of the Extended spectral subtraction method [13],[12] with testing on well defined condition sets, analyze the effect of LDA-based RASTA-like filtration for filters designed on Czech, explore the benefit of the combination of EXTEN and RASTA and evaluate the performance of the proposed front-end algorithm on Aurora 2 & 3.
Table 3: Results of P+E+L front-end compared to WI007 baseline on Aurora 3.

<table>
<thead>
<tr>
<th>Cond.</th>
<th>wm</th>
<th>mm</th>
<th>hm</th>
<th>wm</th>
<th>mm</th>
<th>hm</th>
<th>wm</th>
<th>mm</th>
<th>hm</th>
<th>wm</th>
<th>mm</th>
<th>hm</th>
</tr>
</thead>
<tbody>
<tr>
<td>base</td>
<td>90.39</td>
<td>72.37</td>
<td>31.06</td>
<td>86.85</td>
<td>73.74</td>
<td>42.23</td>
<td>90.58</td>
<td>79.06</td>
<td>74.28</td>
<td>77.80</td>
<td>47.40</td>
<td>31.90</td>
</tr>
<tr>
<td>P+e+L</td>
<td>87.11</td>
<td>65.46</td>
<td>76.71</td>
<td>90.34</td>
<td>74.53</td>
<td>71.94</td>
<td>89.86</td>
<td>81.55</td>
<td>82.89</td>
<td>79.87</td>
<td>52.4</td>
<td>52.9</td>
</tr>
<tr>
<td>Improv.</td>
<td>-34.13%</td>
<td>-25.01%</td>
<td>+66.22%</td>
<td>+26.54%</td>
<td>+3.01%</td>
<td>+51.43%</td>
<td>-7.64%</td>
<td>+11.89%</td>
<td>+33.48%</td>
<td>+9.32%</td>
<td>+9.51%</td>
<td>+30.84%</td>
</tr>
</tbody>
</table>

Table 4: P+E+L: Relative improvement on Aurora 2 CLEAN.

<table>
<thead>
<tr>
<th>Set</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clean</td>
<td>+1.29%</td>
<td>+1.29%</td>
<td>+4.80%</td>
<td>+2.31%</td>
</tr>
<tr>
<td>20 dB</td>
<td>-14.23%</td>
<td>-15.73%</td>
<td>+44.07%</td>
<td>-3.17%</td>
</tr>
<tr>
<td>15 dB</td>
<td>+15.85%</td>
<td>+21.87%</td>
<td>+55.28%</td>
<td>+26.14%</td>
</tr>
<tr>
<td>10 dB</td>
<td>+38.03%</td>
<td>+39.02%</td>
<td>+52.63%</td>
<td>+41.35%</td>
</tr>
<tr>
<td>5 dB</td>
<td>+36.55%</td>
<td>+33.53%</td>
<td>+48.45%</td>
<td>+37.72%</td>
</tr>
<tr>
<td>0 dB</td>
<td>+22.87%</td>
<td>+17.87%</td>
<td>+35.63%</td>
<td>+23.42%</td>
</tr>
<tr>
<td>-5dB</td>
<td>+7.90%</td>
<td>+1.65%</td>
<td>+17.93%</td>
<td>+7.41%</td>
</tr>
<tr>
<td>Average</td>
<td>+26.58%</td>
<td>+24.92%</td>
<td>+43.82%</td>
<td>+28.77%</td>
</tr>
</tbody>
</table>

Table 5: P+E+L: Relative improvement on Aurora 2 MULTI.

<table>
<thead>
<tr>
<th>Set</th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clean</td>
<td>-20.35%</td>
<td>-20.35%</td>
<td>-17.84%</td>
<td>-19.85%</td>
</tr>
<tr>
<td>20 dB</td>
<td>-25.44%</td>
<td>-68.81%</td>
<td>+16.89%</td>
<td>-34.32%</td>
</tr>
<tr>
<td>15 dB</td>
<td>-36.06%</td>
<td>-35.54%</td>
<td>+17.01%</td>
<td>-25.24%</td>
</tr>
<tr>
<td>10 dB</td>
<td>-28.63%</td>
<td>-25.44%</td>
<td>+11.85%</td>
<td>-19.26%</td>
</tr>
<tr>
<td>5 dB</td>
<td>+2.88%</td>
<td>+4.59%</td>
<td>+29.47%</td>
<td>+8.88%</td>
</tr>
<tr>
<td>0 dB</td>
<td>+23.07%</td>
<td>+21.94%</td>
<td>+38.91%</td>
<td>+25.78%</td>
</tr>
<tr>
<td>-5dB</td>
<td>+17.79%</td>
<td>+17.68%</td>
<td>+19.94%</td>
<td>+18.18%</td>
</tr>
<tr>
<td>Average</td>
<td>+11.03%</td>
<td>+5.86%</td>
<td>+32.82%</td>
<td>+14.15%</td>
</tr>
</tbody>
</table>

5. Conclusions

In this work a powerful front-end tool CutCopy designed for feature extraction and for enhancement of speech was presented. The algorithm steps described above were implemented in C++ so that there is maximum flexibility in their interconnection, optimal algorithm sharing and also transparency in code for the ease of adding new methods for preprocessing, signal enhancement, feature extraction and post-processing. The whole software package with complete documentation and source code is available on the internet address http://noel.feld.cvut.cz/speechlab.

There were several techniques for additive-noise and channel-mismatch removal and also for parametrization mentioned and implemented in the front-end. For the purpose of completing the knowledge about the features of extended spectral subtraction algorithm there were done a number of comparative experiments with SpeechDat-E, and Aurora 2&3 databases. A complete front-end combining extended spectral subtraction and LDA RASTA-like filtration has been tested.

All the experiments were evaluated in comparison to the WI007 front-end for the reason that the end-pointing mechanism is not yet completely implemented and the results would not be comparable to WI008. The new comparable results will be presented at the poster.

The experiments with extended spectral subtraction have shown the reasonable improvement of recognition accuracy in all conditions and SNRs. Since this method does not require a voice-activity detector and allows the setting of suppression threshold, it is suitable for the improvement the performance in general noisy environment and for speech enhancement.

The LDA RASTA-like filtration decreases the recognition accuracy when trained on multicondition data. On the other hand, it was shown that it can be successfully combined with EXTEN method yielding noise and channel-robust system that can be employed in high-mismatched conditions.

6. Acknowledgements

This paper is the part of the research supported by grant “Voice Technologies for Support of Information Society” with No. GACR-102/02/0124 (2002-2004).

7. References


