Post-Processing of Feature for Noise Robust Speech Recognition

Guanghu Shen, Ho-Youl Jung, and Hyun-Yeol Chung, Member, IEEE

Abstract—We present an effective and simple noise robust front-end based on post-processing of standard mel-frequency cepstral coefficients (MFCCs) features. A feature processing method consisting of silence energy normalization, cepstral mean and variance normalization, and Auto-Regression and Moving-Average (ARMA) filtering in both log-energy and cepstral domain for noise-robust feature extraction is proposed. This technique is named SEN-MVN-A processing. From the experiments conducted on Aurora 2.0 database, we showed that SEN-MVN-A provides an averaged improvement of word recognition accuracy of 15.7%, 21.5% and 6.4% for test sets A, B and C, respectively, when compared with the baseline results.

Index Terms—Silence energy normalization, cepstral mean normalization, cepstral mean and variance normalization, ARMA filter, robust speech recognition

I. INTRODUCTION

In real world speech recognition applications, robust features are highly desired in order to offer acceptable recognition performance under various noisy conditions. Mel-frequency cepstral coefficients (MFCCs) have been well accepted as a good choice for speech features with reasonable robustness, and many advanced techniques have been developed based on them. Normalizing the MFCCs parameters has been a well-known approach to improve the robustness of the feature parameters. Cepstral mean normalization (CMN) [1] is also well-known method to reduce the environmental distortion. It assumes that the mean of the cepstral coefficients is invariant for various utterances. Therefore, there is no relevant information in the mean, and subtracting it reduces only irrelevant information. In CMN, the irrelevant information is assumed to be convolutional channel noise or spectral tilt. In some cases, such a strong assumption may cause a loss of relevant information, but the greater reduction of irrelevant information in adverse conditions generally results in better performance. A natural extension of CMN is cepstral mean and variance normalization (MVN) [2], where the assumption is still stronger. In MVN, the mean and the variance of the cepstral coefficients of clean speech are assumed to be invariant. Therefore, removing mean and variance is assumed to reduce only irrelevant information, no matter what that information may be.

In clean speech the spikes might contain important information about the speech utterance, while in noisy speech these spikes are more likely to be caused by noise. To smooth out the spikes a feature processing technique consisting of mean and variance normalization, and Auto-Regression and Moving-Average (ARMA) filtering has been proposed. This method called MVA [3], [4] proved to be remarkably beneficial on system performance, due to the emphasis in the low-frequency part and de-emphasis in the high-frequency part of the ARMA filter.

As is well known, the energy parameter has been widely used as an extension to the basic features of MFCCs to improve the recognition accuracy in speech recognition. However, these energy features are often vulnerable to noise and thus their discriminating capability is limited. Recently, some methods have been proposed to enhance these energy features. Silence energy normalization (SEN) [5] which is one of them used the high-pass filtered log-energy as the feature for speech/non-speech classification, and then the log-energy of non-speech frames is set to be a small constant while that of speech frames is kept unchanged.

And one method called SEN-MVN [5] which is integrated SEN with cepstral mean and variance normalization (MVN) was presented to obtain further improved performance, since they are performed on different features.

To make the overall frequency plots of clean and noisy samples quite similar, we would like to propose one new method based on SEN-MVN with additional ARMA filter, which is called as SEN-MVN-A. We can believe that it will prove best performance. Actually, there is an inherent trade-off in choosing the order M of the ARMA filter, because the idea of smoothing out a spiky time sequence quite natural.

The remainder of the paper is organized as follows. In section II, we will introduce our post-processing of feature technique which consisting of SEN, CMN, MVN, ARMA filter, and the proposed algorithm i.e. SEN-MVN-A. In section III experimental results comprising with the baseline system defined in [6], and conclusions are given in the last section.

II. POST-PROCESSING OF FEATURE

The speech signal produced by speaker is transmitted over some channel before it reaches the recording device, and these channels easily disturb the original speech signal. Finally, the convolutional distortions will be multiplicative in the spectrum domain. Due to the logarithmic compression of the filter bank
channels before the cosine transformation, the distortions in the form of multiplicative become to ones in the form of additive in the cepstrum domain [7].

Thus a simple and effective way of channel normalization is to subtract the mean of each cepstral coefficient (CMN) which will remove the time-invariant distortions introduced by the transmission channel and the recording device. Furthermore it is known that normalizing the variance of cepstral coefficients (MVN) helps to improve recognition in adverse conditions.

A. Cepstral Mean and Variance Normalization

We start with standard MFCCs, c(1)…c(12), along with their delta and delta-delta as our raw features. For a given utterance, we represent the data by a matrix C whose element

\[ C_n[d] \]

is the \( d \)th component of the feature vector at time \( n \), \( 1 = ... N \), the number of frames in the utterance and \( d = 1 ... D \), the dimension of the feature space. In other words, each row of \( C \) represents a feature vector and each column represents a time sequence. The first step is standard cepstral mean normalization (CMN) defined by:

\[
C_n'[d] = C_n[d] - \mu[d]
\]

where

\[
\mu[d] = \frac{1}{N} \sum_{n=1}^{N} C_n[d]
\]

Variance normalization (VN) is defined by:

\[
\overline{C}_n[d] = C_n'[d] / \sigma[d] = (C_n[d] - \mu_n[d]) / \sigma[d]
\]

\[
\sigma[d] = \sqrt{\left(\sum_{n=1}^{N} (C_n[d] - \mu[d])^2 \right) / N}
\]

where \( \overline{C} \) is the mean and variance normalized feature and \( \sigma[d] \) is an estimated variance of the \( d \)th feature component.

B. Silence Energy Normalization

When observing the log-energy contour of a clean utterance and its noise-corrupted counterpart, we found that the high-energy speech portions are relatively less influenced by noise and sometimes keep the ripple characters. On the other hand, the low-energy non-speech portions of the clean utterance are relatively “flat” in the contour, and they much more vulnerable to noise since their log-energy levels are significantly elevated. Furthermore, the “flatness” of the non-speech portion is kept and sometimes further enhanced by the effect of noise. Silence energy normalization (SEN) which was introduced in [5] handles well the problem of non-speech portion while speech portions keep unchanged.

The procedure of SEN is consisted by two steps, the first step is to classify each frame as speech or non-speech (silence), and the second is to normalize the log-energy of each silence frame to be a small constant.

Here, we use a simple IIR high-pass filter different from the delta filter in [8], and its input-output relationship is

\[
e_n' = \frac{1}{2} (e_{n+1} - e_{n-1})
\]

where \( e_n \) is the log-energy of the \( n \)th frame and \( e_n' \) is the corresponding output of filter.

Next, according to the output of filter \( e_n' \), the normalized log-energy \( \overline{e}_n \) for the \( n \)th frame can be obtained by the following equation.

\[
\overline{e}_n = \begin{cases} e_n & \text{if } e_n' > T \\ \epsilon & \text{if } e_n' \leq T \end{cases}
\]

where \( T \) is the threshold and \( \epsilon \) is a small constant. That is, if \( e_n' \) is smaller than the threshold \( T \), then the \( n \)th frame is classified as silence and its log-energy is normalized to be \( \epsilon \).

Otherwise, the \( n \)th frame is classified as speech and its log-energy is unchanged. Here the threshold \( T \) is set in an utterance-wise manner, and it equals to the average of \( e_n' \) in an utterance. That is

\[
T = \frac{1}{N} \sum_{n=1}^{N} e_n'
\]

where \( N \) is the number of total frames in an utterance.

Fig. 1 shows the SEN-processed log-energy contour of the clean utterance, and the log-energy of silence portions are normalized as a small value. (here, \( \epsilon = 1 \).)

C. ARMA Filtering

Auto-Regression Moving Average (ARMA) filter, defined by:

\[
\tilde{C}_n[d] = \begin{cases} 
\sum_{i=1}^{M} \tilde{C}_{n-i}[d] + \sum_{j=0}^{M} \tilde{C}_{n+j}[d] \overline{C}_n[d] \\ (2M + 1) \overline{C}_n[d] 
\end{cases} \quad \text{if } M < n \leq N - M \\
\overline{C}_n[d] \quad \text{otherwise}
\]

(8)

where \( M \) is the order of the ARMA filter. The special case of \( M = 0 \) degenerates to no ARMA filtering.

In order to analyze the relationship between \( \overline{C} \) and \( \tilde{C} \) in the frequency domain, we rewrite the equation (8) as:
(2M + 1)\tilde{C}_s[d] - \ldots - \tilde{C}_{n-M} = \tilde{C}_s[d] + \ldots + \tilde{C}_{n-M} \quad (9)

From the equation (9), the transfer function is:

\[ H(z) = \frac{1 + z + \ldots + z^M}{2M + 1 - z^{-1} - \ldots - z^{-M}} \quad (10) \]

The frequency response of the ARMA filter of order \( M \) is:

\[ H(e^{jw}) = \frac{1 + e^{jw} + \ldots + e^{jMw}}{2M + 1 - e^{-jw} - \ldots - e^{-jMw}} \]

\[ = \frac{1 - e^{j(M+1)w}}{2M + 2 - (2M + 1)e^{jw} - e^{-jMw}} \quad (11) \]

Note that for \( \omega = 0 \), \( H(e^{jw}) = 1 \). The frequency responses of the cases \( M = 2, 4 \) are plotted in Fig. 2.

\[ \text{Fig. 2. Gains and phase shifts of the ARMA filters.} \]

D. SEN-MVN-A

Through analysis of upper several algorithms, we can find that the performance is not enough when we do processing by using upper algorithms, respectively. From the references we also can know that some integrated algorithms i.e. SEN-MVN [5], MVA (MVN-ARMA) [2],[3] have been proposed, recently. And these algorithms showed good recognition results.

In this paper, to solve the mismatch between training and test data for automatic speech recognition systems, we would like to propose one new algorithm based on SEN-MVN with additional ARMA filter, which is called as SEN-MVN-A.

As we know, ARMA filter is one kind of low-pass filter which is used to smooth out the sequences toward temporal similitude, and emphasis in the low-frequency part and de-emphasis in the high-frequency part of features which is consisted of log-energy and cepstral coefficients. It means that we do normalization processing for log-energy with SEN, for cepstral coefficients with MVN, then do filtering for both processed features with ARMA filter.

In Fig. 4, the time sequences of log-energy and C[1] are plotted for the speech signals of the utterance of digit string 52109 corrupted by different levels of additive subway noise (from the Aurora 2.0 database). In the log-energy (left column), we can see that SEN mainly deals with non-speech portions while keeping speech portions unaltered. And we also find that the accuracy of SEN was decreased as decreasing of SNR, while mismatching between noisy speech and clean speech became smaller after adding ARMA filter processing. In the cepstral coefficients (right column), MVN-A processed features are not only reduced the temporal variation but also smoothed the sequences. Especially, MVN-A processed features makes the overall frequency plots of clean and noisy speech quite similar because of character of ARMA filter.

The implementation of our proposed algorithm i.e. SEN-MVN-A in automatic speech recognition system was shown in Fig. 3.

\[ \text{Fig. 3. Automatic Speech Recognition System with proposed SEN-MVN-A. FE: feature extraction; SEN: silence energy normalization; MVN: mean and variance normalization; A: ARMA filter; SEN-MVN-A: proposed algorithm based on SEN-MVN with additional ARMA filter} \]

III. EXPERIMENTAL RESULTS

A. Experimental Setup

The proposed system was evaluated on Aurora 2.0 database [6]. Two training conditions (clean condition and multi condition) and three test set (sets A, B and C) were defined by Aurora 2.0. The test set A included four different type of noises (subway, babble, car and exhibition), while the test set B included another four different types of noises (restaurant, street, airport and train station). The test set C then included two noise types respectively from sets A and B (subway and street), plus additional convolutional noise. Six different SNR values, ranging from 20dB to -5dB, were tested in each case.

In this experiment, speech signal with pre-emphasis was window of length 25 ms with 10 ms frame period. The frequency warping factor was set to 0.4. The HMM was trained on clean condition with 16 states per word and a mixture of 6 Gaussians per state. As front-end, each utterance was first converted into a stream of 12 mel-frequency cepstral coefficients (MFCCs) and log-energy, plus theirs delta and delta-delta were the components in the final used 39-dimensional feature vectors.

B. Recognition Results

We compared the performance only in clean training condition, and averaged word accuracy (as percentages) over clean-0dB SNR test data was shown in Table 1.
<table>
<thead>
<tr>
<th>Set A</th>
<th>Baseline</th>
<th>SEN</th>
<th>CMN</th>
<th>SEN-CMN</th>
<th>SEN-CMN-A(2)</th>
<th>SEN-CMN-A(4)</th>
<th>MVN</th>
<th>SEN-MVN</th>
<th>SEN-MVN-A(2)</th>
<th>SEN-MVN-A(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clean</td>
<td>98.94</td>
<td>98.70</td>
<td>98.98</td>
<td>98.77</td>
<td>98.85</td>
<td>98.76</td>
<td>98.98</td>
<td>98.79</td>
<td>98.76</td>
<td>98.64</td>
</tr>
<tr>
<td>20dB</td>
<td>94.99</td>
<td>96.73</td>
<td>96.90</td>
<td>96.17</td>
<td>96.50</td>
<td>96.43</td>
<td>95.98</td>
<td>95.67</td>
<td>96.25</td>
<td>96.76</td>
</tr>
<tr>
<td>15dB</td>
<td>86.93</td>
<td>93.26</td>
<td>92.30</td>
<td>92.02</td>
<td>93.05</td>
<td>92.65</td>
<td>91.65</td>
<td>93.13</td>
<td>94.31</td>
<td>93.04</td>
</tr>
<tr>
<td>10dB</td>
<td>67.28</td>
<td>83.65</td>
<td>76.15</td>
<td>83.07</td>
<td>84.66</td>
<td>82.57</td>
<td>80.43</td>
<td>86.45</td>
<td>85.09</td>
<td></td>
</tr>
<tr>
<td>5dB</td>
<td>39.36</td>
<td>61.17</td>
<td>43.37</td>
<td>64.08</td>
<td>68.60</td>
<td>61.53</td>
<td>57.47</td>
<td>71.42</td>
<td>69.53</td>
<td></td>
</tr>
<tr>
<td>0dB</td>
<td>17.07</td>
<td>29.01</td>
<td>22.05</td>
<td>34.51</td>
<td>41.01</td>
<td>30.69</td>
<td>26.62</td>
<td>42.43</td>
<td>46.06</td>
<td></td>
</tr>
<tr>
<td>average</td>
<td>67.43</td>
<td>77.09</td>
<td>71.62</td>
<td>78.10</td>
<td>80.44</td>
<td>77.11</td>
<td>75.19</td>
<td>81.41</td>
<td>83.16</td>
<td>80.76</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Set B</th>
<th>Baseline</th>
<th>SEN</th>
<th>CMN</th>
<th>SEN-CMN</th>
<th>SEN-CMN-A(2)</th>
<th>SEN-CMN-A(4)</th>
<th>MVN</th>
<th>SEN-MVN</th>
<th>SEN-MVN-A(2)</th>
<th>SEN-MVN-A(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clean</td>
<td>98.94</td>
<td>98.70</td>
<td>98.82</td>
<td>98.77</td>
<td>98.85</td>
<td>98.76</td>
<td>98.98</td>
<td>98.79</td>
<td>98.76</td>
<td>98.64</td>
</tr>
<tr>
<td>20dB</td>
<td>92.35</td>
<td>96.81</td>
<td>94.58</td>
<td>96.74</td>
<td>96.98</td>
<td>96.77</td>
<td>96.37</td>
<td>96.67</td>
<td>96.93</td>
<td>96.43</td>
</tr>
<tr>
<td>15dB</td>
<td>80.79</td>
<td>94.06</td>
<td>87.42</td>
<td>93.43</td>
<td>94.29</td>
<td>93.73</td>
<td>92.07</td>
<td>93.99</td>
<td>94.52</td>
<td>93.01</td>
</tr>
<tr>
<td>10dB</td>
<td>58.06</td>
<td>85.46</td>
<td>71.76</td>
<td>85.03</td>
<td>87.01</td>
<td>84.00</td>
<td>81.90</td>
<td>87.08</td>
<td>88.77</td>
<td>94.80</td>
</tr>
<tr>
<td>5dB</td>
<td>32.03</td>
<td>64.07</td>
<td>48.05</td>
<td>65.93</td>
<td>70.61</td>
<td>62.89</td>
<td>58.64</td>
<td>71.30</td>
<td>74.03</td>
<td>68.48</td>
</tr>
<tr>
<td>0dB</td>
<td>14.62</td>
<td>31.41</td>
<td>23.01</td>
<td>36.71</td>
<td>42.03</td>
<td>32.31</td>
<td>26.72</td>
<td>42.32</td>
<td>46.56</td>
<td>41.81</td>
</tr>
<tr>
<td>average</td>
<td>62.80</td>
<td>78.42</td>
<td>70.61</td>
<td>79.44</td>
<td>81.63</td>
<td>78.07</td>
<td>75.78</td>
<td>81.69</td>
<td>83.33</td>
<td>80.53</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Set C</th>
<th>Baseline</th>
<th>SEN</th>
<th>CMN</th>
<th>SEN-CMN</th>
<th>SEN-CMN-A(2)</th>
<th>SEN-CMN-A(4)</th>
<th>MVN</th>
<th>SEN-MVN</th>
<th>SEN-MVN-A(2)</th>
<th>SEN-MVN-A(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clean</td>
<td>99.00</td>
<td>98.74</td>
<td>99.09</td>
<td>98.84</td>
<td>98.89</td>
<td>98.89</td>
<td>99.10</td>
<td>99.00</td>
<td>98.80</td>
<td>98.71</td>
</tr>
<tr>
<td>20dB</td>
<td>94.83</td>
<td>94.00</td>
<td>96.59</td>
<td>95.14</td>
<td>95.80</td>
<td>95.57</td>
<td>95.45</td>
<td>95.31</td>
<td>96.07</td>
<td>95.57</td>
</tr>
<tr>
<td>15dB</td>
<td>88.66</td>
<td>86.75</td>
<td>90.93</td>
<td>90.01</td>
<td>91.73</td>
<td>90.35</td>
<td>88.69</td>
<td>91.05</td>
<td>92.67</td>
<td>90.54</td>
</tr>
<tr>
<td>10dB</td>
<td>75.23</td>
<td>69.44</td>
<td>73.42</td>
<td>77.86</td>
<td>81.86</td>
<td>75.10</td>
<td>74.23</td>
<td>81.51</td>
<td>83.48</td>
<td>80.14</td>
</tr>
<tr>
<td>5dB</td>
<td>50.85</td>
<td>43.55</td>
<td>41.61</td>
<td>53.87</td>
<td>60.20</td>
<td>47.87</td>
<td>51.14</td>
<td>63.03</td>
<td>63.98</td>
<td>58.90</td>
</tr>
<tr>
<td>0dB</td>
<td>23.83</td>
<td>20.51</td>
<td>22.16</td>
<td>27.22</td>
<td>34.14</td>
<td>25.79</td>
<td>24.23</td>
<td>35.46</td>
<td>35.99</td>
<td>33.16</td>
</tr>
<tr>
<td>average</td>
<td>72.06</td>
<td>68.83</td>
<td>70.63</td>
<td>73.82</td>
<td>77.10</td>
<td>72.26</td>
<td>72.14</td>
<td>77.53</td>
<td>78.50</td>
<td>76.17</td>
</tr>
</tbody>
</table>

Fig. 4. The time sequences of the log-energy and cepstral coefficient $C(1)$ for the digital string $52109$ corrupted with different levels of noises after each post-processing, respectively. In SEN-A and MVN-A cases, the order of ARMA filter is 2.

The experimental notations for Table 1 are as following.

1) **SEN**: silence energy normalization
2) **CMN**: cepstral mean normalization
3) **MVN**: cepstral mean and variance normalization
4) **SEN-CMN**: integration of SEN and CMN
5) **SEN-CMN-A(M)**: ARMA filtering after SEN-CMN ($M$ is order of ARMA filter)
6) **SEN-MVN**: integration of SEN and MVN
7) **SEN-MVN-A(M)**: ARMA filtering after SEN-MVN ($M$ is order of ARMA filter)

From the recognition results in test set A and B, we can see that SEN has significant improvements of recognition accuracy for both stationary and non-stationary noise conditions. Especially, SEN gives 9.7% and 15.6% of word accuracy improvements comparing with baseline results, 6.6% and 2.3% of average word accuracy improvements comparing with CMN and MVN, respectively. We also can see that the integrated algorithms i.e. SEN-CMN and SEN-MVN have better recognition results than those which are not combined, respectively. We consider that is because of normalization for both log-energy and cepstral coefficients in speech features. In case of additional ARMA filter based on upper integrated algorithms, there is improvement of performance in all environments comparing with the case without ARMA filter. Especially, we obtained an average improvement of 1.7% over SEN-MVN-A(2) comparing with SEN-MVN, 2.3% over...
SEN-CMN-A(2) comparing with SEN-CMN. However, the recognition results are decreased, when the order of ARMA filter is increased from 2 to 4. It is not means that more high order of ARMA filter is better than lower one.

From the recognition results for test set C, there is not much effect over SEN, CMN and MVN. Because convolutional noise is existed in this test data, and these algorithms are not suitable for this kind of noise. However, the integrated algorithms showed some improvement recognition results comparing with baseline and those which are not combined, respectively. Especially, we obtained an average improvement of 1.0% over SEN-MVN-A(2) comparing with SEN-MVN, 3.3% over SEN-CMN-A(2) comparing with SEN-CMN.

IV. CONCLUSION

In this paper, we proposed a post-processing of feature technique which is consisted of silence energy normalization for log-energy, mean and variance normalization for cepstral coefficients, and Auto-Regression and Moving-Average (ARMA) filtering for both log-energy and cepstrum domain. This technique is named SEN-MVN-A processing. From the experiments conducted on Aurora 2.0 database, we showed that SEN-MVN-A provides an averaged improvement of word recognition accuracy of 15.7%, 21.5% and 6.4% for test sets A, B and C, respectively, when compared with the baseline results.

REFERENCES