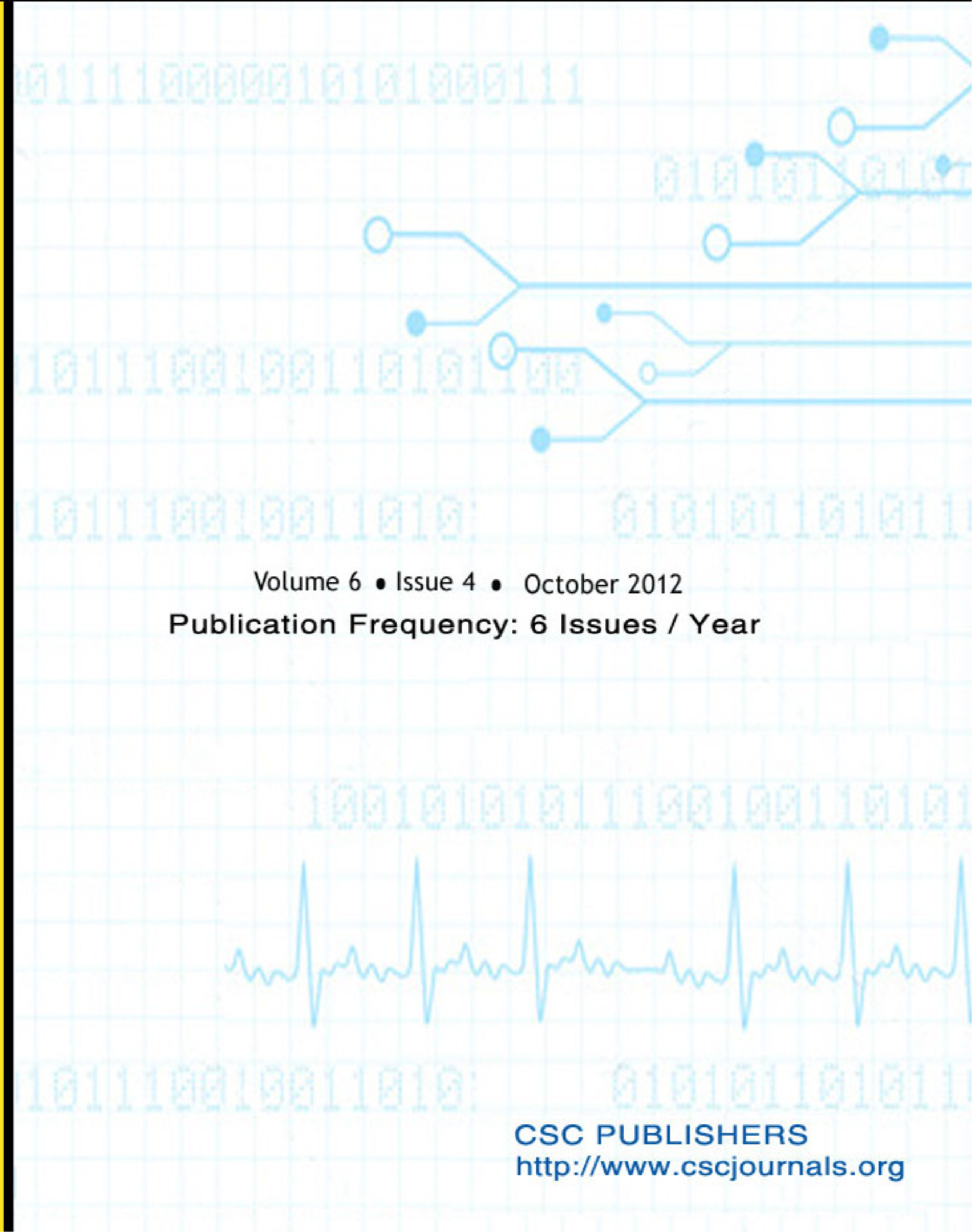
Editor-in-Chief
Dr. Saif alZahir

SIGNAL PROCESSING (SPIJ)

AN INTERNATIONAL JOURNAL

ISSN : 1985-2339



Volume 6 • Issue 4 • October 2012
Publication Frequency: 6 Issues / Year

CSC PUBLISHERS
http://www.cscjournals.org

SIGNAL PROCESSING: AN INTERNATIONAL JOURNAL (SPIJ)

VOLUME 6, ISSUE 4, 2012

EDITED BY
DR. NABEEL TAHIR

ISSN (Online): 1985-2339

International Journal of Computer Science and Security is published both in traditional paper form and in Internet. This journal is published at the website http://www.cscjournals.org, maintained by Computer Science Journals (CSC Journals), Malaysia.

SPIJ Journal is a part of CSC Publishers
Computer Science Journals
http://www.cscjournals.org

SIGNAL PROCESSING: AN INTERNATIONAL JOURNAL (SPIJ)

Book: Volume 6, Issue 4, October 2012
Publishing Date: 24-10-2012
ISSN (Online): 1985-2339

SPIJ Journal is a part of CSC Publishers
http://www.cscjournals.org

Typesetting: Camera-ready by author, data conversation by CSC Publishing Services – CSC Journals, Malaysia

CSC Publishers, 2012

EDITORIAL PREFACE

This is *Fourth* Issue of Volume *Six* of the Signal Processing: An International Journal (SPIJ). SPIJ is an International refereed journal for publication of current research in signal processing technologies. SPIJ publishes research papers dealing primarily with the technological aspects of signal processing (analogue and digital) in new and emerging technologies. Publications of SPIJ are beneficial for researchers, academics, scholars, advanced students, practitioners, and those seeking an update on current experience, state of the art research theories and future prospects in relation to computer science in general but specific to computer security studies. Some important topics covers by SPIJ are Signal Filtering, Signal Processing Systems, Signal Processing Technology and Signal Theory etc.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Started with Volume 6, 2012, SPIJ appears with more focused issues related to signal processing studies. Besides normal publications, SPIJ intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

This journal publishes new dissertations and state of the art research to target its readership that not only includes researchers, industrialists and scientist but also advanced students and practitioners. The aim of SPIJ is to publish research which is not only technically proficient, but contains innovation or information for our international readers. In order to position SPIJ as one of the top International journal in signal processing, a group of highly valuable and senior International scholars are serving its Editorial Board who ensures that each issue must publish qualitative research articles from International research communities relevant to signal processing fields.

SPIJ editors understand that how much it is important for authors and researchers to have their work published with a minimum delay after submission of their papers. They also strongly believe that the direct communication between the editors and authors are important for the welfare, quality and wellbeing of the Journal and its readers. Therefore, all activities from paper submission to paper publication are controlled through electronic systems that include electronic submission, editorial panel and review system that ensures rapid decision with least delays in the publication processes.

To build its international reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for SPIJ. We would like to remind you that the success of our journal depends directly on the number of quality articles submitted for review. Accordingly, we would like to request your participation by submitting quality manuscripts for review and encouraging your colleagues to submit quality manuscripts for review. One of the great benefits we can provide to our prospective authors is the mentoring nature of our review process. SPIJ provides authors with high quality, helpful reviews that are shaped to assist authors in improving their manuscripts.

Editorial Board Members
Signal Processing: An International Journal (SPIJ)

TABLE OF CONTENTS

Volume 6, Issue 4, October 2012

Pages

Real-time DSP Implementation of Audio Crosstalk Cancellation Using Mixed Uniform Partitioned Convolution

Chunduri SreenivasaRao *chsrinivas19800305@rediffmail.com*
Research Scholar, ECE Department,
KL University
Vijayawada, 522502, INDIA

NVK Mahalakshmi *mahalakshminvk.nvk@gmail.com*
Assistant Professor, ECE Department
SRK Institute of Technology,
Vijayawada, 521108, INDIA

Dr. Dhulipalla VenkataRao *dvenky221101@rediffmail.com*
Principal, Narasaraopet Instt. of Tech.
Narasaraopet, Guntur, 522601, INDIA

Abstract

For high fidelity sound reproduction, it is necessary to use long filter coefficients in audio crosstalk cancellation. To implement these long filters on real-time DSP processors, conventional overlap save technique suffers from more computational power as well as processing delay. To overcome these technical problems, *mixed uniform partitioned convolution* technique is proposed. This method is derived by combining uniform partitioned convolution with mixed filtering technique. With the proposed method, it is possible to perform audio crosstalk cancellation even at the order of ten thousand filter taps with less computations and short processing delay. The proposed technique was implemented on 32-bit floating point DSP processor and design was provided with efficient memory management to achieve optimization in computational complexity. The computational comparison of this method with conventional methods shows that the proposed technique is very efficient for long filters.

Keywords: Convolution, Crosstalk Cancellation, FFT, Mixed filtering, Partitioned Convolution, Overlap Save Method.

1. INTRODUCTION

3D audio systems have the potential to be used in many spatial audio applications such as home theatre entertainment, gaming, teleconference and remote control. To reproduce the realistic spatial audio, the challenging task of any 3D audio system is to have the ability to reproduce spatial reverberation characteristics and spatial audio pattern at the desired locations. This could be achieved by binaural synthesis and audio cross-talk cancellation (CTC). In 1983, head related transfer function (HRTF) technology was developed to transform the sound field of a particular location to the head by convolving the sound with appropriate pair of HRTF functions. Headphones have excellent spatial characteristics such as channel separation and equalization, but they are inconvenient and little bit cumbersome to use when more number of listeners is enjoying the audio. An alternative to HRTF technology is conventional stereo loudspeaker system located exactly in front of the listener. In this case, transmission path equalization is obtained by inverting acoustic transfer function matrix between the two loudspeakers and the two ears of listener, which is called crosstalk cancellation and is particularly required to cancel the unwanted crosstalk from each speaker to the opposite ear [1][2][3][4].

To obtain such equalization for transmission path, the impulse responses of 2x2 system inversion matrix (*$h_{aa}(n)$, $h_{ab}(n)$, $h_{ba}(n)$, $h_{bb}(n)$* as shown in Fig.1) may last for several hundreds of milliseconds, which leads to the requirement of thousands of FIR filter coefficients as impulse responses [5]. Due to this, the implementation of these long filters on real-time DSP processors requires more computational power. To overcome the complexity issues, it is essential to develop new implementation techniques without compromising for performance.

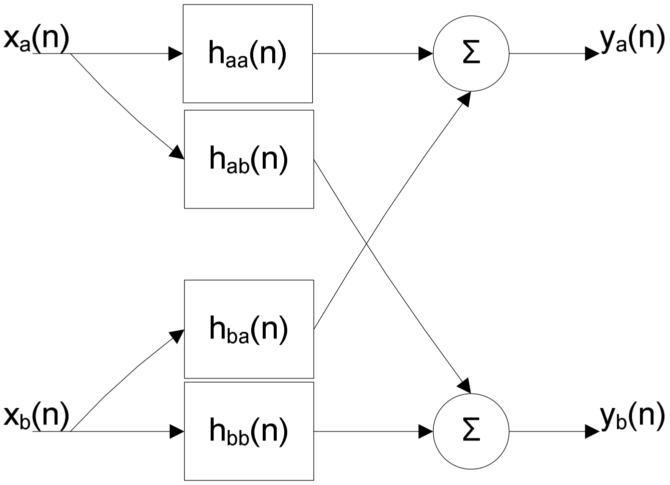


FIGURE 1: Audio Crosstalk Cancellation (CTC) for stereo source.

Historically, time domain convolution is well known technique. Even though this method is the original method, it won't be preferred for long filters, in general, as it suffers from more computational power. On other hand, overlap save & overlap add methods are frequency domain methods and are efficient to handle the computational complexity problems in real-time implementation. In these methods, the length of FFT is derived as N = L+M-1, which must be a power of 2 due to the usage of FFT, where L and M are frame size and filter length respectively. For the case of L=256 & M=8192, N becomes 8447 and could be chosen as 16384 by adding additional zeros. Due to additional zeros, FFT size increases and hence, the increase in computational power. In addition to this, the additional zeros cause the delay in output response at least by M [5][6][7].

To overcome delay issues, in 1988, Vetterli proposed running convolution based on multi-rate methods, in which impulse response is divided into bi-orthonormal filter banks continuously till minimum FFT Size reaches such as equivalent to frame size, L. After bi-orthonormal filtering process, required interpolation techniques will be applied to obtain the final filtered signal. In this, delay issue was solved but it suffers from computational complexity as filtering process could be performed for every sub filter bank and this technique involves more buffering of data[8][9].

Uniform partitioned convolution is a kind of technique where computational complexity as well as delay issues is resolved. If this technique is applied individually to each filter of Fig.1, the implementation complexity is huge and internal DSP memory may not hold all required buffers, particularly for long filters [10][11][12][13][14][15][16][17].

To avoid such problems, uniform partitioned convolution is combined with mixed filtering in this paper and presented as a new proposed algorithm to reduce computational complexity as well as processing delay. With efficient memory management and the properties of FFT, the proposed technique is very good choice for audio CTC for long filters.

This paper is organized as follows. Section 2 provides the review of mixed filtering and uniform partitioned convolution. Later the combination of these two techniques is explained as proposed method in section 3. It also discusses theoretical computational complexity, design to achieve efficient memory management and optimization techniques. Section 4 details about the experimental details and results. The computational complexity of proposed method is compared with that of overlap save method. Finally chapter 5 provides the conclusion and future scope to update the proposed method.

2. REVIEW OF PREVIOUS WORK

In this section, Mixed Filtering and uniform partitioned convolution methods are reviewed.

2.1 Mixed Filtering

The name implies that this method is able to perform all filtering operations of CTC in a single equation. This is possible by forming a complex sequence with real-time outputs $y_a(n)$, $y_b(n)$. By doing so, one could arrive at the frequency domain equivalent of output complex sequence as

$$Y(k) = Y_a(k) + jY_b(k) = X_a(k)H_a(k) + X_b(k)H_b(k) \qquad \rightarrow \qquad (1)$$

where

$$H_a(k) = H_{aa}(k) + j\,H_{ab}(k)$$
$$H_b(k) = H_{ba}(k) + j\,H_{bb}(k)$$

The computational complexity of equation (1) includes one FFT computation with decomposition (to evaluate $X_a(k)$ *and* $X_b(k)$), complex frequency multiplication and one IFFT. The real and imaginary components of IFFT output yield $y_a(n)$ and $y_b(n)$ respectively [5].

2.2 Uniform Partitioned Convolution

In this method, the length of impulse response is uniformly partitioned into small lengths so that overlap save method is applied to each partitioned impulse response and finally adding all outputs of partitioned filters yield the convolved output. Fig. 2 shows the signal processing involved in this method [10][11].

Let $h(n)$ of length M be the impulse response and frame length be L. Fig. 2A shows the time delay line filter. Let $h_0(n)$, $h_1(n)$,..., $h_{m-1}(n)$ where $m = M/L$ be the partitioned impulse responses, which are obtained by dividing the impulse response length by L so that the length of each partitioned impulse response becomes L. Fig. 2B shows the application of overlap save method to each partitioned impulse response, where FFT/IFFT size is equal to 2L. After first frame is processed, L samples of IFFT output are transmitted as filtered output. For 2nd frame, first frame will be delayed by L samples and provided as input to 2nd partitioned filter. Now overlap save method is applied to both partitioned filters and IFFT outputs are summed to yield filtered output of 2nd frame. This process is continued till last partitioned filtering process. Instead of finding FFT and IFFT for time delayed frames and frequency multiplied outputs, it is better to optimize the structure with single FFT and IFFT as shown in Fig. 2C just by delaying the FFT outputs. When 2nd frame arrives, FFT output of 1st frame becomes the input 2nd partitioned filter. The complex outputs of all partitioned frequency multipliers are added and a single IFFT is applied to the complex sum. From the IFFT output, L samples are transmitted as overall filtered output.

FFT size of 2L means that the appended zero samples are L in size so that the processing delay is L samples in worst case whereas overlap save method for original impulse response produces at least M samples delay. Hence this method provides less delay compared to that of overlap save method.

Computational complexity of this method is explained as follows. For each frame, one FFT and one IFFT of size 2L are required. The frequency multiplier length is 2L. Such frequency multipliers are m = M/L and hence complex multiplications of *2L*.M/L = 2M are needed. All frequency multipliers have to be added before providing as input to IFFT and hence 2L (m-1) = 2(M - L) complex additions are required.

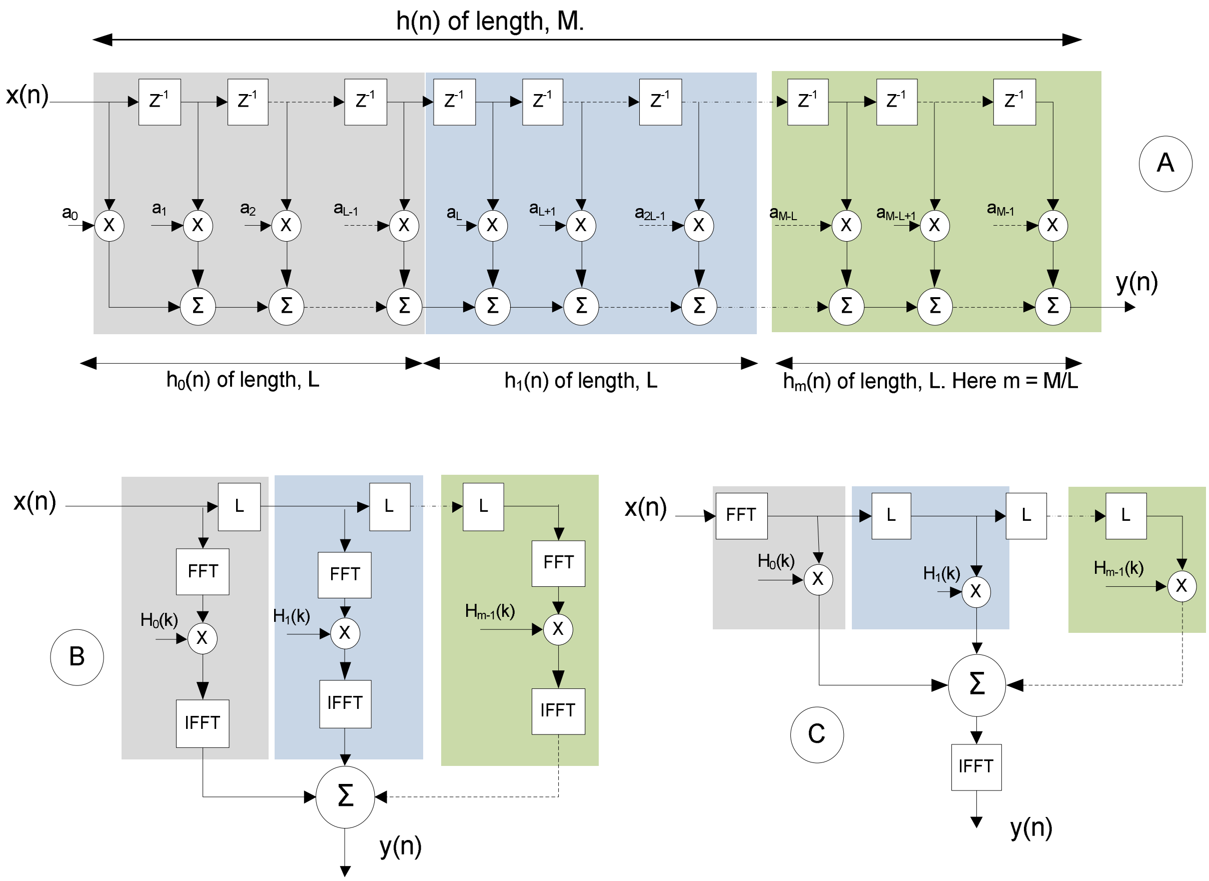


FIGURE 2: Interpretation of Uniform Partitioned Convolution using block diagram representation.

3. PROPOSED ALGORITHM

The proposed algorithm combines both the methods mentioned in Section 2. To proceed for the proposed algorithm, let us partition the impulse responses, $h_a(n)$ & $h_b(n)$ of equation (1). The time domain equivalents of $H_a(k)$ & $H_b(k)$ are given by $h_{aa}(n) + j\ h_{ab}(n)$ & $h_{ba}(n)+j\ h_{bb}(n)$ respectively. The length of these impulse responses are M. By partitioning these into m parts, where m = M/L, the resultant partitioned impulse responses are given by

$$\begin{aligned} h_a(n) &= \{h_{a,0}(n),\ h_{a,1}(n),\ h_{a,2}(n),....,\ h_{a,m-1}(n)\} \\ &= \{h_{aa,0}(n)+j\,h_{ab,0}(n),\ h_{aa,1}(n)+j\,h_{ab,1}(n),\ ....,\ h_{aa,m-1}(n)+j\,h_{ab,m-1}(n)\} \end{aligned}$$

$$\begin{aligned} h_b(n) &= \{h_{b,0}(n),\ h_{b,1}(n),\ h_{b,2}(n),....,\ h_{b,m-1}(n)\} \\ &= \{h_{ba,0}(n)+j\,h_{bb,0}(n),\ h_{ba,1}(n)+j\,h_{bb,1}(n),\ ....,\ h_{ba,m-1}(n)+j\,h_{bb,m-1}(n)\} \end{aligned}$$

The length of each partitioned sequence now becomes L. As per overlap save method, FFTs of these partitioned responses found out by appending L zeros to each impulse response. The frequency equivalents, $H_{a,0}(k)$, $H_{a,1}(k)$,... $H_{a,m-1}(k)$ *and* $H_{b,0}(k)$, $H_{b,1}(k)$,... $H_{b,m-1}(k)$ are obtained in this way. Once partitioned FFT coefficients are found, the rest of the algorithm is based on the application of uniform partitioned convolution approach as per equation (1). Instead of using two IFFTs, it is better to apply single IFFT to the complex frequency sum, $[X_a(k)H_a(k)+X_b(k)H_b(k)]$. IFFT output provides the outputs $y_a(n)$ & $y_b(n)$ in complex form.

The z-domain equivalent of equation (1) is given by

$$
\begin{aligned}
Y(z) &= Y_a(z) + jY_b(z) = X_a(z)H_a(z) + X_b(z)H_b(z) \\
&= X_a(z)\left[H_{a,0}(z) + z^{-L}H_{a,1}(z) + \ldots + z^{-(m-1)L}H_{a,m-1}(z)\right] + \\
&\qquad X_b(z)\left[H_{b,0}(z) + z^{-L}H_{b,1}(z) + \ldots + z^{-(m-1)L}H_{b,m-1}(z)\right] \\
&= X_a(z)\sum_{i=0}^{m-1} z^{-iL}H_{a,i}(z) + X_b(z)\sum_{i=0}^{m-1} z^{-iL}H_{b,i}(z) \\
&= \sum_{i=0}^{m-1}\left[X_a(z)z^{-iL}\right]H_{a,i}(z) + \left[X_b(z)z^{-iL}\right]H_{b,i}(z) && \rightarrow (2)
\end{aligned}
$$

$$
y(n) = y_a(n) + j\,y_b(n) = z^{-1}\left\{\sum_{i=0}^{m-1}\left[X_a(z)z^{-iL}\right]H_{a,i}(z) + \left[X_b(z)z^{-iL}\right]H_{b,i}(z)\right\} \qquad \rightarrow (3)
$$

The block diagram of the proposed algorithm was shown in Fig. 3. The steps involved in proposed algorithm are as follows.

a. Partition the long impulse responses and find the complex frequency equivalents as stated above.
b. Receive the 1st frames of inputs $x_a(n)$ & $x_b(n)$ and store them in memory buffers of length 2L each.
   Note: Initially memory buffers contain zeros and filling of input frames into memory buffers is based on overlap save method.
c. Find out FFT of overlapped 1st frame and store complex FFT outputs, $X_a(k)$ & $X_b(k)$ in separate buffers.
   Note: Reference [7] could be followed to find $X_a(k)$ & $X_b(k)$.
d. Perform complex frequency multiplication between frequency partitioned coefficients and frequency delayed input frames. Each time one frequency multiplication is performed, the resultant complex output is added to previous multiplier output so that complex sum will be provides as input to IFFT.
e. Evaluate step (d) for both of inputs $x_a(n)$ & $x_b(n)$ and add the corresponding complex sums to yield $[X_a(k)H_a(k)+X_b(k)H_b(k)]$.
f. Now apply IFFT to the output in step (e) and transmit real & imaginary parts of complex IFFT output as $y_a(n)$ & $y_b(n)$ respectively.
   Note: As per overlap save method, only L valid samples will transmitted from IFFT output.
g. Repeat steps (b) to (f) for each new frame.

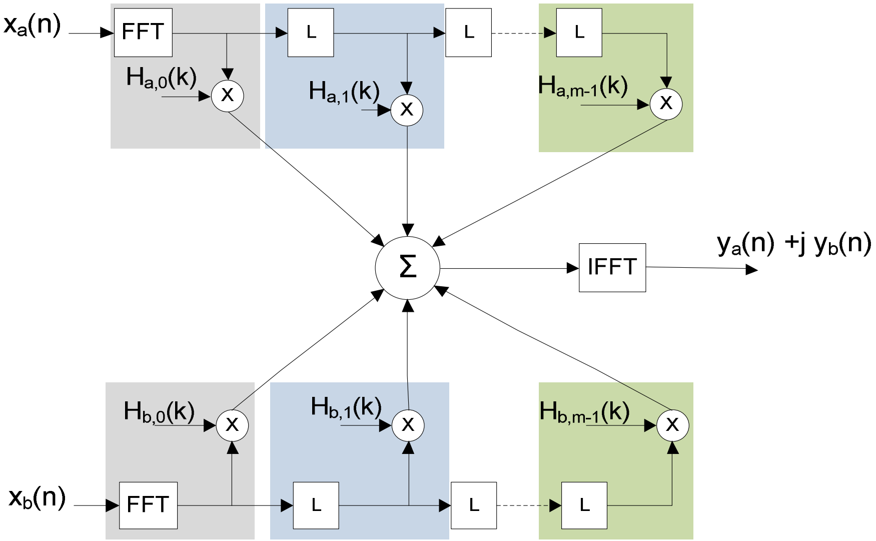


FIGURE 3: Block diagram of Mixed Uniform Partitioned Convolution to obtain CTC outputs

3.1 Theoretical Computational Complexity

The following table provides the details of computations required for proposed algorithm. Here $O(N)=N.\log_2 N$

To Calculate	Computational complexity		Remarks
	Complex Multiplications	Complex Additions	
$X_a(k)$ & $X_b(k)$.	0.5 O(2L)	O(2L)+2. 2L = O(2L) + 4L	FFTs of $x_a(n)$ & $x_b(n)$ could be found with single FFT and decomposition[7]
$X_a(k)H_a(k)$	2L.M/L=2M	2(M-L)	Each Partitioned frequency multiplication requires 2L multiplications. Such partitions are M/L and hence 2M complex multiplications are needed. All partitioned multiplier outputs are to be added, which requires 2(M-L) complex additions
$X_b(k)H_b(k)$	2L.M/L=2M	2(M-L)	”
$Y(k)$	-	2L	The outputs of $X_a(k)H_a(k)$ & $X_b(k)H_b(k)$ are complex sequences are of length 2L each. As per equation (1), the addition of these outputs index by index requires 2L complex additions
$y(n)$	0.5 O(2L)	O(2L)	Complexity of IFFT with size equal to 2L

TABLE 1: Computational complexity of proposed algorithm

Total Complex Multiplications : 4M + O(2L)
Total Complex Additions : 4M + 2 O(2L) + 2L

3.2 Efficient Memory Management

After going through the steps of the proposed algorithm, one can believe that the proposed method requires more copying routines (and hence more processing cycles) to perform FFT, frequency multiplication and IFFT evaluation. But by storing FFT output buffers in systematic way, these copying routines could be avoided. Fig. 4 depicts the approach that was followed for implementation. The diagram is showing only the real buffers of FFT. Such kind of buffers is needed for imaginary buffers also, which was not shown in the diagram.

A dedicated memory of size 2M is allocated for real & imaginary parts of $X_a(k)$ & $X_b(k)$ to store FFT values of current frame as well as delayed frames. Also the real & imaginary parts of partitioned coefficients are also arranged in memory buffers of size 2M in sequence i.e. $H_{a,0}(k)$, $H_{a,1}(k)$,...$H_{a,m-1}(k)$, etc. Initially all input buffers hold zeros. When 1st frame arrives, FFT of input buffers could be stored in last 2L locations of the dedicated 2M length buffer directly. The pointer for the last 2L location in dedicated 2M buffer could be passed as argument into FFT function evaluation. Now, these 2L values are multiplied directly with first 2L values of coefficient buffer index by index. The resultant 2L values could be added to another 2L length dedicated real & imaginary buffers. These dedicated buffers hold the final complex frequency multiplication output of size 2L and will be directly provided as input to IFFT evaluation. The next 2L locations of real & imaginary buffers are accessed in circular fashion and multiplied with associated coefficient buffers index by index and finally added to the real & imaginary dedicated IFFT input buffers. This process is continued till m = M/L stages of multiplication are performed. IFFT is evaluated with 2L size dedicated real & imaginary buffers. IFFT output produces 2L complex values of which L real & imaginary values are transmitted as $y_a(n)$ & $y_b(n)$ respectively.

When 2nd frame arrives, the FFT values of 1st frame need not be disturbed. FFT of 2nd frame will be stored in 2L locations previous to 1st frame. The buffers are accessed in circular fashion and complex frequency multiplication could be performed in same way as explained earlier with the associated partitioned complex frequency coefficients. This procedure will be continued in this way for every new frame received.

By assuming real & imaginary buffers as circular buffers, intermediate copying routines could be minimized for complex frequency multiplication as well as FFT and IFFT evaluation.

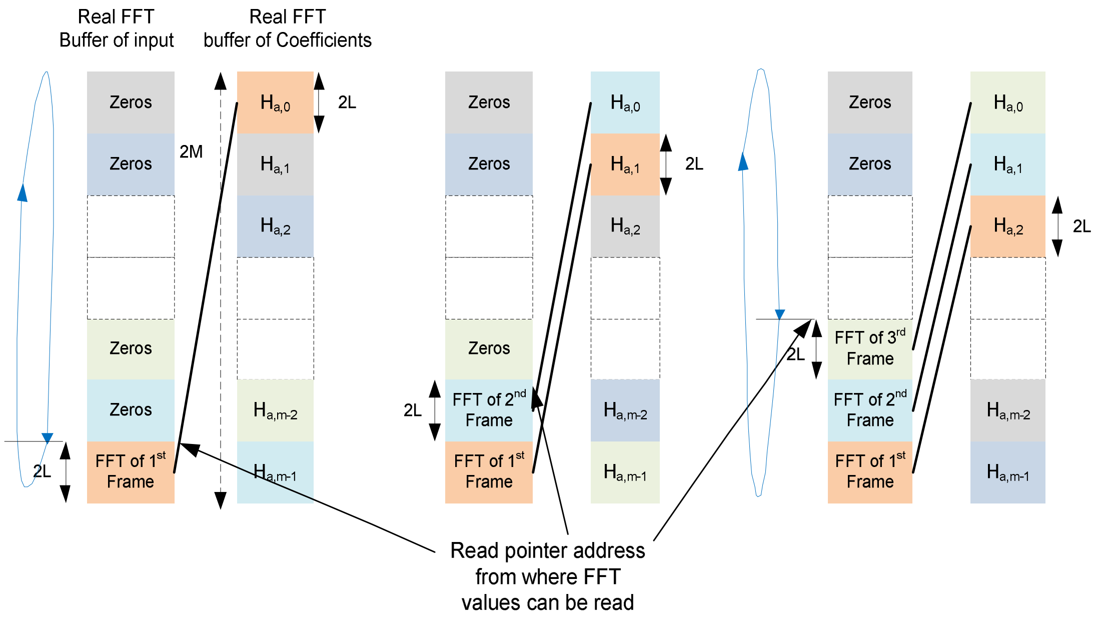
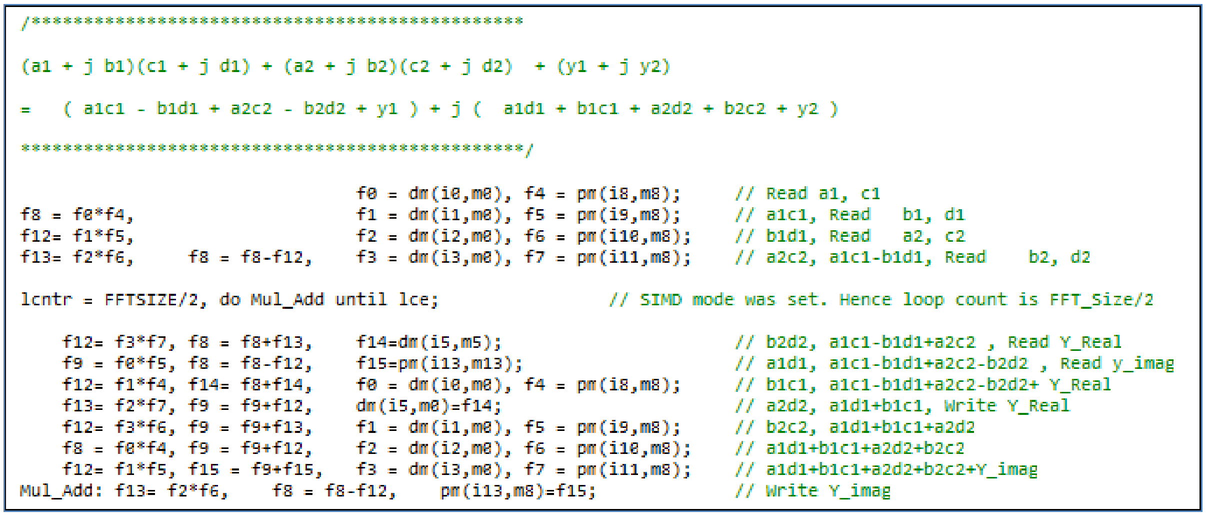


FIGURE 4: Efficient memory management in SHARC processors for Mixed Uniform Partitioned Convolution.

3.3 Efficient Complex Frequency Multiplication

The sum of complex multiplications, i.e. $[X_a(k)H_a(k)+X_b(k)H_b(k)]$, was implemented with multiplication and add instructions in parallel with data move operations efficiently on SHARC processor. This piece of code was provided here with 8 instructions inside the loop. The loop counter size is 2L, which means that this code is valid for one partitioned filter and the same code is called m times for all partitioned filters. The following analogy was made for easy understanding.

$X_a(k)$	→	a1+jb1
$H_a(k)$	→	c1+jd1
$X_b(k)$	→	a2+jb2
$H_b(k)$	→	c2+jd2



```
/**************************************************

(a1 + j b1)(c1 + j d1) + (a2 + j b2)(c2 + j d2)  + (y1 + j y2)

=   ( a1c1 - b1d1 + a2c2 - b2d2 + y1 ) + j (  a1d1 + b1c1 + a2d2 + b2c2 + y2 )

**************************************************/

                                f0 = dm(i0,m0), f4 = pm(i8,m8);     // Read a1, c1
f8 = f0*f4,                     f1 = dm(i1,m0), f5 = pm(i9,m8);     // a1c1, Read   b1, d1
f12= f1*f5,                     f2 = dm(i2,m0), f6 = pm(i10,m8);    // b1d1, Read   a2, c2
f13= f2*f6,     f8 = f8-f12,    f3 = dm(i3,m0), f7 = pm(i11,m8);    // a2c2, a1c1-b1d1, Read    b2, d2

lcntr = FFTSIZE/2, do Mul_Add until lce;                // SIMD mode was set. Hence loop count is FFT_Size/2

    f12= f3*f7, f8 = f8+f13,    f14=dm(i5,m5);                      // b2d2, a1c1-b1d1+a2c2 , Read Y_Real
    f9 = f0*f5, f8 = f8-f12,    f15=pm(i13,m13);                    // a1d1, a1c1-b1d1+a2c2-b2d2 , Read y_imag
    f12= f1*f4, f14= f8+f14,    f0 = dm(i0,m0), f4 = pm(i8,m8);     // b1c1, a1c1-b1d1+a2c2-b2d2+ Y_Real
    f13= f2*f7, f9 = f9+f12,    dm(i5,m0)=f14;                      // a2d2, a1d1+b1c1, Write Y_Real
    f12= f3*f6, f9 = f9+f13,    f1 = dm(i1,m0), f5 = pm(i9,m8);     // b2c2, a1d1+b1c1+a2d2
    f8 = f0*f4, f9 = f9+f12,    f2 = dm(i2,m0), f6 = pm(i10,m8);    // a1d1+b1c1+a2d2+b2c2
    f12= f1*f5, f15 = f9+f15,   f3 = dm(i3,m0), f7 = pm(i11,m8);    // a1d1+b1c1+a2d2+b2c2+Y_imag
Mul_Add: f13= f2*f6,    f8 = f8-f12,    pm(i13,m8)=f15;             // Write Y_imag
```

In above piece of code, y_1 & y_2 are of size 2L each, which hold the real and imaginary outputs obtained by summing the complex frequency multiplication outputs of all partitioned filters. These two buffers are provided as inputs to IFFT evaluation.

4. EXPERIMENTAL RESULTS & DISCUSSION

The proposed method “Mixed Uniform Partitioned Convolution” was implemented on SHARC ADSP-21469, 32 bit floating point DSP processor [18] with EZ-Kit Lite to measure the computational complexity. A dedicated buffer of size 2L is used for each input frames $x_a(n)$ & $x_b(n)$. Overlap save method is followed to overlap previous frames for calculation of FFT for each frame. Fixed memory buffers of size 2L are allocated to store real & imaginary FFT coefficients of all partitioned impulse responses. The complex frequency multiplication of equation (2) was implemented very efficiently using SIMD of SHARC processor. For IFFT calculation, FFT algorithm was reused by swapping the real & imaginary buffers and the scaling factor of 1/N was applied to the FFT outputs after swapping real & imaginary buffers again.

Filter Length, M	Mega Peak Cycle count	Filter Length, M	Mega Peak Cycle count
512	0.02137	9216	0.19573
1024	0.02628	9728	0.20579
1536	0.0312	10240	0.21565
2048	0.03611	10752	0.22561
2560	0.04103	11264	0.23762
3072	0.04594	11776	0.24698
3584	0.05086	12288	0.25532
4096	0.05577	12800	0.26490
4608	0.06069	13312	0.27398
5120	0.0656	13824	0.28536
5632	0.07052	14336	0.29621
6144	0.07543	14848	0.30453
6656	0.08035	15360	0.31762
7168	0.08526	15616	0.32203
7680	0.09018	16128	0.33018
8192	0.09509	16384	0.33987
8704	0.18577		

TABLE 2: Proposed Algorithm - Mega Peak Cycle counts for frame length, L=256 and variable filter lengths.

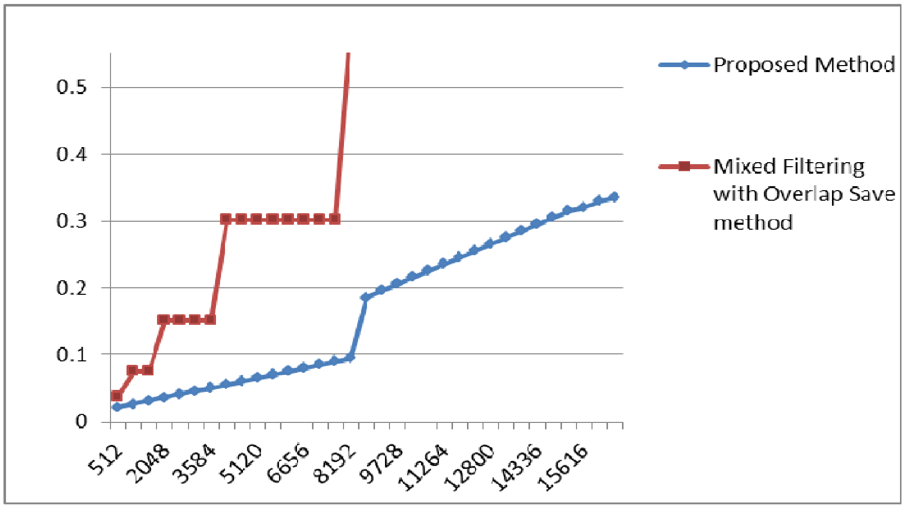


FIGURE 5: Computational complexity comparison. X-axis represents filter length, M. Y-axis represents Mega Peak cycle count. The above results are valid for frame length of L=256.

The cycle counts were calculated on SHARC processor by varying the filter length for fixed block size, L=256. Table 2 shows the computational complexity details along with that of Mixed filtering with overlap save method. Fig. 5 shows the comparison of computation cycles between Mixed filtering with overlap save method (Results were taken from Reference [7] for this method) and proposed method.

In the graph, one can observe that slight increase in mega peak cycle count for filter length of M=8704. For the cases of more than M=8704, the required internal DSP memory is not able to hold all the required buffers such as delayed FFT values of input as well as FFT coefficients. To handle this, all the contents of these buffers will be written into external storage device such as SDRAM. From SDRAM, these buffers could be read whenever needed using Direct Memory Access. To avoid too many cycles due to this memory transfer, DMA was performed in background with DSP core process. Due to all these processing, some extra cycles are needed compared to the actual signal processing and hence the mega peak cycle count was increased for these cases.

Similarly, mega peak cycle count is constant in Mixed filtering with overlap save method for few of the cases, for example M=2048 to 3584. This is obviously expected because in all these cases, FFT length is 4096 and all the operations are based on this factor. This is where the proposed method is advantageous than the overlap save method.

The experimental results clearly indicate that the proposed method provides more savings in computational complexity by following the efficient design as explained sub-sections 3.2 and 3.3. The advantage of proposed algorithm is that FFT computational complexity is a function of frame length unlike on coefficient length as in overlap save method. By segmenting filtering process into M/L parts, one can achieve attractive computational savings and also savings in memory usage.

5. CONCLUSION & SCOPE OF FUTURE WORK

To reduce the processing delay for long filters in audio crosstalk cancellation, an efficient method with combination of mixed filtering and uniform partitioned convolution is proposed to implement on real-time DSP processors. With efficient internal DSP memory management, it is possible to perform audio CTC with less computational complexity even at longer filter lengths. The results clearly indicate that the proposed method is highly dominant in both terms of computational complexity and processing delay.

This work could be extended to mixed non-uniform partitioned convolution, with which, one can obtain more efficient results. But this method is basically useful in applications related to operating systems and involves more complex process. Also instead of concentrating on FFT, it is better to use Fast Hartley Transform to reduce the computational complexity. Because FHT is always real in nature and operates directly on real signal as opposed to FFT, which always operates on complex signal, in general. The major area where more computations needed in the proposed method is complex frequency multiplication. By either utilizing the DSP architecture core instruction set effectively or simple mathematical equations, it is possible to work on this for effective computational complexity.

6. REFERENCES

[1] M. Otani and S. Ise, “Fast calculation system specialized for head-related transfer function based on boundary element method”, Journal of Acoustical Society of America, Vol. 119, 2006, No. 5, pp 2589-2598

[2] Kirkeby ole, Rubak Per, Nelson Philip A. and Farina Angelo, “Design of Crosstalk cancellation Networks by using Fast deconvolution” in AES 15, May 1999, pp 9900-9905

[3] Lentz Tobias and Scmitz Oliver, "Adaptive Cross-talk cancellation system for a moving listener" in AES 21st International Conference Proc., June 2002. Paper No. 00134

[4] Linwang, Fuliang Yin and zhe Chen, "A Stereo Crosstalk cancellation system based on common- acoustical pole/zero model', AES, August 2010

[5] SreenivasaRao. Ch, R. Udayalakshmi and Jeyasingh P. "Fast implementation of audio crosstalk cancellation of audio crosstalk cancellation on DSP processors," in AES 45 Conference Proc., March 1-4, 2012, Paper No. 2-2

[6] John G. Proakis and Dimitris G. Manolakis, "Digital Signal Processing Principles, Algorithms and Applications", 3rd Edition, Page No. 430 to 476

[7] Richard G Lyons, "Understanding Digital Signal Processing", 3rd Edition, published on November 11, 2010.

[8] Jason R. VandeKieft, April 30, 1998, "Computational improvements to linear convolution with multi-rate filtering methods" http://mue.music.miami.edu/thesis/jvandekieft/jvtitle.htm.

[9] M.Vetterli, "Running FIR and IIR Filters using Multi-rate Filter Banks", IEEE transactions on Acoustics, Speech and Signal Processing, May 1988, Vol. 36, No.5.

[10] Eric Battenbaerg and Rimas Avizienis. "Implementing Real-time Partitioned Convolution Algorithms on Conventional Operating Systems", Proc. of 14th Int. Conference on Digital Audio Effects, Paris, France, Sept 19-23, 2001.

[11] Anders Torger and Angelo Farina, " Real-time Partitioned Convolution for Ambiophonic Surround Sound", IEEE Workshop on applications of Digital Signal Processing to Audio and Acoustics 2001, New Paltz, New York, W2001-4.

[12] Garcia Guillermo, "Optimal Filter Partition for efficient Convolution with short input/output delay" in AES 113th International Conference Proc., October 2002, pp. 2660.

[13] WG Gardiner, "Efficient Convolution without input-output delay", Journal of AES, Vol. 43, No. 3, 1995, pp. 127-136.

[14] J. Hurchalla, "A time distributed FFT for efficient low latency convolution", AES Convention 129, November 2010, Paper No.8257

[15] J. Hurchalla, "Low latency convolution in one dimension via two dimensional convolutions-An intuitive approach", AES Convention 125, October 2008, Paper No. 7634.

[16] E. Armelloni, C. Giottoli and A. Farina, "Implementation of Real-time partitioned convolution on a DSP board", IEEE Workshop on Applications of Signal processing to Audio and Acoustics, October 19-22, 2003, New Paltz, NY.

[17] Eric Battenberg, David Wessel & Juan Colmenares, "Advances in the Parallelization of Music and Audio Applications", ebookbrowse.com/wessel-parlab-retreat-winter-2010-ppt-d59199484

[18] Analog Devices Inc., "ADSP-214xx SHARC Processor Hardware Reference Manual", Rev 0.3, Part Number 82-000469-01, July 27, 2010.

A Novel, Robust, Hierarchical, Text-Independent Speaker Recognition Technique

Prateek Srivastava *prateek.k.srivastava@gmail.com*
National Institute of Technology
Rourkela, India, 769008

Reena Panda *reena.panda@gmail.com*
National Institute of Technology
Rourkela, India, 769008

SankarsanRauta sankarsan.1946*@gmail.com*
National Institute of Technology
Rourkela, India, 769008

Abstract

Automatic speaker recognition system is used to recognize an unknown speaker among several reference speakers by making use of speaker-specific information from their speech. In this paper, we introduce a novel, hierarchical, text-independent speaker recognition. Our baseline speaker recognition system accuracy, built using statistical modeling techniques, gives an accuracy of 81% on the standard MIT database. We then propose and implement a novel state-space pruning technique by performing gender recognition before speaker recognition so as to improve the accuracy/timeliness of our baseline speaker recognition system. Based on the experiments conducted on the MIT database, we demonstrate that our proposed system improves the accuracy over the baseline system by approximately 2%, while reducing the computational time by more than 30%.

Keywords: Speaker Recognition, Gender classification, Mel Frequency Cepstral Coefficients, Cepstral Mean Subtraction, Gaussian Mixture Model.

1. INTRODUCTION

Speaker recognition is the task of automatically recognizing/identifying an unknown speaker among several reference speakers using speaker-specific information included in speech waves [10]. Such a system can have several potential applications such as a biometric tool for security purposes. Speech being one of the most natural and common form of communication, any speech-based security system would be non-intrusive and havehigher user acceptance. Also such systems can be easily integrated into the ubiquitous telephone network, thereby providing access controlfor banking transactions by telephone, automatictelephonetransactions such as voice mail and credit card verification, and remote access to computers via modems on dial-up telephone lines. Such a system can also have potential applications in forensics.

Speaker recognition [5, 21, 22] combines both speaker verification and speaker identification. Speaker verification is the technique to verify a person's claimed identity by making use of the speech cures. On the other hand, in speaker identification, no identity claims are made and the system has to identify the speaker. Significant work has been done in the area of speaker recognition over the past years. The most notable and widely referred approaches are:- the Gaussian mixture model (GMM - UBM) [19], and the mixed GMM- UBM and SVM technique [23]. Speaker recognition systems can be further divided into text-dependent and text-independent systems. In text-dependent systems [24], the recognition phrases/words are constant or known a priori. On the other hand, in text-independent systems, there are no constraints on the words which the speakers are allowed to use and thus, text independent recognition is considered to be a more challenging task.

In this paper, we propose a text-independent speaker recognition system based on Gaussian Mixture Models (GMMs) which is proven to be a powerful tool and is often employed in text-independent classification tasks. We also propose a technique for speeding up and improving the accuracy of the speaker identification task by pruning the search space by dropping out the unlikely speakers by making use of gender recognition before speaker identification.

The rest of the paper is organized as follows. In Section 2, we describe our speaker recognition technique. Section 3 discusses the improvements that we propose to our speaker recognition system. Section 4 provides a description of the experiments along with a detailed analysis of the results. We finally conclude the paper in Section 5 with notes regarding the future work.

2. GMM-Based Speaker Recognition

The recognition system is divided into two phases namely training phase and testing phase. In training phase, speech samples are collected pre-processed and then speaker-specific features are extracted from them. Thereafter, the different speaker classes are statistically modeled using GMMs. In the testing phase, features are extracted from the test samples and their likelihood of match is estimated against the trained models. The model against which the test sample yields the highest likelihood score is identified as the speaker class. This is shown in Figure 1.

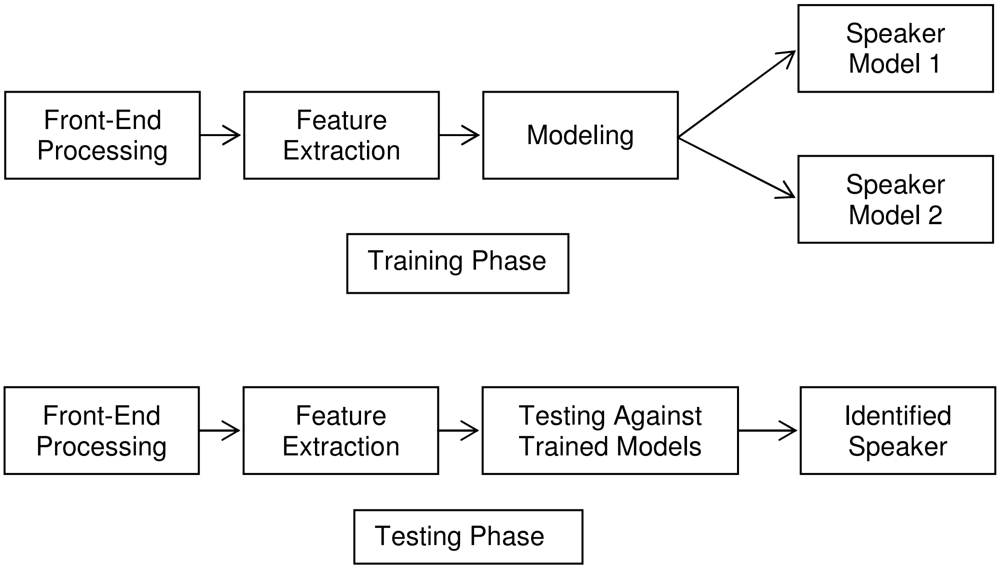


FIGURE 1: Pictorial Representation of the Speaker Recognition System.

In the following subsection, we describe the techniques that we use for front-end processing, feature extraction and feature matching respectively.

2.1 Front end Processing

2.1.1 Pre-emphasis

The sampled speech is pre-emphasized to enhance the high frequency components of the spectrum, especially the so-called formants, against the lower frequencies which contain most of the signal’s power, but are known to be rather irrelevant for speech intelligibility. Pre-emphasis of the high frequencies is done to obtain similar amplitudes for all the formants [8]. This is performed by applying a first order FIR filter to the speech signal:

$$s[k] = s[k] – a_{1.}\, s[k-1] \text{ where } a_1 = 0.97$$

2.1.2 Framing

The resulting pre-emphasized speech signal is then divided into smaller parts out of which certain features essential for recognition are extracted. These short-time intervals of the speech signal are called frames. Since the frame duration is very small, each frame is assumed to be a stationary process and is assumed to have a constant spectrum. Overlapping of the frames is done so that the adjoining frames would overlap to achieve a smoother development of the short-time characteristics of the individual signal blocks [10]. Overlapping is done mainly to avoid loss of information.

2.1.3 Windowing

All these frames are then multiplied by a window function. This is required to smooth the edges of each frame to reduce the discontinuities or abrupt changes at the endpoints. Windowing also serves to reduce the spectral distortion that arises from the windowing itself [10]. Here, in our experiments, we have made use of a hamming window, which is characterized by:

$$w(n) = 0.54 – 0.46 \cos \left(\frac{2\pi n}{N-1} \right)$$

where, N = width in samples and n is an integer with values $0 < n < N- 1$

2.2 Feature Extraction and Modeling

The acoustic signal contains different kinds of information about the speaker. The signal processing involved changes depending on the type of characteristics we are interested in the speaker. The basic aim of feature extraction in our recognition system is to reduce the amount of data while retaining the speaker-dependent and gender-specific information.

2.2.1 Mel-Frequency Cepstral Coefficients (MFCCs)

MFCCs have by far, proved to be the most successful and robust feature for recognition purposes. The MFCC feature set is based on the human perception of sound i.e., on the known evidence that the information carried by low-frequency components of the speech signal are phonetically more important for humans than the high-frequency components [9]. This is expressed in the mel-frequency scale, which is linear frequency spacing below 1000 Hz and a logarithmic spacing above 1000 Hz .The MFCC feature extraction algorithm [3, 4, 11] is shown in Figure 2.

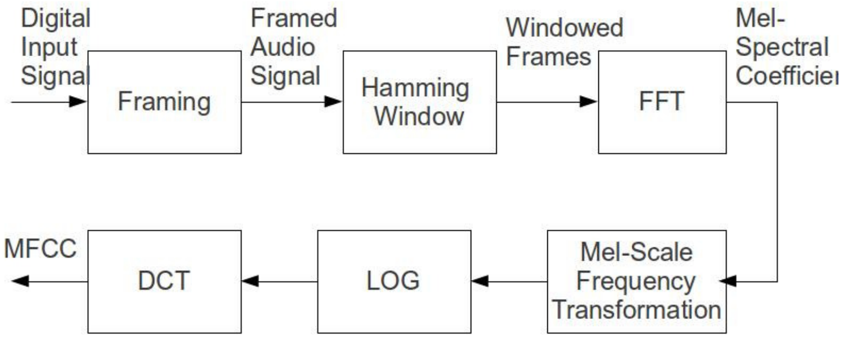


FIGURE 2:MFCC Feature Extraction Process.

The final MFCC feature vector is composed of 39 parameters (including the delta and delta- delta coefficients which are added to model the inter-frame dependencies in speech and are the time derivatives of the basic static parameters). However these delta and delta-delta coefficients can increase the feature vector by up to 24 dimensions. So, in this paper we have used the Delta Cepstral energy (DCE) and Delta-Delta Cepstral Energy (DDCE) that can compactly represent the delta and delta-delta cepstral information in one-dimensional feature [12]. For any one frame, they are calculated as follows:-

$$\text{DCE} = \sum_{l=1}^{L} (\Delta MFCC)^2$$

$$\text{DDCE} = \sum_{l=1}^{L} (\Delta^2 MFCC)^2$$

where, $\Delta MFCC_l$, $\Delta^2 MFCC_l$ are the l^th^delta and delta-delta cepstral coefficients and L is the number of MFCCs.

2.2.2 Maximum Auto-Correlation Value (MACV)

The pitch frequency is an extremely important property of speech and defines the periodicity of a speech signal. However the accurate pitch extraction is not an easy task due to the non-stationarity and quasi-periodicity of speech signal, as well as the interaction between the glottal excitation and the vocal tract. Also speech frames are not always periodic and pitch cannot be determined for the unvoiced frames. So, here we have used the Maximum Auto-correlation algorithm [10] (MACV) which does not use pitch value directly as a feature and works well for both voiced and unvoiced frames. It captures the periodicity characteristics of speech signal in an indirect manner in the form of voicing information.

2.2.3 Cepstral Mean Subtraction (CMS)

Practically, the speech samples in the database are collected using different microphones, each having its own inbuilt channel noise. This channel noise gets convolved with the environmental noise. To remove the variability in different speech samples owing to the use different microphones, we make use of the Cepstral mean features (CMS). After the features are extracted from each speech sample, the mean of the whole feature set is calculated and is subtracted from each frame to get the Cepstral mean features. It is assumed throughout that the speech signal has a zero mean and the channel noise is finite. It has been established experimentally in prior research work that CMS yields more robust features than MFCC by itself.

2.3 Gaussian Mixture Modeling

Gaussian mixture models (GMMs) [19, 20] are parametric representation of a probability density function. When trained to represent the distribution of a feature vector, GMMs can be used as classifiers. GMMs have proved to be a powerful tool for distinguishing acoustic sources with different general properties. The use of GMMs for modeling activity is motivated by the interpretation that the (1) uni-variate Gaussian densities have a simple and concise representation, depending uniquely on two parameters, mean and variance, (2) they are capable to model arbitrary densities, (3) the Gaussian mixture distribution is universally studied and its behaviors are widely known, (4) a linear combination of Gaussian basis functions is capable of modeling a large class of sample distributions. In principle, the GMM can approximate any probability density function to an arbitrary accuracy.

A GMM is a weighted sum of M component densities as shown in figure, given by the equation:-

$$P\ (x_t\ |\lambda_s) = \sum_{i=1}^{M} p_i\, b_i\ \ (x(t))$$

Here, x_t is a sequence of feature vectors from the activity data, x(t) is feature vector having D-dimensionality. $b_i(s)$ is the Gaussian probability distribution function (PDF) associated with the i^th^ mixture component and is given by:

$$b_i(x_t)=\frac{1}{2\pi^{D/2}|\sum_s^i|^{1/2}}e^{\left(-\frac{1}{2}\right)(x-\mu i)\hat{}T\sum_i^s{}^{-1}(x-\mu i)}$$

Here, $\mu_i$is the mean vector and $\sum^s_i$ is the covariance matrix of the i^{th} mixture component.

The mixture weights are such that:-

$$\sum_{i=1}^{M} p_i \quad = 1$$

Each trained speaker is thus, represented by a Gaussian mixture model, collectively represented by:-

$$\lambda_s=\{\mu_i , \sum_i , p_i \}$$

where, i=1,2 ,…M, μ_i , $\sum_i$, p_i represent the mean, covariance and weights of the i^{th}mixture respectively.

In this paper, the models are trained using the expectation-maximization (EM) algorithm [18]. The basic idea of the EM algorithm is as follows:- Beginning with an initial model λ, to estimate a new model λ', such that p (X | λ') ≥ p(X | λ).The new model then becomes the initial model for the next iteration and the process is repeated until some convergence threshold is reached.

But during the implementation of the EM algorithm, a singularity problem arises which limits the training to a limited number of Gaussians. To avoid these problems, a variance flooring method is generally used. However in our experiments, we find that the variance flooring method is also not able to solve the singularity problem altogether. In our experiments, we found out that if, an optimum splitting of mean is implemented during the EM implementation, then it deals with the singularity problem completely and we are able to train the data to any number of Gaussians. So, in our experiments, we have proposed and implemented this optimum splitting of mean technique so as to overcome the singularity problem.

3 Proposed Improvements to the Speaker Recognition System

In this paper, we propose a novel technique to improve both the accuracy and computational speed of the speaker identification task by pruning the state search space. We propose to do so by dropping out the more unlikely speakers from the search space by preceding the speaker identification stage with a gender recognition stage. The basic idea of the two approaches we have followed for Speaker Identification purposes is depicted below in Figure 3, where the top figure corresponds to the original speaker identification system and the bottom figure demonstrates our proposed changes.

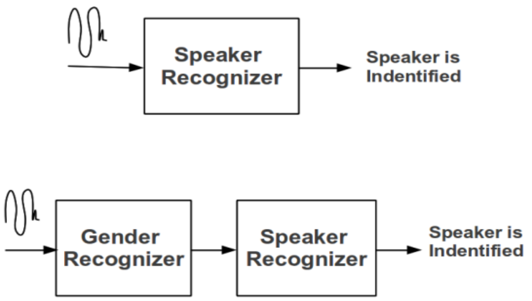


FIGURE 3: Speaker recognition system and Hierarchical Speaker Recognition System.

From our speaker recognition experiments, we observed that some of the incorrect recognition cases resulted from confusions with speakers of a different gender. In a separate set of experiments, where we performed gender recognition using speech features, we achieved significantly higher recognition accuracies. So, we tried to improve our speaker recognition system by implementing a pruning stage before the actual speaker recognition stage where we first estimate the speaker's gender and then perform speaker recognition on the identified smaller speaker set. Such a system has two main advantages :- a) First of all, if the implemented gender recognition system is highly accurate, then it would allow to reduce the inter-gender confusions thereby resulting in a higher overall recognition accuracy, b) A gender recognizer before the speaker recognizer prunes the state space and thus, the computational speed of the overall system improves. The results of this improved recognition system is provided in the next section.

The hierarchical recognition system improves the performance by reducing the inter gender misclassification. The hierarchical approach exploits the difference in statistical properties of male and female during 1st phase of recognition. This approach also provides us the flexibility to use more targeted feature for different gender cluster. In this paper, we have used different feature set for gender recognition phase and speaker recognition phase in hierarchical recognition system.

4 Experiments and Results

4.1 Dataset

In this paper, we have conducted the experiments on the MIT database and a self-collected database:-

- MIT Database
  - Was collected by a prototype hand held device in order to simulate scenarios encountered by real-world speech recognition and verification systems.
  - Used different locations as well as different microphones.
  - Total 48 speakers with 22 females and 26 males.
  - Sampled at 16k Hz.

- Self-Collected Database
  - To deal with real time noisy condition
  - Total 20 Indian speakers including 9 females and 11 males.
  - Sampled at 16k Hz.
  - Different microphone and collected in different sessions.

4.2 Experiments and Results

During the training phase, the speech signals from each speaker class were pre-emphasized using a first order FIR filter (pre-emphasis coefficient = 0.97). Then they were divided into 20 ms frames with an overlap of 10ms. Each frame was then, windowed using a hamming window. Features are then extracted from each windowed frame. In our experiments, we have used feature vectors composed of 12 lowest Mel-frequency cepstral coefficients computed using 21 Mel-spaced filters (the 0th coefficients being excluded because they carry little speaker-specific information), the delta and delta-delta coefficients, the delta and delta-delta cepstral energy, 5 MACV features derived from the auto-correlation function and the cepstral mean subtraction features (determined for each utterance). After extraction of features, the speaker classes were statistically modeled using GMMs. EM algorithm was then used for estimating the parameters of the GMM class. At the end of the training phase, we were thus, left with Gaussian mixture models, representing each speaker class. Experiments were conducted with 32 and 64 mixtures.

In the testing phase, similarly features were extracted for the test utterances of the corresponding databases. Their likelihoods were estimated against the trained models. The model against which it yielded the highest likelihood score was identified as the speaker.

4.2.1 Gender Recognition in Standard Database

The first set of experiments was conducted to perform gender recognition on the complete set of male and female files in the MIT database. The accuracies obtained for different sets of features are shown in Table 1.

Sl No	Features used	No. of mixtures	Male	Female	Overall accuracy
1	MFCC+DCE+ DDCE	16	90.01	97.96	93.795
2	MFCC+DCE+ DDCE+ MACV(5)	32	95	98.4	96.619
3	MFCC+DCE+ DDCE+ MACV(3)	64	94.36	97.7	95.95

TABLE1: Gender Recognition Accuracies on MIT Database.

The MFCC+DCE+DDCE served as our baseline system which gave an accuracy of 93.795%. Including 3 MACV features improved the results by almost 2% with a marginal increase in the dimensionality. Including 5 MACV features again increased the accuracy of the system.

4.2.2 Gender Recognition on Self-Collected Database

We repeated the same set of gender recognition experiments as discussed above on the self-collected database. For the feature set composed of MFCC, DCE, DDCE and 5 MACV features, we obtained 100% accuracies in distinguishing between the male and female speaker classes.

4.2.3 Speaker Recognition on Standard Database

I) In the 3rd set of experiments, 48 speaker models (48 male/female speakers) were trained with MFCC+ΔMFCC+ ΔΔMFCC (39 feature vector set) using 64 Gaussian mixture models and tested using the test utterances of the speakers (other than the training utterances). The results are shown in Table 2. We obtained an overall accuracy of 81.058% and out of which the female and male accuracies are 78.6209% and 83.1204%.

Spk.	Accuracy	Spk.	Accuracy	Spk.	Accuracy
f00	85.19	f16	85.19	m10	81.48
f01	81.48	f17	72.22	m11	92.59
f02	61.11	f18	70.37	m12	81.48
f03	59.26	f19	64.18	m13	77.78
f04	68.54	f20	90.74	m14	66.67
f05	79.63	f21	87.04	m15	88.89
f06	70.37	m00	92.59	m16	68.54
f07	98.15	m01	87.04	m17	77.78
f08	81.48	m02	83.33	m18	88.89
f09	87.04	m03	68.52	m19	83.33
f10	90.04	m04	83.33	m20	77.78
f11	75.93	m05	90.74	m21	77.78
f12	90.74	m06	98.15	m22	87.04
f13	85.19	m07	94.44	m23	83.33
f14	83.33	m08	88.89	m24	72.26
f15	61.11	m09	100	m25	68.52

TABLE2: Speaker Recognition Accuracies for MFCC+ΔMFCC+ ΔΔMFCC on MIT Database.

II) In the next set of experiments, the 48 speaker recognition system were built using CMS with 64 Gaussian mixtures. We improved overall accuracy to 83.869 % as compared to the baseline system accuracy of 81.058%. It is observed that though for few speakers, the accuracy went down as compared to standard MFCC but in general, it increased for all the speakers. The results are shown in Table 3.

Spk.	Accuracy	Spk.	Accuracy	Spk.	Accuracy
f00	88.89	f16	68.52	m10	85.19
f01	70.37	f17	66.67	m11	100
f02	85.19	f18	66.67	m12	74.07
f03	68.52	f19	81.48	m13	72.22
f04	53.70	f20	100	m14	70.37
f05	94.44	f21	92.59	m15	98.15
f06	74.07	m00	98.15	m16	100
f07	100	m01	94.44	m17	87.04
f08	83.33	m02	85.19	m18	83.33
f09	62.96	m03	83.33	m19	74.07
f10	92.59	m04	100	m20	92.59
f11	74.07	m05	85.19	m21	87.21
f12	85.19	m06	81.48	m22	81.48
f13	94.44	m07	100	m23	81.48
f14	98.15	m08	90.74	m24	62.96
f15	68.15	m09	100	m25	87.04

TABLE3: Speaker Recognition Accuracies for CMS Feature set on MIT Database.

We observe that system accuracy for 64 Gaussians is best for this dataset.

4.2.4 Speaker Recognition on Self-Collected Database

In the next set of experiments, we performed speaker recognition on the self-collected database. Preliminary 8-speaker models were built with single Gaussian mixture modeling. With the 39-vector set MFCC, 100% accuracies were obtained for the training data. In another set of experiments, 8 speaker models were made using MFCC and CMS features and the accuracies obtained was found to be 95.413%. The results are shown in Table 4.

Sl. No	Speaker	Accuracy
1	Spk 1	100
2	Spk 2	100
3	Spk 3	100
4	Spk 4	96.67
5	Spk 5	76.67
6	Spk 6	96.67

7	Spk 7	93.3
8	Spk 8	100

TABLE4: Speaker Recognition Accuracies for CMS Feature set on Self-Collected Database.

4.2.5 Hierarchical Speaker Recognition System

As discussed before in Section III, we then performed experiments to improve our speaker recognition accuracies by combining the system with a gender recognizer. The final system that was built would first classify the speaker's gender and then, it will recognize the speaker's identity in that speaker class. We performed the baseline experiment on the modified system and the results are shown in Table 5.

Spk.	Accuracy	Spk.	Accuracy	Spk.	Accuracy
f00	87.04	f17	68.52	m11	88.89
f01	59.26	f18	57.41	m12	74.07
f02	88.81	f19	94.07	m13	57.41
f03	66.67	f20	100	m14	75.47
f04	35.42	f21	96.30	m15	66.67
f05	90.74	m00	96.30	m16	96.30
f06	70.37	m01	94.44	m17	79.63
f07	98.15	m02	88.89	m18	83.33
f08	66.67	m03	66.67	m19	96.30
f09	72.22	m04	98.15	m20	88.89
f10	92.59	m05	90.74	m21	88.89
f11	81.48	m06	88.89	m22	81.48
f12	90.74	m07	100	m23	88.89
f13	96.30	m08	66.67	m24	70.37
f14	83.33	m09	96.30	m25	88.89
f15	79.63	m10	81.48	-	-
f16	77.78				

TABLE5: Table showing accuracies for speaker recognition using gender recognition

The overall system accuracy improved from 81.058% (with the baseline system) to 82.56% (with our novel proposed system). Also, the computational time of the system reduced from 64.812s to 44.078 seconds. The comparison of the accuracies of the systems 4.2.2(I) and 4.2.3 can be graphically seen in Figure 4. It is expected that performing the same experiments using the CMS feature would also improve the performance in a similar fashion, which we may perform in future.

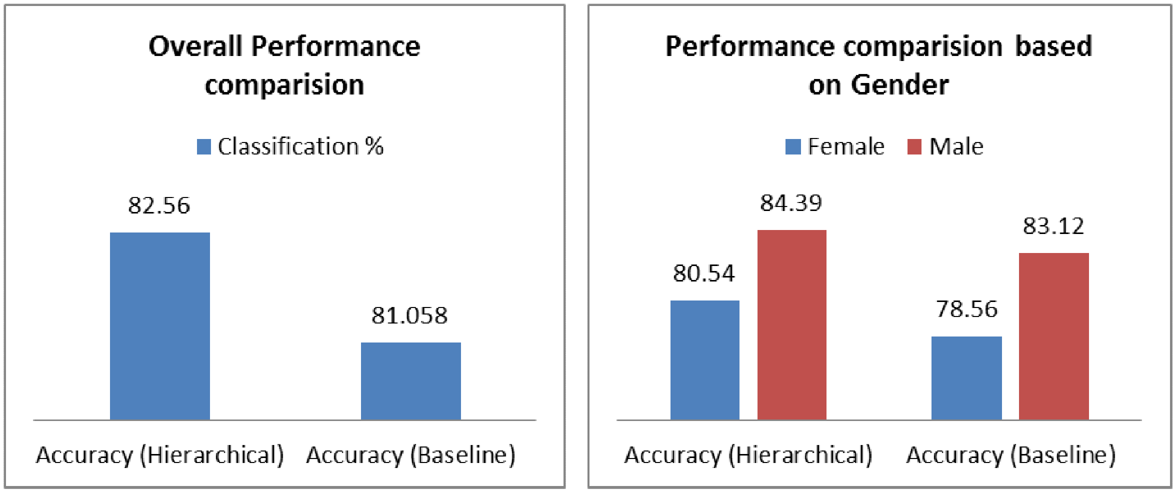


FIGURE 4: Figure showing the performance improvement using the hierarchical recognizer

5 Conclusion and Future Work

In this paper, we first propose a text-independent speaker recognizer using Gaussian Mixture Models. For a combination of 39 MFCC features, we obtained an accuracy of 81.058% on the MIT Database, which served as our baseline system. In another set of experiments, we demonstrated that using the CMS feature improves the accuracy of the system to 83.869%. Since the number of speakers in MIT Dataset is 48, the performance of system is around 83% but the recognition of self-collected dataset, which contains only 8 speakers is relatively high around 96%.

We then proposed a novel technique to improve the performance of our baseline speaker recognizer by implementing gender recognition before the speaker recognition. Through experimental results, we finally show that the enhanced system has improvedthe system accuracy by more than 1.85% while reducing computational time by over 30%. Thus, the proposed hierarchical approach provides a better performance compared to our baseline.

As a part of the future work, we would suggest to implement some additional features which would be having more speaker-relevant information. In future, the learning from this system can be adopted to build a real time system as this approach effectively reduces the recognition time. Also, the overall accuracy of the system can be possibly improved by developing GMMs which take care of the degree of overlap between different speaker classes and thereby, giving more weightage to the non-overlapped segments.

6 Acknowledgement

The authors gratefully acknowledge the contributions of Prof S. Umesh (*IIT Madras*), Dr Shakti Rath and DrSanand for their guidance during the initial phase of this project.

7 References

[1] X. Huang, A.Acero and H.-W.Hon, *Spoken language processin*g, Upper Saddle River, New Jersey, Prentice Hall PTR, 2001.

[2] S. Furui, Digital Speech Processing, Synthesis and Recognition, New York, Marcel Dekker, 2001.

[3] J. R. Deller, J. H. L. Hansen, J. G. Proakis, Discrete-Time Processing of Speech Signals, Piscataway (N.J.), IEEE Press, 2000.

[4] X. Huang, A. Acero and H.-W.Hon, Spoken language processing, Upper Saddle River, New Jersey, Prentice Hall PTR, 2001.

[5] D. A. Reynolds, “An Overview of Automatic Speaker Recognition Technology”, ICASSP 2002, pp 4072-4075.

[6] EvgenyKarpov, ‘Real-Time Speaker Identification’, University of Joensuu Department of Computer Science Master’s Thesis

[7] Mohamed FaouziBenZeghibaa, ‘Joint Speech And Speaker recognition' IDIAP RR 05- 28, February 2005

[8] J.R Deller, J.H.L. Hansen, J .G. Proakis, Discrete –Time processing of speech signals, Piscataway (N.J.),/IEEE Press,2000

[9] Brett Richard Wildermoth,'Text Independent Speaker Recognition using source based features', January 2001, Griffith university , Australia.

[10] H. Gish and M. Schmidt, “Text Independent Speaker Identification”, IEEE Signal Processing Magazine, Vol. 11, No. 4, 1994, pp. 18-32.

[11] MohaddesehNosratighods ,EliathambyAmbikairajah ,and Julien Epps “SPEAKER VERIFICATION USING A NOVEL SET OF DYNAMIC FEATURES”

[12] J .M.Naik ,”Speaker Verifiaction-A tutorial”, IEEE Communications Magazine, January 1990,pp.42-48.

[13] D. A. Reynolds, “An Overview of Automatic Speaker Recognition Technology”, ICASSP 2002, pp 4072-4075.

[14] J.P. Campbell, “Speaker Recognition: A Tutorial”, Proc. of the IEEE, vol. 85, no. 9, Sept 1997, pp. 1437-1462

[15] H. Gish and M. Schmidt, “Text Independent Speaker Identification”, IEEE Signal Processing Magazine, Vol. 11, No. 4, 1994, pp. 18-32.

[16] D. Reynolds, R. Rose, “Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models”, IEEE transactions on speech and audio processing, Vol. 3, No1, 1995, pp. 72-83

[17] Jeff A. Bilmes , “A Gentle Tutorial of the EM Algorithm and its application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models”, TR-97-021, April 1998

[18] leonard, R. ,G. ,' A Database for speaker independent digit recognition' , Proc. ICASSP 84 , Volume 3, p. 42.11, 1984

[19] D. A. Reynolds, A Gaussian mixture modeling approach to text independent speaker identification, Ph.D. thesis, Georgia Institute of Technology, Atlanta, Ga, USA, September 1992.

[20] S. Roberts, D. Husmeier, I. Rezek, andW.Penny, “Bayesian approaches to gaussian mixture modeling,” IEEE Trans. Pattern Anal. Machine Intell., vol. 20, pp. 1133–1142, Nov. 1998.

[21] Atal, B.S“Automatic recognition of speakers from their voices,” Proc. IEEE, vol. 64, pp. 460–475, 1976.

[22] SadaokiFurui“Speaker-dependent-feature extraction, recognition and processing techniques,” Speech Commun., vol. 10, pp. 505–520, 1991.

[23] Campbell W, Sturim D, Reynolds D, Solomonoff A. SVM-based speaker verification using a GMM supervector kernel and NAP variability compensation. In: Proceedings of the international conference on acoustics, speech and signal processing; 2006. p. 1–97.

[24] Herbert, M., 2008. Text-dependent speaker recognition. In: Benesty, J., Sondhi, M., Huang, Y. (Eds.), Springer Handbook of Speech Processing. Springer-Verlag, Heidelberg, pp. 743–762.

INSTRUCTIONS TO CONTRIBUTORS

The *International Journal of Signal Processing (SPIJ)* lays emphasis on all aspects of the theory and practice of signal processing (analogue and digital) in new and emerging technologies. It features original research work, review articles, and accounts of practical developments. It is intended for a rapid dissemination of knowledge and experience to engineers and scientists working in the research, development, practical application or design and analysis of signal processing, algorithms and architecture performance analysis (including measurement, modeling, and simulation) of signal processing systems.

As SPIJ is directed as much at the practicing engineer as at the academic researcher, we encourage practicing electronic, electrical, mechanical, systems, sensor, instrumentation, chemical engineers, researchers in advanced control systems and signal processing, applied mathematicians, computer scientists among others, to express their views and ideas on the current trends, challenges, implementation problems and state of the art technologies.

To build its International reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for SPIJ.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Started with Volume 6, 2012, SPIJ appears with more focused issues related to signal processing studies. Besides normal publications, SPIJ intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

We are open to contributions, proposals for any topic as well as for editors and reviewers. We understand that it is through the effort of volunteers that CSC Journals continues to grow and flourish.

SPIJ LIST OF TOPICS

The realm of Signal Processing: An International Journal (SPIJ) extends, but not limited, to the following:

- Biomedical Signal Processing
- Communication Signal Processing
- Detection and Estimation
- Earth Resources Signal Processing
- Industrial Applications
- Optical Signal Processing
- Radar Signal Processing
- Signal Filtering
- Signal Processing Technology
- Software Developments
- Spectral Analysis
- Stochastic Processes
- Acoustic and Vibration Signal Processing
- Data Processing
- Digital Signal Processing
- Geophysical and Astrophysical Signal Processing
- Multi-dimensional Signal Processing
- Pattern Recognition
- Remote Sensing
- Signal Processing Systems
- Signal Theory
- Sonar Signal Processing
- Speech Processing

CALL FOR PAPERS

Volume: 6 - **Issue: 5**

i. Paper Submission: October 31, 2012 **ii. Author Notification:** November 30, 2012

iii. Issue Publication: December 2012

CONTACT INFORMATION

Computer Science Journals Sdn BhD
M-3-19 Plaza Damas, Sri Hartamas
50480, Kuala Lumpur, MALAYSIA

Phone: 006 03 6207 1607
Fax: 006 03 6207 1697

Email: cscpress@cscjournals.org

CSC PUBLISHERS © 2012
COMPUTER SCIENCE JOURNALS SDN BHD
M-3-19, PLAZA DAMAS
SRI HARTAMAS
50480, KUALA LUMPUR
MALAYSIA

PHONE: 006 03 6207 1607
FAX: 006 03 6207 1697
EMAIL: cscpress@cscjournals.org