

Analisis *Running Time* Algoritma MFCC-Multikanal dan MFCC -Dualkanal untuk Ekstraksi

Roy Rudolf Huizen², Florentina Tatrin Kurniati²

STIKOM Bali

Jln Raya Puputan 86 Renon Denpasar Bali, Telp. 0361-244445

e-mail: ¹roy@stikom-bali.ac.id, ²floren@stikom-bali.ac.id

Abstrak

Identifikasi suara prosesnya diawali dengan mengekstraksi sampel kata. Ekstraksi ciri merupakan bagian penting pada pengenalan suara. Fungsinya untuk memperoleh karakteristik pola frekuensi pada setiap kata. Salah satu metode ekstraksi ciri yang andal adalah MFCC (Mel-Frequency Cepstral Coefficients). Metode ini telah banyak dikembangkan guna meningkatkan nilai akurasi. Pengembangan metode terkadang tidak memperhatikan kompleksitas algoritma, namun hanya sebatas nilai akurasi. Agar dapat mengetahui kompleksitas algoritma perlu dilakukan analisis dengan menggunakan uji Big O, yaitu dengan mengukur dalam dimensi waktu (*running time*). Pada penelitian ini varian MFCC yang akan di analisis adalah MFCC-multikanal dan MFCC dualkanal. Pengujian menggunakan beberapa varian data mulai dari berjumlah 1, 10, 20, 40, 200 dan 400. Hasil pengujian diperoleh untuk varian MFCC multikanal *running time* terendah adalah 0.0774 detik dan tertinggi 23.7211 detik, sedangkan pada MFCC dualkanal, *running time* terendah 0.00398 detik dan tertinggi 12.58 detik. Berdasarkan hasil tersebut MFCC multikanal mempunyai *running time* lebih tinggi dibandingkan MFCC dualkanal. Agar *running time* tidak terlalu tinggi pengembangan metode MFCC perlu memperhatikan kompleksitas algoritma sehingga nilai akurasi dan kompleksitas dapat selaras.

Kata kunci: Ekstraksi Ciri, *Running Time*, MFCC-multikanal, MFCC-dual kanal.

Abstract

Voice identification process begins with extracting said words. Feature extraction is an important part of speech recognition. Its function is to obtain the frequency pattern characteristics in each word. One reliable feature extraction method is MFCC (Mel-Frequency Cepstral Coefficients). This method has been developed to increase the value of accuracy. Method development sometimes does not pay attention to the complexity of the algorithm, but only to the value of accuracy. In order to find out the complexity of the algorithm, it is necessary to analyze it using the Big O test, namely by measuring it in the dimension of *running time*. In this study MFCC variants to be analyzed are MFCC-multichannel and MFCC dual-channel. The test uses several data variants ranging from 1, 10, 20, 40, 200 and 400. The test results obtained for the MFCC multi-channel variant *running time* is the lowest 0.0774 seconds and the highest 23.7211 seconds, while in MFCC dual-channel, the *running time* is lowest 0.00398 seconds and the highest 12.58 seconds. Based on these results, multichannel MFCC has a higher *running time* than MFCC dual-channel. So that *running time* is not too high the development of the MFCC method needs to pay attention to the complexity of the algorithm so that the value of accuracy and complexity can be aligned.

Keywords: Feature Extraction, *Running Time*, MFCC-Multichannel, MFCC-Dual Channel.

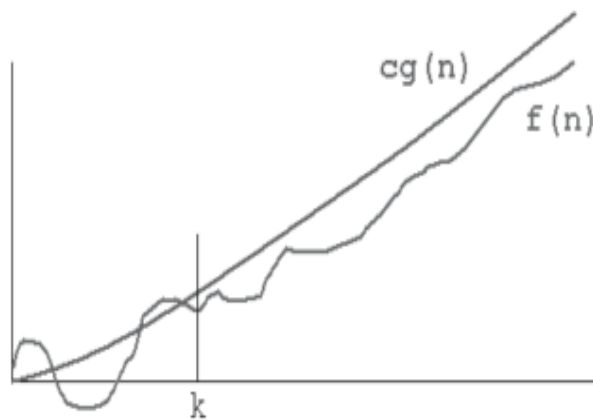
1. Pendahuluan

Setiap individu terdapat pola frekuensi suara berbeda antara satu dengan yang lainnya [1]. Pada rekaman suara, pola ciri frekuensi dapat digunakan untuk mengetahui identitas individu [2]. Ciri pada suara terdiri atas frekuensi dasar dan resonansi [3]. Frekuensi resonansi terbagi atas resonansi 1 hingga resonansi 4. Proses untuk memperoleh pola ciri dengan mengekstraksi sampel kata [4]. Salah satu metode ekstraksi ciri yang andal adalah MFCC. Tahapan MFCC secara umum terdiri atas *pre-emphasis*, *frame blocking*, *windowing*, FFT, MFCC, iFFT, Mel *Cepstrum* [5]. Pada perkembangannya metode MFCC telah banyak mengalami modifikasi, salah satunya adalah pada bagian pemisah frekuensi. Pemisahan tersebut dilakukan untuk memilah frekuensi dasar dan frekuensi resonansi. Pemilahan tersebut seperti dilakukan oleh Huizen et al. (2017) mengembangkan metode MFCC secara multikanal. MFCC multikanal memisahkan komponen

ciri menjadi lima bagian yaitu F0, F1, F2, F3 dan F4 [7]. Pengembangan metode MFCC dilakukan guna meningkatkan akurasi. Namun demikian keandalan suatu algoritma selain akurasi perlu juga dilihat berdasarkan kompleksitas. Cara mengetahui kompleksitas suatu algoritma dengan analisis fungsi asimtotik atau dikenal dengan notasi Big O [8]. Kompleksitas menghitung kecepatan komputasi (*running time*) dan kebutuhan memori. Notasi Big O dapat menjadi gambaran kompleksitas suatu algoritma, serta mendeskripsikan *running time* berdasarkan jumlah data yang diproses [9], [10]. Untuk mengetahui pengaruh pengembangan (varian) suatu algoritma perlu dilakukan uji kompleksitas. Pada penelitian ini guna mengetahui pengaruh pengembangan suatu metode dilakukan uji Big O pada varian MFCC multikanal dan MFCC duakanal.

2. Kompleksitas Algoritma

Secara teoretis asimtotik terdapat tiga notasi yaitu O (Big-oh), Ω (Omega), θ (Teta) notasi tersebut masing-masing merupakan batas atas, batas bawah dan ketiga adalah batas antara [9]. Pada notasi Big O menggambarkan pertumbuhan suatu algoritma seperti ditunjukkan pada Gambar 1. Analisis ini dapat digunakan untuk mengetahui kebutuhan memori dan waktu proses (*running time*) [8].



Gambar 1. Keluaran Notasi Big O pada suatu algoritma.
Sumber : (Devi et al. 2011)

Nilai kompleksitas diketahui dengan merepresentasikannya menggunakan sejumlah data masukan. Jumlah data sebanyak n, dan proses yang berjalan merupakan kompleksitas suatu algoritma, terhadap waktu dan kebutuhan memori [11]. Fungsi kompleksitas f(n) adalah O(g(n)) dengan konstanta positif c dan n₀ di mana f(n) ≤ c x n₀ untuk semua n ≥ n₀. Set fungsi dari O(g(n)), [9] ditunjukkan pada Persamaan (1).

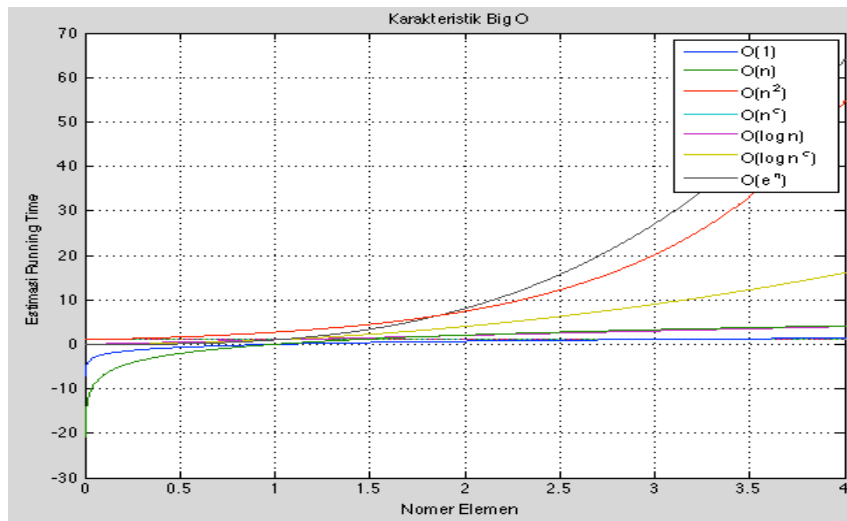
$$O(g(n)) = \begin{cases} f(n) | \exists \text{ konstanta positif } c \text{ dan } n_0 \\ \text{Sehingga } f(n) \leq c \times g(n) \forall n \geq n_0 \end{cases} \quad (1)$$

Jika suatu data dengan ukuran bebas (n) maka *running time* mempunyai nilai g(n), Gambaran keluaran menggunakan notasi Big O [8] ditunjukkan pada Tabel 1;

Tabel 1 Order Big O.

No	Order	Name
1	O(1)	Konstan
2	O(n)	Linear
3	O(n ²)	Quadratic
4	O(n ^c)	Polinomial
5	O(log n)	logaritmic
6	O((log(n)) ^c)	Polylogarithmic
7	O(C ⁿ)	Ekspensial

Running time merupakan waktu yang dibutuhkan suatu algoritma untuk menyelesaikan proses satu siklus, order dari algoritma berdasarkan sifatnya seperti ditunjukkan pada Tabel 1, sedangkan tampilan secara grafis ditunjukkan pada Gambar 2.

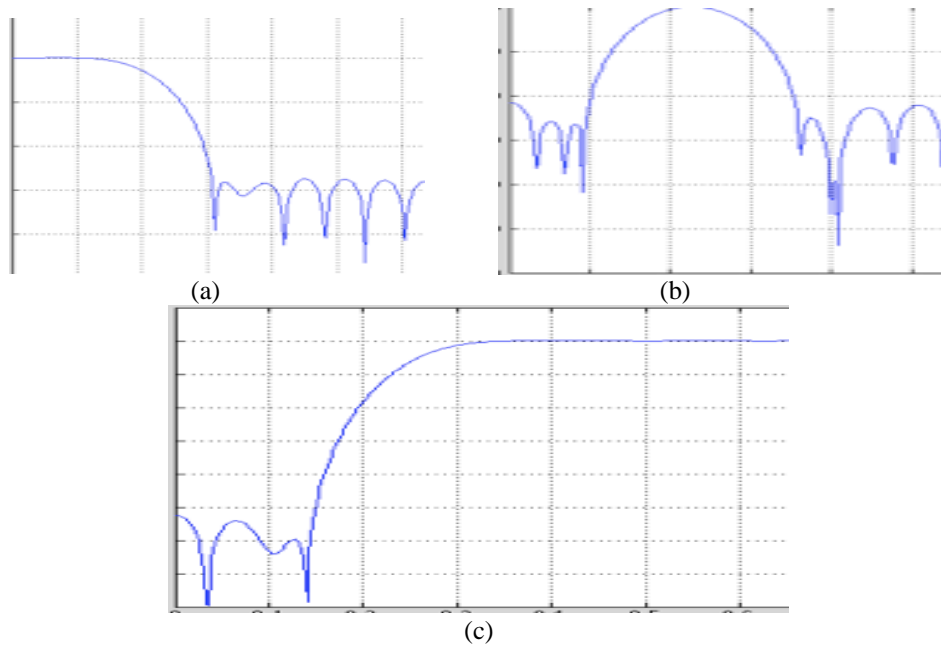


Gambar 2. Karakteristik Big O masing-masing Order.

2.1. Filter Digital

Filter digital digunakan untuk memilah frekuensi. Lebar bidang pada filter digital disesuaikan dengan kebutuhan. Frekuensi yang dikehendaki akan diteruskan dan frekuensi yang tidak dikehendaki akan direduksi. Fungsi pada filter digital ditunjukkan pada Persamaan (2) [12]. Sebagai ilustrasi untuk filter lolos bawah (LPF), filter lolos antara (BPF) dan filter lolos atas (HPF) ditunjukkan pada Gambar 3.

$$\begin{aligned}
 h[n] &= \frac{\Omega_0}{\pi} \text{sinc}(\Omega_0 n) \\
 h[n] &= \text{sinc}(\pi n) - \frac{\Omega_0}{\pi} \text{sinc}(\Omega_0 n) \\
 h_{bp} &= h_{LPH} - h_{LPL}
 \end{aligned}
 \tag{2}$$



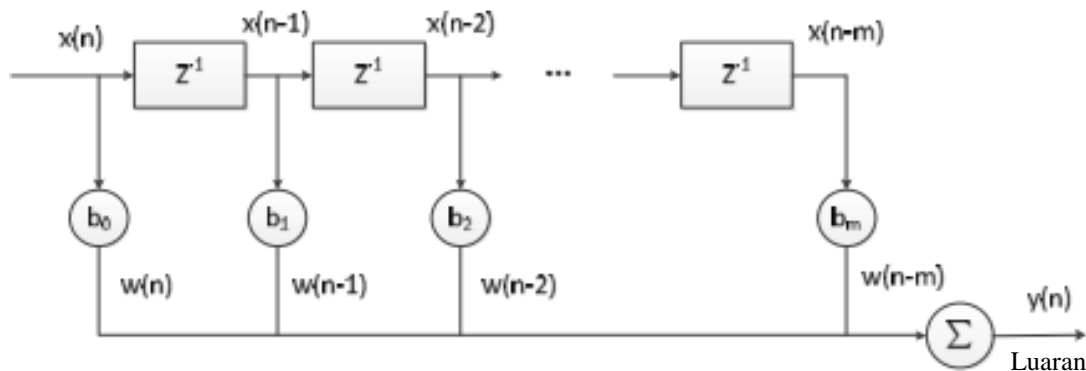
Gambar 3. Respons filter (a) LPF (b) BPF (c) HPF [13].

Gambar 3(b) merupakan filter lolos antara (BPF) yang meneruskan frekuensi antara sesuai dengan frekuensi yang dikehendaki. Pada Persamaan (2) pembentukan filter lolos antara (BPF) mengikuti frekuensi

batas bawah (fcL) dan batas atas (fcH), demikian pula untuk pembentukan LPF dan HPF [14]. Persamaan filter digital pada ranah waktu seperti ditunjukkan Persamaan (3).

$$\begin{aligned}
 y(n) &= b(1) x(n) + b(2) x(n-1) + b(3) x(n-2) \dots \\
 &\quad + b(nb+1) x(n-nb) - a(2)y(n-1) \\
 y[n] &= \sum_{k=0}^M b_k x[n - k] = \sum_{k=0}^M h[k] x[n - k]
 \end{aligned}
 \tag{3}$$

Berdasarkan Persamaan (3), ilustrasi filter digital seperti ditunjukkan pada Gambar 4. Masukan filter digital adalah sinyal diskret X(n). Setiap sinyal diskret dilakukan penundaan dan dikalikan dengan bobot. Keluaran dari filter digital berupa sinyal diskret yang telah dijumlahkan. Blok diagram filter digital dengan model FIR seperti yang ditunjukkan pada Gambar 4.



Gambar 4. Blok diagram filter FIR (Xing dan Zhang, 2011).

Filter digital pada metode ekstraksi MFCC multikanal dan MFCC dualkanal berfungsi sebagai pemilah frekuensi. Masing-masing lebar kanal dibuat berdasarkan kebutuhan untuk ekstraksi. Jenis filter yang digunakan pada MFCC multikanal dan MFCC dualkanal ditunjukkan pada Tabel 2.

Tabel 2. Jenis Filter.

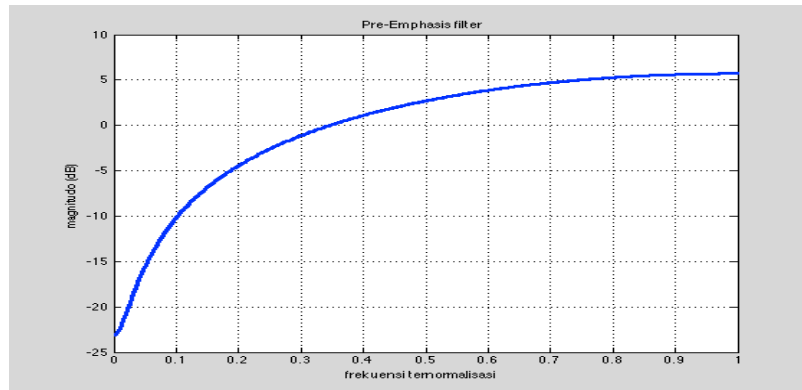
No Kanal	Jenis Filter MFCC Dualkanal	Jenis Filter MFCC Multikanal
1	LPF	LPF
2	HPF	BPF
3	-	BPF
4	-	BPF
5	-	HPF

2.2. Metode ekstraksi MFCC

Tahapan ekstraksi pada MFCC alur proses dijelaskan sebagai berikut,

- a. *Pre-emphasis* menggunakan tapis orde pertama untuk mereduksi derau aditif [15]. Derau tersebut direduksi menggunakan Persamaan (4), sedangkan untuk respons *pre-emphasis* seperti ditunjukkan pada Gambar 5.

$$s'(n) = s(n) - \bar{a} s(n - 1)
 \tag{4}$$



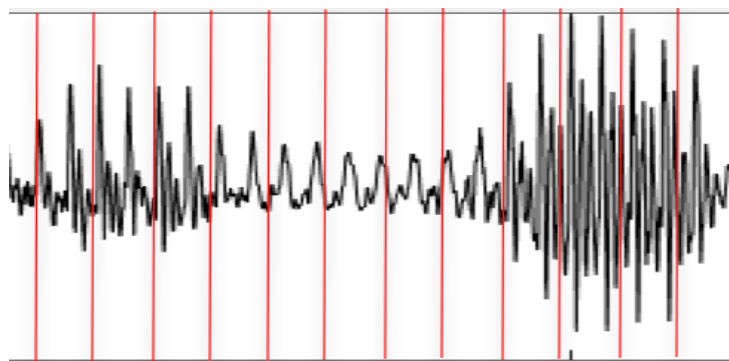
Gambar 5. Respons *pre-emphasis*

$s'(n)$ merupakan keluaran *pre-emphasis* sedangkan $s(n)$ sinyal masukan, dimana untuk n merupakan urutan dari sampel, sedangkan \bar{a} koefisien *filter pre-emphasis* dengan nilai antara 0 sampai 1 [16].

- b. *Frame-Blocking* untuk membagi sampel kata menjadi beberapa *frame*, menggunakan Persamaan (5). Pembagian sampel kata menjadi beberapa *frame* bertujuan untuk memperoleh ciri frekuensi di setiap *frame* [17].

$$f(\ell; n) = s'(n + M(L - 1)) \quad n=0,1,\dots,N-1, \quad \ell=0,1,\dots,L-1 \quad (5)$$

Ilustrasi pembagian sampel kata menjadi beberapa *frame* seperti ditunjukkan pada Gambar 6. Lebar *frame* menjadi penentu keunikan frekuensi, semakin lebar *frame* menyebabkan nilai frekuensi menjadi tidak unik. Sedangkan semakin sempit *frame* menyebabkan tidak terdapat nilai frekuensi, sehingga tidak diperoleh ciri. Lebar setiap *frame* diatur dengan lebar tertentu agar diperoleh pola frekuensi unik secara optimal [5].

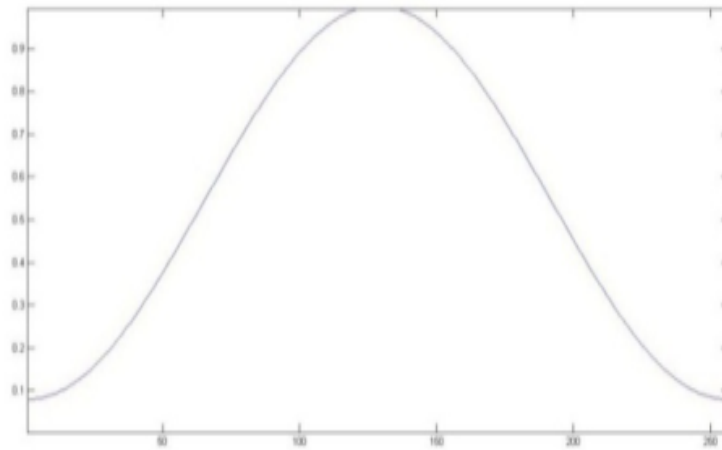


Gambar 6. *Frame-Blocking*.

- c. *Windowing* akan mengurangi kebocoran spektrum yang disebabkan rendahnya *sampling rate* di setiap *frame* [17]. Jenis *windowing* yang digunakan adalah *Hamming*, ditunjukkan pada Persamaan (6). Sebagai ilustrasi *windowing Hamming* ditunjukkan pada Gambar 7.

$$w(n) = 0,54 - 0,46 \cos\left(\frac{2\pi n}{N-1}\right) \quad 0 \leq n \leq N-1 \quad (6)$$

$$x(n) = w(n) * x(n)$$



Gambar 7. Windowing hamming.

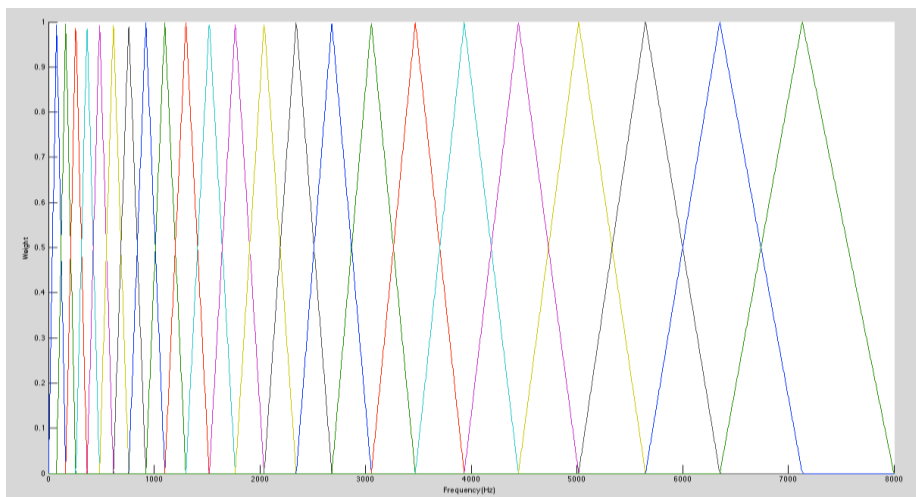
- d. DFT digunakan untuk mengubah setiap *frame* dalam ranah waktu ke ranah frekuensi [18] menggunakan Persamaan (7).

$$X_{(k)} = \sum_{n=1}^N x_{(n)} \cdot e^{-j\left[\frac{2\pi kn}{N}\right]}$$

atau

$$X_{(k)} = \sum_{n=1}^N x(n) \left[\cos\left(\frac{(2\rho nkT)}{N}\right) - j \cdot \sin\left(\frac{(2\rho nkT)}{N}\right) \right] \tag{7}$$

- e. Mel *scale filterbank* merupakan filter yang mengonversi frekuensi ke frekuensi Mel. Di mana nilai frekuensi Mel mempunyai karakteristik mendekati dengan pendengaran manusia [19]. Pendekatan frekuensi ke frekuensi Mel ditunjukkan Persamaan (8). Sebagai ilustrasi respons filter Mel ditunjukkan pada Gambar 8.



Gambar 8. Ilustrasi Skala Mel (Singh dkk., 2014).

$$f_{mel} = 2595 \cdot \log_{10} \left[1 + \frac{f_{in}}{700} \right], \text{ atau}$$

$$f_{mel} = 1127 \ln \left[1 + \frac{f_{in}}{700} \right] \tag{8}$$

Skala Mel digunakan *filterbank* dengan *filter triangular* [19], menggunakan Persamaan (9).

$$f_b(m) = f_{Mel}^{-1} \left[f_{Mel}(f_{lin-low}) + m \frac{f_{mel}(f_{lin-high}) - f_{mel}(f_{lin-low})}{P + 1} \right]$$

$$H_M(K) = \begin{cases} 0 & \text{for } k < f_b(m - 1) \\ \frac{k - f_b(m - 1)}{f_b(m) - f_b(m - 1)} & \text{for } f_b(m - 1) \leq k \leq f_b(m) \\ \frac{f_b(m - 1) - k}{f_b(m - 1) - f_b(m)} & \text{for } f_b(m) \leq k \leq f_b(m - 1) \\ 0 & \text{for } k > f_b(m - 1) \end{cases} \quad (9)$$

Bagian *cepstrum* merupakan koefisien *cepstral* yang merupakan representasi ciri, untuk memperoleh koefisien *cepstral* digunakan Persamaan (10). Jumlah koefisien yang digunakan 2 hingga 13 [10].

$$a_{mfc_n^m_chn(k)} = \ln \left(\sum_{k=1}^K |X_{n_chn}(k)|^2 H_m K_{chn} \right)$$

$$c_n^q_chn = \sum_{m=1}^P a_{mfc_n^m_chn} \cos \left[m \left(q - \frac{1}{2} \right) \frac{\pi}{P} \right] \quad (10)$$

2.3. Metode Ekstraksi MFCC-Multikanal dan MFCC Dualkanal

Varian MFCC telah banyak dikembangkan di antaranya adalah MFCC multikanal seperti yang dilakukan oleh Huizen et al. (2017). Metode tersebut digunakan untuk identifikasi rekaman suara yang terpengaruh faktor penuaan. Pada metode tersebut, proses ekstraksi menggunakan pemilah frekuensi (filter digital). Pengujian yang dilakukan belum memperhitungkan kompleksitas algoritma.

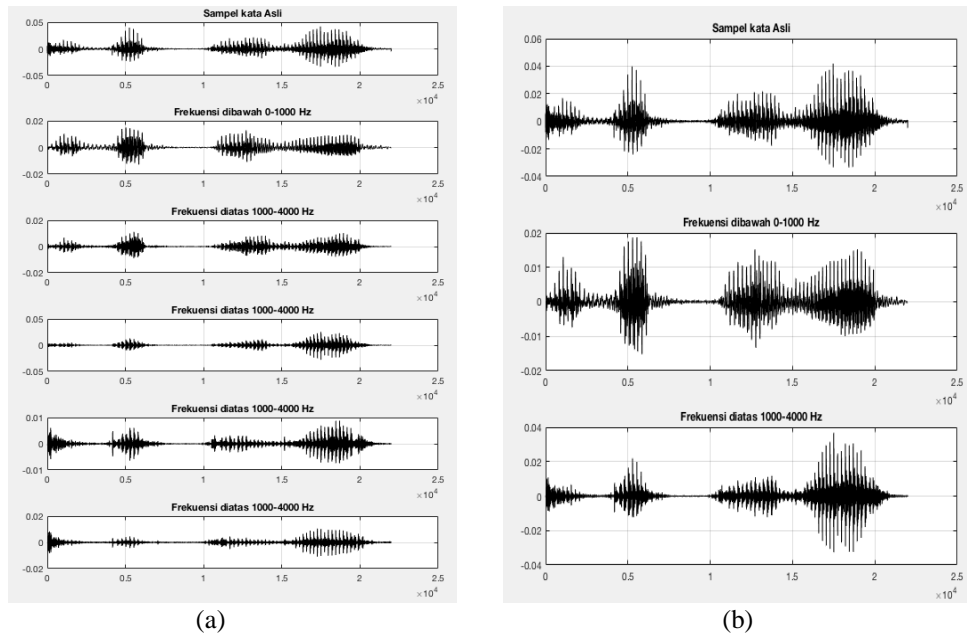
Metode ekstraksi yang akan dikembangkan berikut ini menggunakan varian MFCC menjadi dualkanal. Berikut ditunjukkan metode MFCC Multikanal dan MFCC Dualkanal pada Tabel 3.

Tabel 3. Tahapan Metode MFCC Multikanal dan MFCC Dualkanal.

MFCC Dualkanal	MFCC Multikanal (Huizen et al. 2017)
Pre-Emphasis	Pre-Emphasis
	<i>Filter FIR membentuk :</i>
Kanal 1 frekuensi dibawah 1 KHz	Kanal 1 frekuensi antara 20 Hz- 500Hz
Kanal 2 Frekuensi diatas 1 KHz	Kanal 2 frekuensi 450 Hz – 1000 Hz
	Kanal 3 frekuensi 950 Hz – 2000 Hz
	Kanal 4 frekuensi 1950 Hz – 3000 Hz
	Kanal 5 frekuensi 2950 Hz – 4000 Hz
Frame Blocking (kanal 1, kanal 2)	<i>Frame-Blocking (kanal 1, kanal 2, kanal 3, kanal 4 dan kanal 5)</i>
Windowing (kanal 1, kanal 2)	<i>Windowing (kanal 1, kanal 2, kanal 3, kanal 4 dan kanal 5)</i>
FFT (kanal 1, kanal 2)	<i>FFT (kanal 1, kanal 2, kanal 3, kanal 4 dan kanal 5)</i>
Mel Scale Filter Bank (kanal 1, kanal 2)	<i>Mel Scale Filterbank (kanal 1, kanal 2, kanal 3, kanal 4 dan kanal 5) varian tiap-tiap kanal M5FB, M2FB dan M1FB</i>
IDFT (kanal 1, kanal 2)	<i>IDFT (kanal 1, kanal 2, kanal 3, kanal 4 dan kanal 5)</i>
Cepstrum (kanal 1, kanal 2)	<i>Cepstrum (kanal 1, kanal 2, kanal 3, kanal 4 dan kanal 5)</i>

Perbedaan kedua metode terletak pada proses pemilahan frekuensi. Pada metode MFCC multikanal pemilah frekuensi menggunakan 5 filter. Sedangkan pada metode MFCC dualkanal menggunakan 2 jenis filter. Hasil pemilahan sampel kata dengan menggunakan MFCC multikanal dan

MFCC dualkanal ditunjukkan pada Gambar 9. Keluaran filter digital, masing-masing di ekstraksi dengan mengacu pada Tabel 2.



Gambar 9. Keluaran Filter dengan Metode (a) MFCC multikanal (b). MFCC dualkanal.

3. Hasil dan Analisis

Pengujian metode ekstraksi MFCC multikanal dan MFCC dualkanal untuk mengetahui kompleksitas algoritma dengan menggunakan beberapa varian data. Hasil pengujian menggunakan data tunggal untuk MFCC dualkanal ditunjukkan pada Tabel 4. Sedangkan untuk MFCC multikanal hasilnya ditunjukkan pada Tabel 5.

Tabel 4. Pengujian Kompleksitas Algoritma untuk MFCC dualkanal.

MFCC dualkanal	Running Time (detik)
Pre-Emphasis	0.002
Kanal 1 frekuensi dibawah 1 KHz Kanal 2 Frekuensi diatas 1 KHz	0.0024
MFCC Kanal1, Kanal 2	0.003

Tabel 5. Pengujian Kompleksitas Algoritma untuk MFCC Multikanal.

MFCC multikanal (Huizen et al. 2017)	Running Time (detik)
Pre-Emphasis	0.007
Filter FIR membentuk : Kanal 1 frekuensi antara 20 Hz- 500Hz Kanal 2 frekuensi 450 Hz – 1000 Hz Kanal 3 frekuensi 950 Hz – 2000 Hz Kanal 4 frekuensi 1950 Hz – 3000 Hz Kanal 5 frekuensi 2950 Hz – 4000 Hz	0.004
MFCC Kanal1, Kanal 2, Kanal3, Kanal 4, Kanal 5	0.078

Pengujian menggunakan beberapa varian data untuk keseluruhan proses metode ekstraksi MFCC multikanal dan MFCC dualkanal hasilnya ditunjukkan pada Tabel 6.

Tabel 6. Pengujian Kompleksitas Algoritma untuk MFCC multikanal dan MFCC dualkanal.

Jumlah data	MFCC multikanal (detik)	MFCC dualkanal (detik)
1	0.077	0.040
10	0.686	0.381
20	1.309	0.729
40	2.498	1.368
100	6.273	3.390
200	11.758	6.382
400	23.721	12.584

Hasil pengujian menunjukkan bahwa banyaknya varian pada metode MFCC multikanal menyebabkan *running time* lebih lama dibandingkan dengan algoritma kompleksitas rendah. Seperti hasil ditunjukkan pada Tabel 3 dan 4. Pengujian berikutnya dengan menggunakan varian data hingga 400 sampel kata. Terdapat perbedaan waktu yang cukup tinggi antar keduanya. Agar pengembangan suatu algoritma dapat andal perlu adanya keselarasan memperhatikan kompleksitas dan nilai akurasi kecocokan hasil identifikasi.

4. Kesimpulan

Berdasarkan hasil pengujian dapat disimpulkan sebagai berikut;

1. Pada varian MFCC multikanal *running time* terendah adalah 0.077 detik dan tertinggi 23.721 detik dengan menggunakan varian data uji 1 sampai dengan 400 sampel kata.
2. Pada MFCC dualkanal, *running time* terendah 0.0040 detik dan tertinggi 12.58 detik dengan menggunakan varian data uji 1 sampai dengan 400 sampel kata
3. MFCC Multikanal mempunyai *running time* lebih tinggi dibandingkan MFCC Dualkanal. Agar *running time* tidak terlalu tinggi, pengembangan metode MFCC perlu memperhatikan kompleksitas algoritma sehingga nilai akurasi dan kompleksitas dapat selaras.

Daftar Pustaka

- [1] R. G. Hautamaki, M. Sahidullah, T. Kinnunen, and V. Hautam, "Age-Related Voice Disguise and its Impact on Speaker Verification Accuracy," *Conf. Speak. Lang. Recognit. Work. Speak. Lang. Recognit. Work.*, pp. 277–282, 2016.
- [2] P. Qi and L. Wang, "Experiments of GMM based speaker identification," *2011 8th Int. Conf. Ubiquitous Robot. Ambient Intell.*, pp. 26–31, Nov. 2011.
- [3] U. Reubold, J. Harrington, and F. Kleber, "Vocal aging effects on F0 and the first formant: A longitudinal analysis in adult speakers," *Speech Commun.*, vol. 52, no. 7–8, pp. 638–651, Jul. 2010.
- [4] M. J. Alam, T. Kinnunen, P. Kenny, P. Ouellet, and D. O'Shaughnessy, "Multitaper MFCC and PLP features for speaker verification using i-vectors," *Speech Commun.*, vol. 55, no. 2, pp. 237–251, Feb. 2013.
- [5] A. Buono, W. Jatmiko, B. Kusumoputro, D. I. Komputer, K. I. P. B. Darmaga, L. K. Komputasional, F. I. Komputer, and U. Indonesia, "SEBAGAI ESKTRAKSI CIRI PADA SISTEM IDENTIFIKASI PEMBICARA MENGGUNAKAN HIDDEN MARKOV MODEL (HMM)," vol. 13, no. 1, pp. 87–93, 2009.
- [6] R. R. Huizen, J. E. Istiyanto, and A. E. Putra, "Identification of Voice Utterance with Aging Factor Using the Method of MFCC Multichannel," *Int. J. Adv. Stud. Comput. Sci. Eng.*, vol. 6, no. 1, pp. 14–19, 2017.
- [7] S. B. Zinnat, R. Marzia, and A. Siddique, "Automatic Word Recognition for Bangla Spoken Language," *Int. Conf. Signal Propag. Comput. Technol.*, pp. 470–475, 2014.
- [8] S. G. Devi, K. Selvam, and S. P. Rajagopalan, "An Abstract to Calculate Big O Factors of Time and Space Complexity of Machine Code," no. Seiscon, pp. 844–847, 2011.
- [9] L. Rahman, S. A. Khan, A. Kabir, and S. Miah, "A Model for Set Theoretic Analysis of Algorithm With Asymptotic Perspectives," no. 3, 2013.
- [10] O. Chia Ai, M. Hariharan, S. Yaacob, and L. Sin Chee, "Classification of speech dysfluencies with MFCC and LPCC features," *Expert Syst. Appl.*, vol. 39, no. 2, pp. 2157–2165, Feb. 2012.
- [11] R. A. Nim, "Analisis Kecepatan Sorting Dengan Notasi Big O."
- [12] K. R. Borisagar and G. R. Kulkarni, "Simulation and Performance Analysis of Adaptive Filter in Real Time Noise over Conventional Fixed Filter," *2012 Int. Conf. Commun. Syst. Netw. Technol.*, pp. 621–624, May 2012.
- [13] G. Xing and Y. Zhang, "Analysis and comparison of RLS adaptive filter in signal De-noising," *2011*

- Int. Conf. Electr. Control Eng.*, pp. 5754–5758, Sep. 2011.
- [14] X. Qiu and L. Wu, “Active impulsive noise control algorithm with post adaptive filter coefficient filtering,” *IET Signal Process.*, vol. 7, no. 6, pp. 515–521, Aug. 2013.
- [15] X. Zhao and D. Wang, “Analyzing Noise Robustness Of MFCC And GFCC Features In Speaker Identification,” pp. 7204–7208, 2013.
- [16] W. C. Hsu, W. H. Lai, and W. P. Hong, “Usefulness of residual-based features in speaker verification and their combination way with linear prediction coefficients,” *Proc. ISM Work. 2007 9th IEEE Int. Symp. Multimed. - Work.*, pp. 246–251, 2007.
- [17] P. M. Chauhan, “Mel Frequency Cepstral Coefficients (MFCC) Based Speaker Identification in Noisy Environment Using Wiener Filter,” *Green Comput. Commun. Electr. Eng. (ICGCCEE)*, pp. 1–5, 2014.
- [18] A. S. Bhalerao, “Implementation of Automatic Speaker Recognition on TMS320C6713 Using MFCC,” pp. 4–7, 2013.
- [19] Abriyono and A. Harjoko, “Pengenalan Ucapan Suku Kata Bahasa Lisan Menggunakan Ciri LPC, MFCC, dan JST,” *IJCCS*, vol. 6, no. 2, pp. 23–34, 2012.
- [20] A. K. Singh, R. Singh, and A. Dwivedi, “Mel Frequency Cepstral Coefficients Based Text Independent Automatic Speaker Recognition Using Matlab,” pp. 524–527, 2014.