Indonesian Journal of Electronics and Instrumentation Systems (IJEIS)
Vol.11, No.2, October 2021, pp. 155~166
ISSN (print): 2088-3714, ISSN (online): 2460-7681
DOI: https://doi.org/10.22146/ijeis.53437

Deteksi Kesalahan Pengucapan Huruf Jawa *Carakan* dengan Jaringan Syaraf Tiruan Perambatan Balik

JK Aditya Christya Buditama*[1], Catur Atmaji [2], Agfianto Eko Putra[3]
[1]Prodi Elektronika dan Instrumentasi, DIKE, FMIPA UGM, Yogyakarta, Indonesia
[2]Departemen Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta, Indonesia
e-mail: ***[1]jkadityacb@gmail.com,** [2]catur_atmaji@ugm.ac.id, [3]agfi@ugm.ac.id

Abstrak

Bahasa Jawa merupakan kebudayaan bangsa Indonesia yang perlu dilestarikan dan dipelajari, tetapi banyak pelajar bahasa Jawa yang melakukan kesalahan dalam pengucapan huruf Jawa dan sulit menganilisis kesalahan oleh pengajar manusia karena sulit ditemukan dan mahal, sehingga dibutuhkan deteksi kesalahan pengucapan huruf Jawa. Sistem sejenis telah diterapkan dalam berbagai bahasa, tetapi belum diimplementasi untuk huruf Jawa carakan. Penelitian ini mengembangkan deteksi kesalahan pengucapan huruf Jawa dengan Jaringan Syaraf Tiruan Perambatan Balik (JST-PB). Data berupa sinyal pengucapan teks yang telah ditentukan oleh 24 penutur sebanyak 5 kali. Metode ALNS kemudian digunakan sebagai segmentasi otomatis sinyal ke silabel. JST-PB menerima masukan nilai statistik dari Mel-Frequency Cepstral Coefficient (MFCC) dengan 7 dan 14 koefisien. 10-Fold Cross Validation digunakan untuk validasi data dan menguji secara keseluruhan. Deteksi kesalahan pengucapan huruf Jawa carakan dengan 7 koefisien MFCC menghasilkan akurasi tertinggi sebesar 80,07%. Sedangkan deteksi kesalahan pengucap huruf Jawa carakan dengan 14 koefisien MFCC menghasilkan akurasi 82,36%.

Kata kunci*— MFCC, JST-PB, kesalahan ucapan, huruf Jawa.*

Abstract

Javanese is an Indonesian culture which needs to be preserved, but many Javanese students make mistakes in the pronunciation of Javanese letters and find it difficult to analyze errors by human teachers because of the limited time and subjective assessment, so a system is needed to detect incorrect pronunciation of Javanese letters. Mispronunciation detection system has been widely applied in foreign languages, but the system has not been implemented for Javanese carakan letters. This research develops the Javanese letters mispronunciation detection system using Back-Propagation Artificial Neural Networks (BP-ANN). The dataset is obtained from the recorded pronunciation of hanacaraka texts by 24 speakers with 5 repetitions. ALNS method then used to automatically segment the signal into syllables. ANN-PB use statistical value of Mel-Frequency Cepstral Coefficient (MFCC) method with 7 and 14 coefficients. 10-Fold Cross Validation is used to validate and test the system. The Javanese mispronunciation detection using 7MFCC coefficients produces the highest accuracy of 80,07%. While the Javanese mispronunciation detection using 14 MFCC coefficients produces an accuracy of 82.36% at the highest.

Keywords— *MFCC, BP-ANN, Mispronunciartion, Javanese letters*

Received January 15th,2020; Revised October 27th, 2021; Accepted October 31th, 2021

1. PENDAHULUAN

Bahasa Jawa merupakan bahasa yang berkembang di wilayah Pulau Jawa, Indonesia. Menurut [1], penggunaan bahasa Jawa masih banyak diterapkan pada komunikasi sehari-hari, tetapi tidak dapat dipungkiri apabila terjadi penurunan jumlah penggunaan bahasa Jawa. Salah satu alasan menurunnya pengunaan bahasa Jawa adalah struktur penulisan huruf Jawa yang sulit. Struktur penulisan dalam bahasa Jawa sejatinya menggunakan huruf *carakan*, yang dikenal juga dengan *hanacaraka*. Hal ini menyebabkan diperlukannya proses studi khusus dalam mempelajari pengucapan huruf *carakan* yang baik dan benar, hanya saja jumlah pengajar manusia yang ada sangat terbatas. Pengajar manusia juga membutuhkan biaya yang mahal dan penilaiannya bersifat subjektif sesuai pada penelitian [2].

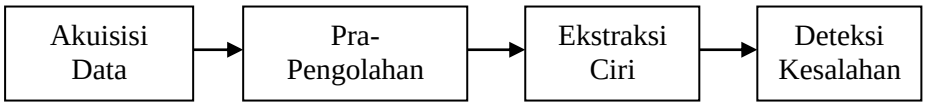
Berbagai teknik pemrosesan sinyal digital telah banyak diimplementasikan untuk mendeteksi kesalahan pengucapan untuk bahasa Inggris. Pada penelitian [3], deteksi kesalahan pengucapan Bahasa Inggris dilakukan dengan analisis intonasi dan ritme sinyal suara. Deteksi kesalahan pengucapan lainnya adalah dengan memanfaatkan SVM yang dilatih untuk prediksi kesalahan pengucapan pada 5 fonem, yakni: Th, S, Jh, Ch, dan Zh [4]. Walaupun demikian, deteksi menjadi hanya bekerja pada ruang lingkup kecil. Hal ini juga terjadi pada [5] yang secara spesifik mendeteksi kesalahan pengucapan *deletion error* pada 1 fonem, yakni "*g-dropping"* pada pengucapan Bahasa Inggris.

Selain pengucapan formal, beberapa sistem serupa bahkan diterapkan pada pengucapan non-formal seperti yang dilakukan pada penelitian [6]. Sistem yang dirancang bertujuan mendeteksi kesalahan pengucapan bahasa Inggris pada suatu lagu yang berjalan. Hasil dari sistem pada [7] menunjukkan bahwa jumlah kesalahan pengucapan yang dideteksi pada lagu mengalami peningkatan. Pada deteksi kesalahan pengucapan Bahasa Inggris, nilai akurasi tertinggi diperoleh [7] mencapai lebih dari 98%. Menurut [7], proses deteksi dimulai dengan memanfaatkan analisis waktu-frekuensi untuk melakukan ekstraksi fitur dengan algoritma viterbi dari sinyal pengucapan dan dilanjutkan dengan pengenalan pola pengucapan yang benar dengan JST yang diperkuat dalam proses klasifikasinya dengan PID-Fuzzy. Dibandingkan dengan penelitian-penelitian sebelumnya, dapat disimpulkan bahwa JST memberikan nilai akurasi terbaik, sehingga metode JST akan dipakai dalam penelitian ini.

Walaupun demikian, seluruh sistem tersebut diterapkan untuk bahasa Inggris dan belum untuk bahasa Jawa. Metode jaringan syaraf tiruan telah diimplementasikan untuk klasifikasi pengucapan fonem vokal bahasa Jawa pada [8], tetapi belum digunakan untuk deteksi pengucapan huruf Jawa *carakan*, sehingga diperlukan penelitian lebih lanjut untuk mengimplementasikan jaringan syaraf tiruan perambatan balik (JST-PB) untuk deteksi kesalahan pengucapan huruf Jawa *carakan*.

2. METODE PENELITIAN

Deteksi yang akan dirancang memiliki fungsi untuk mampu mendeteksi kesalahan pengucapan huruf Jawa dengan metode JST perambatan balik. Proses deteksi yang akan dirancang akan menerima masukan berupa suara pengucapan, membandingkan ciri-ciri data dengan ciri-ciri dari data set yang telah dilatih sebelumnya dengan JST perambatan balik, dan memberikan keluaran berupa bilangan biner yang merepresentasikan apakah pengucapan yang dilakuakan adalah benar atau salah (1 atau 0). Proses deteksi dilakukan melalui 4 tahap utama, yakni: akuisisi data, pra-pengolahan, ekstraksi fitur, dan deteksi kesalahan pengucapan. Secara singkat, proses deteksi dapat dijelaskan dengan blok diagram pada Gambar 1.



Gambar 1 Blok diagram proses deteksi

2.1 Akuisisi Data

Tahap pertama adalah akuisisi data set, baik data latih maupun data uji. Data berupa sinyal suara pengucapan teks *hanacaraka* di lingkungan tertutup. Jumlah fonem yang diamati dibatasi hingga 20 fonem. Angka ini ditentukan untuk menghindari ruang lingkup yang terlalu kecil sesuai pada [6]. Data diambil dari penutur sejumlah 24 orang (12 laki-laki dan 12 perempuan) dengan pengulangan sebanyak 5 kali. Jumlah data yang akan digunakan sebanyak 120 data dalam format .wav dengan frekuensi *sampling* 41,1 kHz, pada *channel mono* .

Setiap data dinilai dan ditentukan oleh ahli linguistik sebagai pengucapan yang salah (positif) dan benar (negatif). Ahli linguistik berasal dari *abdi dalem* KHP Krida Mardawa, Kraton Yogyakarta berjumlah 1 orang. Berdasarkan jenisnya, kesalahan pengucapan fonem dibagi menjadi 3 jenis, yaitu: *phoneme substitution, redundant phoneme,* dan *omitted phoneme* [4]. *Phoneme substitution* terjadi ketika fonem konsonan yang diucapkan berubah dari seharusnya. *Redundant phoneme* terjadi ketika terdapat pengucapan fonem yang tidak perlu dari pengucapan seharusnya. *Omitted phoneme* terjadi apabila terdapat pengucapan fonem yang menghilang dari seharusnya.

2.2 Pra-Pengolahan

Data set hasil akuisisi tersimpan dalam bentuk kalimat sehingga perlu disegmentasi secara otomatis untuk dapat digunakan dalam membangun deteksi kesalahan pengucapan. Sesuai pada [9], teknik ALNS mampu memberikan performa segmentasi silabel dengan baik, sehingga diterapkan pada penelitian ini.

Dalam pra-pengolahan, sinyal hasil akuisisi data akan mengalami 2 proses utama, yakni *pre-emphasis* dan segmentasi. Proses *pre-emphasis* berfungsi sebagai tapis *high-pass* bertujuan untuk menghilangkan sinyal *noise* yang masih terkandung dalam data. Nilai koefisien α yang digunakan dalam *pre-emphasis* untuk segmentasi adalah 0.9.

Segmentasi dilakukan dengan maksud memotong sinyal suara ke dalam silabel. Ada beberapa tahap yang perlu dilakukan pada proses segmentasi sesuai pada penelitian [9]. Sinyal diblok ke dalam beberapa *frame* pendek dengan menggunakan jendela Hamming berdurasi 10 milidetik (440 sampel data). Setiap *frame* memiliki tingkat *overlapping* sebesar 75% (330 sampel data). Nilai energi jangka pendek (STE) kemudian didapatkan dengan nilai kuadrat dari sinyal pada setiap *frame* sesuai persamaan (1).

$$E = \sum_{i=1}^{N} S_i^2 \quad (1)$$

Normalisasi ditambahkan untuk mendeteksi *frame-frame* dengan tingkat energi yang sangat rendah dan menormalisasikannya menurut *frame* dengan tingkat energi tinggi. Logika *fuzzy* selanjutnya digunakan untuk menghasilkan kontur energi jangka pendek yang lebih lembut (*smoothed*). Logika *Fuzzy* yang digunakan memiliki 11 aturan (A) sesuai pada **Tabel 1**. Masukan dari *fuzzy* (x_i) adalah nilai energi pada 7 frame sebelumnya yang . Setiap fungsi keanggotaan logika *fuzzy* memiliki nilai tengah (C_A) dan lebar keanggotaan sebesar w=0,36 Jarak keanggotaan tiap aturan *fuzzy* berada pada sepanjang $w/2$ di kiri dan kanan dari C_A. Fungsi keanggotaan *fuzzy* yag digunakan sesuai pada persamaan (2).

Tabel 1 Aturan Linguistik Logika *Fuzzy*

No.	Aturan Linguistik *Fuzzy*
1.	Jika “most” masukan adalah sangat kecil positif, maka keluaran sangat kecil positif.
2.	Jika “most” masukan adalah kecil positif, maka keluaran kecil positif.
3.	Jika “most” masukan adalah sedang positif, maka keluaran sedang positif.
4.	Jika “most” masukan adalah besar positif, maka keluaran besar positif.
5.	Jika “most” masukan adalah sangat besar positif, maka keluaran sangat besar positif.

Tabel 1 (Lanjutan)

No.	Aturan Linguistik *Fuzzy*
6.	Jika “most” masukan adalah sangat kecil negatif, maka keluaran sangat kecil negatif.
7.	Jika “most” masukan adalah kecil negatif, maka keluaran kecil negatif.
8.	Jika “most” masukan adalah sedang negatif, maka keluaran sedang negatif.
9.	Jika “most” masukan adalah besar negatif, maka keluaran besar negatif.
10.	Jika “most” masukan adalah sangat besar negatif, maka keluaran sangat besar negatif
11.	Selain itu, keluaran nol.

$$\mu_A(x_i) = \begin{cases} \dfrac{-2(x_i - C_A)}{w+1}, & C_A - \dfrac{w}{2} < x_i < C_A \\ \dfrac{2(x_i - C_A)}{w+1}, & C_A < x_i < C_A + \dfrac{w}{2} \\ 0, & Otherwise \end{cases} \quad (2)$$

Nilai fungsi linguistik “most” dari setiap fungsi aturan *fuzzy* kemudian didapatkan sesuai persamaan (3). Masukan dari fungsi linguistik “most” adalah hasil perbandingan jumlah anggota kelas fuzzy dengan jumlah keseluruhan data yang diamati.

$$\mu_{most}(z) = \begin{cases} 0, & z \leq 0{,}1 \\ 0{,}5\left(1 - cos\left[\dfrac{\pi(z-1)}{0{,}8}\right]\right), & 0{,}1 < z < 0{,}9 \\ 1, & z \geq 0{,}9 \end{cases} \quad (3)$$

Derajat keaktifan setiap fungsi aturan dihitung dari mencari hasil produk dari nilai median seluruh nilai keanggotaannya dengan nilai dari fungsi linguistik “most” yang didapatkan. Proses penghitungan derajat keaktifan ditunjukkan pada Persamaan (4).

$$\lambda_A = median[\mu_A(x_i) : x_i \in A] * \mu_{most}\left[\frac{jumlah\ x_i \in A}{jumlah\ total\ x_i}\right] \quad (4)$$

Hasil produk korelasi dari nilai derajat keaktifan seluruh kelompok keanggotaan digunakan untuk mencari nilai perubahan energi jangka pendek (ΔE). Nilai ΔE dijumlahkan dengan nilai energi pada *frame* awal untuk menentukan perubahan energi pada frame selanjutnya untuk melembutkan kontur energinya. Perubahan nilai energi ditunjukkan pada Persamaan (5). Pada hasil dari *fuzzy-smoothing* seringkali ditemukan puncak kecil di antara dua suku kata. Hal ini menyebabkan puncak tersebut dianggap sebagai *local maxima.* Untuk menghindari hal tersebut, setiap 7 data *slope* dari kontur diperlembut kembali dengan mengambil nilai *mean* dari 7 data di setiap *slope*.

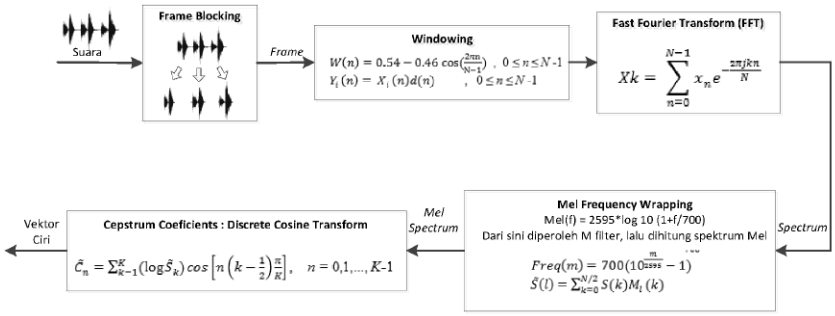
$$\Delta E = \sum_{A=1}^{11} \lambda_A C_A \quad (5)$$

Terakhir, metode *local minima* dipakai untuk menentukan batasan dari sinyal ucapan agar menentukan batasan setiap suku kata berdasarkan kontur energi hasil *fuzzy-smoothing*. Dalam *local minima* terdapat 3 parameter yang digunakan sebagai acuan penentuan batas silabel, yakni D1, D2, dan Th yang ditentukan melalui proses percobaan. D1 adalah jumlah

frame di sebelah kiri dan kanan dari sampel energi untuk menentukan jika sampel tersebut merupakan nilai energi puncak atau tidak. D2 adalah jumlah *frame* di kiri dan kanan dari sampel energi untuk menentukan jika suatu titik sampel *local minima* merupakan batas silabel atau tidak. Th adalah nilai batas rasio antara suatu titik energi puncak dengan titik energi minimum yang bersangkutan untuk menetukan jika titik energi puncak tersebut merupakan *local maxima*. Nilai parameter yang digunakan dalam mengamati pengucapan huruf Jawa adalah D1=16, D2=25, dan Th= 1.5. Keluaran dari tahap ini adalah lokasi di mana pengucapan suku kata selesai. Sinyal suara kemudian dipotong sesuai batas yang telah ditentukan. Hasil dari proses pra-pengolahan adalah sinyal suara yang telah dipotong ke dalam silabel.

2.3 Ekstraksi Ciri

Hasil dari sinyal yang telah melalui proses pra-pengolahan kemudian menjadi masukan untuk MFCC. Proses kalkulasi MFCC terdiri dari beberapa tahap sekuensial dimulai dari *pre-emphasis*, *frame blocking*, *windowing*, *Fast Fourier Transform* (FFT), *Mel-filterbank*, log, dan *discrete cosine transform* (DCT) [10]. Secara singkat proses ekstraksi ciri akustik dengan MFCC ditunjukkan pada Gambar 2.



Gambar 2 Diagram proses MFCC [10]

. Jumlah sinyal suara sebagai masukan MFCC adalah 2400 sinyal (24 penutur × 5 pengulangan × 20 huruf Jawa). MFCC diawali dengan kembali menerapkan *pre-emphasis*. Koefisien α pada *pre-emphasis* untuk MFCC yang digunakan sebesar 0.97. Sinyal hasil *pre-emphasis* kemudian dipisahkan ke dalam *frame-frame* kecil berdurasi 20 milidetik (880 sampel data). Setiap *frame* mengalami *overlapping* sebesar 50% (440 sampel data). *Windowing* dengan jendela *Hamming* kemudian diterapkan ke setiap *frame*. *Windowing* dilakukan untuk mengatasi sinyal tidak kontinyu pada awal dan akhir *frame* yang disebabkan oleh proses pemotongan sinyal sehingga mengatasi terjadinya *aliasing* sesuai pada [11].

Kemudian dilakukan FFT untuk mengubah sinyal ke dalam domain frekuensi dengan sebelumnya menghitung panjang nilai analisa sebesar 2^n. Panjang data sampel setiap *frame* yang digunakan adalah 880 sampel data sehingga panjang nilai analisa FFT sebesar 2^{10} = 2048, nilai pangkat 2 terdekat dari panjang data sampel setiap *frame*. Tahap berikutnya adalaha menapis setiap sinyal yang berada dalam domain frekuensi ke dalam skala *mel* menggunakan tapis *filterbank*. Jumlah tapis *filterbank* yang digunakan pada penelitian ini adalah 7 atau 14. Tujuan melakukan variasi yaitu untuk mengetahui jumlah koefisien yang bekerja paling optimal untuk identifikasi oleh JST perambatan balik. Tahap terakhir dari MFCC adalah logaritma cepstrum dan DCT untuk mengembalikan data ke dalam ranah waktu.

Hasil akhir ekstraksi ciri MFCC adalah koefisien-koefisien dari sinyal ucapan huruf Jawa yang disimpan dalam bentuk matriks. Dimensi matriks MFCC ditentikan oleh jumlah *frame* dan koefisien MFCC yang digunakan. Oleh sebab itu, ukuran dari matriks hasil MFCC akan selalu berbeda pada data berdurasi berbeda. Untuk menyamakan ukuran matriks yang menjadi masukan dalam JST perambatan balik, 6 nilai statistik dari matriks MFCC tersebut diambil. Nilai statistik yang digunakan adalah *mean*, *median*, *max*, *min*, *sum*, dan standar deviasi. Keluaran dari tahap ekstraksi ciri adalah vektor akustik sepanjang 42 dan 84 ciri.

2.4. Deteksi Kesalahan Pengucapan Huruf Jawa Carakan

Deteksi dilakukan menggunakan 20 jaringan syaraf tiruan perambatan balik dengan arsitektur identik. JST-PB mendeteksi kesalahan pengucapan dengan memberikan nilai "1". Nilai "0" diberikan apabila JST-PB menilai pengucapan yang dilakukan dinilai benar. Jumlah data yang digunakan adalah 2400 data dengan 90% data sebagai data latih dan 10% data sebagai data uji yang ditentukan melalui *k-fold cross validation* (k=10). Fungsi aktivasi yang digunakan adalah fungsi *sigmoid* bipolar pada lapis tersembunyi dan fungsi identitas pada lapis keluaran. Nilai bobot awal dari setiap masukan akan diberikan secara acak.

Seluruh proses akan dilakukan dengan batas *epoch* maksimal mencapai 10.000 kali dengan target *error* minimum sebesar 0,01. Algoritma pembelajaran yang digunakan adalah *gradient descent momentum* dengan *learning rate* adaptif. *Learning rate* yang digunakan memiliki nilai awal sebesar 0,01. Fungsi *error* yang dipakai adalah *mean-squared error*. Deteksi kesalahan pengucapan dibagi ke dalam 2 sesi berdasarkan jumlah koefisien MFCC yang digunakan (7 atau 14 koefisien). Deteksi kesalahan pengucapan menggunakan 7 koefisien MFCC untuk sesi pertama dan 14 koefisien MFCC untuk sesi kedua. JST-PB pada sesi pertama memiliki jumlah masukan 42 vektor ciri, sementara JST-PB pada sesi kedua memiliki jumlah masukan 84 vektor ciri.

3. HASIL DAN PEMBAHASAN

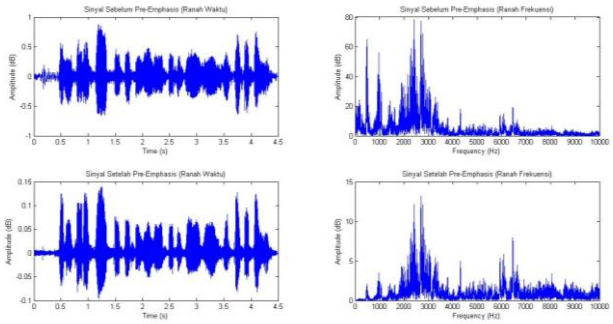
3.1 Hasil Akuisisi Data

Data yang diperoleh dari akuisisi data adalah sinyal suara ucapan hasil perekaman 20 huruf Jawa *carakan* di dalam naskah dengan durasi waktu yang berbeda dan frekuensi pencuplikan sebesar 44.100 Hz. Banyak data yang diperoleh, yaitu sebanyak 24 penutur dengan perulangan lima kali, sehingga diperoleh 120 data rekaman. Penutur pada penelitian ini terdiri dari 12 orang laki-laki dan 12 orang perempuan. Setengah dari penutur memiliki pengetahuan dasar dan mampu berkomunikasi secara fasih dalam Bahasa Jawa. Seluruh penutur berada dalam kondisi yang normal, tenang, dan sehat. Data hasil rekaman disimpan dalam bentuk .wav agar dapat menjadi masukan ke perangkat lunak MATLAB.

Seluruh data kemudian dievaluasi oleh ahli linguistik. Data ditentukan bernilai positif apabila ditemukan kesalahan dalam pengucapannya oleh ahli linguistik yang berasal dari *abdi dalem* Kraton Yogyakarta. Secara keseluruhan, hasil penilaian oleh ahli linguistik mendeteksi sebanyak 45.711 kesalahan pada seluruh huruf yang terdiri dari 3 jenis kesalahan pengucapan, yakni *phoneme substitution, omitted phoneme,* dan *redundant phoneme. Phoneme substitution* paling banyak terjadi pada pengucapan huruf [La] sebanyak 2.292 *error*. *Redundant phoneme* paling banyak terjadi pada pengucapan huruf [Ta] sebanyak 138 *error*. *Omitted phoneme* paling banyak terjadi pada pengucapan huruf [Tha] sebanyak 345 *error*.

3.2 Hasil Pra-Pengolahan

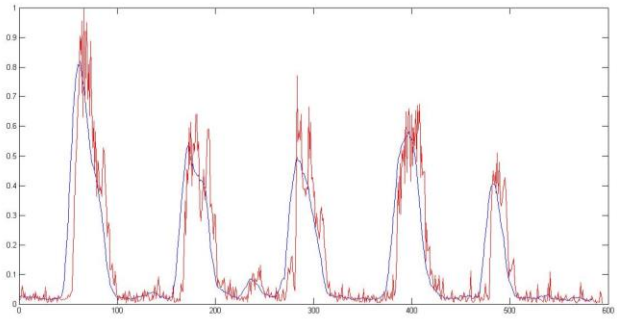
Proses pra-pengolahan secara garis besar, dibagi ke dalam 2 tahap utama, yaitu *pre-emphasis* dan segmentasi otomatis. Pra-pengolahan diawali dengan menapis sinyal suara ucapan menggunakan tapis *pre-emphasis* untuk mempertahankan frekuensi-frekuensi tinggi pada spektrum, dimana pada daerah rendah bentuk spektrum relatif lebih tinggi dan mengalami penurunan pada daerah tinggi. Setelah dilakukan *preemphasis* terlihat distribusi energi pada setiap frekuensi menjadi lebih seimbang dan bentuk sinyal semakin halus. Hal tersebut dapat dilihat pada Gambar 3.



Gambar 3 Hasil *Pre-Emphasis* Sinyal

Selanjutnya segmentasi otomatis dilakukan sesuai [9]. Selanjutnya sinyal hasil *pre emphasis* akan disegmentasi berdasarkan pengucapan suku kata (silabel). Setiap sinyal secara otomatis akan dipisahkan ke dalam 20 silabel. Proses segmentasi dilakukan dengan cara mengekstraksi nilai energi jangka pendek (STE) dalam *frame-frame* kecil sepanjang 10 ms (440 sampel data). Setiap *frame* memiliki tingkat *overlapping* sebesar 75% (330 sampel data). Setiap *frame* kemudian di-*blocking* dengan menggunakan *Hamming Window*. Energi jangka pendek kemudian dinormalisasi sehingga sinyal berada pada rentang 0 hingga

Energi jangka pendek yang telah dinormalisasi akan digunakan sebagai acuan untuk menciptakan kontur energi yang lebih halus dengan logika *fuzzy*. Setiap 7 nilai energi secara berurutan hingga akhir akan menjadi masukan ke dalam fungsi keanggotaan logika *fuzzy*. Keluaran dari logika *fuzzy* adalah kontur STE yang lebih lembut. Pada hasil dari *fuzzy-smoothing* seringkali ditemukan puncak kecil di antara dua suku kata. Hal ini menyebabkan puncak tersebut dianggap sebagai *local maxima*. Untuk menghindari hal tersebut, setiap 11 data *slope* dari kontur diperlembut kembali dengan mengambil nilai *mean*. Pada akhir *fuzzy smoothing* nilai rata-rata dari Perbandingan energi jangka pendek sebelum dan sesudah *fuzzy-smoothing* ditunjukkan pada Gambar 4.



Gambar 4 Hasil Ekstraksi ciri jangka pendek (merah) dengan hasil *fuzzy-smoothing* (biru)

Batas pengucapan silabel kemudian didapatkan dengan mencari titik *local minima* dari kontur energi baru hasil pelembutan logika *fuzzy*. *Local minima* yang dicari memiliki ketentuan D1=16, D2=25, dan Th=1.5. Segmentasi kemudian dilakukan melalui pasca-pengolahan, dengan membagi sinyal suara hasil *pre-emphasis* sesuai jarak antara batas silabel. Keluaran dari segmentasi adalah sinyal mentah yang telah dipotong ke tingkat silabel.

Parameter-parameter yang digunakan dalam analisis performa segemntasi adalah akurasi segmentasi, *insertion*, dan *deletion* pada hasil segmen pada proses pra-pengolahan. Nilai akurasi segmentasi didapatkan dari persentase jumlah batas silabel dengan *error* di bawah 50 ms dari prediksi. Nilai *insertion* merupakan persentase jumlah batas silabel tambahan yang tidak terprediksi. Nilai *deletion* merupakan persentase jumlah batas silabel yang menghilang dari prediksi. Jumlah segmen yang dihasilkan adalah sebesar 2400 data yang masing-masing

mengandung sinyal pengucapan setiap huruf Jawa. Nilai akurasi segmentasi, *insertion*, dan *deletion* pada proses segmentasi otomatis ditunjukkan pada Tabel 2.

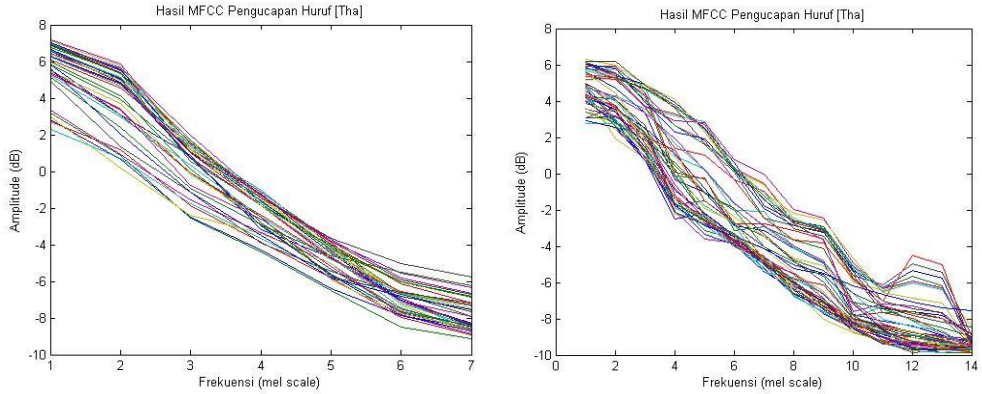
Tabel 2. Hasil Pengujian Segmentasi

Akurasi Segmentasi (%)	*Insertion* (%)	*Deletion* (%)
83,17	6,88	3,46

Nilai akurasi dari proses segmentasi otomatis terhadap pengucapan huruf Jawa ke tingkat silabel sebesar 83,17% dari seluruh data atau sebanyak 1996 data. Nilai *insertion* yang dihasilkan sebesar 6,88% dari seluruh data atau terdapat 165 batas silabel tambahan di luar prediksi yang ditemukan oleh segmentasi. Nilai *deletion* yang dihasilkan sebesar 3,46% dari seluruh data atau terdapat 83 batas silabel yang tidak terdeteksi.

3.3. Hasil Ekstraksi Ciri

Metode *Mel-Frequency Cepstral Coefficient* (MFCC) dipilih sebagai ekstraksi ciri yang digunakan. Matriks yang dihasilkan oleh MFCC akan memiliki dimensi yang berbeda. Dimensi matriks MFCC yang dihasilkan akan sebesar koefisien × jumlah sampel data. Contoh perbandingan nilai *cepstrum* hasil MFCC pada sinyal yang sama menggunakan 7 dan 14 dapat dilihat pada **Gambar 5.**



Gambar 5 Hasil MFCC dengan 7 koefisien (kiri) dan 14 koefisien (kanan)

Hasil dari MFCC belum dapat digunakan karena jumlah masukan JST-PB harus seragam. Penyeragaman dilakukan dengan menghitung nilai statistik, yaitu: nilai *Min, Max, Mean, Median, Sum,* dan Standar Deviasi dari matriks MFCC [11]. Hasil nilai statistik disimpan dalam vektor ciri sepanjang jumlah nilai statistik × jumlah koefisien MFCC agar dapat menjadi masukan ke JST-PB. Contoh hasil nilai statistik dapat dilihat pada Tabel 2.

Tabel 2 Contoh hasil penghitungan nilai statistik 7 koefisien MFCC

Koefisien MFCC	*Min*	*Max*	*Mean*	*Sum*	*Median*	Standar Deviasi
1	-13.468	-3.944	-8.081	-88.898	-8.163	2.331
2	13.354	18.985	17.071	187.872	17.359	1.507
3	3.091	5.578	4.564	50.213	4.673	0.830
4	-0.848	2.942	1.097	12.068	1.128	1.024
5	-0.410	2.302	0.938	10.326	0.922	0.733
6	-0.386	1.108	0.268	2.952	0.227	0.469
7	-2.818	0.517	-1.087	-11.961	-1.230	1.168

3.4. Hasil Jaringan Syaraf Tiruan

Pengujian jaringan syaraf tiruan perambatan balik (JST-PB) dibagi berdasarkan jumlah koefisien MFCC yang digunakan (7 atau 14 koefisien). Pengujian dilakukan dengan menggunakan 10-*Fold Cross Validation* dengan penyajian data hasil menggunakan Matriks *Confusion* untuk memperoleh nilai akurasi, presisi dan *recall.* Jumlah data yang digunakan adalah 2400 data dengan 90% data sebagai data latih dan 10% data sebagai data uji. Nilai akurasi, presisi, dan *recall* rata-rata dari seluruh JST-PB pada sesi 1 dan 2 ditunjukkan pada Tabel 3.

Tabel 3 Hasil Rata-rata Seluruh JST-PB

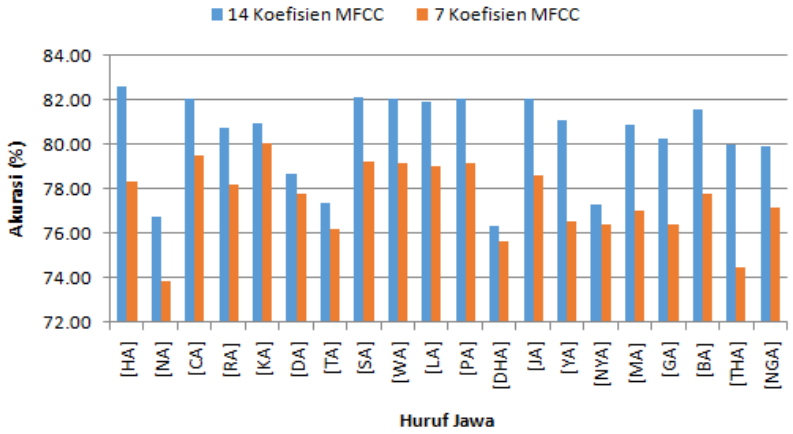
No	JST-PB	RATA-RATA (%)					
		7 Koefisien MFCC			14 Koefisien MFCC		
		AKURASI	PRESISI	RECALL	AKURASI	PRESISI	RECALL
1	[HA]	93.83	94.44	93.81	99.36	99.64	99.01
2	[NA]	91.48	91.82	93.46	95.91	96.18	99.93
3	[CA]	88.81	89.08	92.83	94.8	96.76	97.70
4	[RA]	88.32	90.44	92.70	95.39	96.82	98.34
5	[KA]	91.70	93.66	94.60	94.62	96.06	98.16
6	[DA]	92.29	93.72	95.24	94.64	96.06	98.18
7	[TA]	91.52	92.96	95.06	94.76	96.23	98.36
8	[SA]	92.54	93.96	96.08	94.07	96.19	97.60
9	[WA]	93.66	95.13	97.25	94.56	96.78	97.50
10	[LA]	91.12	93.24	94.65	97.36	98.62	99.67
11	[PA]	89.76	91.98	92.7	97.15	98.30	99.77
12	[DHA]	91.81	93.08	94.12	98.36	98.58	99.72
13	[JA]	90.85	92.00	93.47	98.76	98.88	99.74
14	[YA]	93.46	93.685	94.82	98.85	98.96	99.78
15	[NYA]	95.26	95.38	96.24	98.81	99.03	99.72
16	[MA]	95.36	95.65	96.27	98.73	98.95	99.67
17	[GA]	95.31	95.53	96.22	98.66	98.95	99.71
18	[BA]	95.23	95.45	96.17	98.68	98.94	99.78
19	[THA]	95.16	95.45	96.21	98.75	98.92	99.79
20	[NGA]	95.18	95.44	96.26	97.87	99.10	99.71

Hasil seluruh deteksi menunjukkan bahwa sistem dengan masukan 14 koefisien MFCC menghasilkan kinerja yang lebih baik. Peningkatan nilai akurasi dan *recall* tertinggi terjadi pada JST-PB [Pa] sebesar 7,395% dan 7,07%. JST-PB [Ca] mengalami peningkatan presisi tertinggi sebesar 7,66%. Peningkatan kinerja terendah terjadi pada JST-PB [Wa] dengan perubahan akurasi, presisi, dan *recall* masing-masing sebesar 0,803%, 1,655%, dan 0,245%.

3.5. Hasil Deteksi Kesalahan Pengucapan Keseluruhan

Hasil deteksi kesalahan pengucapan oleh komputer kemudian dibandingkan dengan hasil penilaian oleh ahli linguistik untuk mengetahui akurasi deteksi di setiap huruf Jawa dengan JST-PB 7 koefisien MFCC maupun 14 koefisien MFCC. Deteksi dengan 14 koefisien MFCC masih memberikan performa terbaik dengan nilai akurasi tertinggi sebesar 82,63% pada huruf [Ha]. Akurasi terbaik deteksi dengan 7 koefisien MFCC dihasilkan oleh huruf [Ka] sebesar 80,07%. Nilai akurasi terburuk dihasilkan pada huruf [Na] sebesar 73,86% untuk deteksi dengan 7 koefisien MFCC dan huruf [Dha] sebesar 76,32% untuk deteksi dengan 14 koefisien MFCC. Hasil segmentasi yang kurang baik mempengaruhi hasil deteksi secara keseluruhan sehingga menurun dibandingkan hasil JST-PB. *Insertion* dan *deletion* yang terjadi pada hasil segmentasi menyebabkan deteksi pada pengucapan huruf Jawa berikutnya secara berurutan menjadi tidak

sesuai dengan hasil penilaian oleh ahli lingustik. Diagram balok perbandingan akurasi deteksi kesalahan pengucapan huruf Jawa *carakan* dengan 7 dan 14 koefisien MFCC ditunjukkan pada Gambar 6.



Gambar 6. Perbandingan Akurasi Deteksi Kesalahan (7 dan 14 Koefisien MFCC)

3.6. Pembahasan

Hasil pengujian menunjukan bahwa deteksi kesalahan pengucapan huruf Jawa *carakan* dengan 14 koefisien MFCC menghasilkan kinerja lebih baik dibandingkan deteksi dengan 7 koefisien MFCC. Deteksi kesalahan pengucapan huruf Jawa dengan 14 koefisien MFCC menghasilkan akurasi tertinggi pada huruf [Ha] sebesar 82,63%, sementara deteksi kesalahan pengucapan huruf Jawa dengan 7 koefisien MFCC menghasilkan akurasi tertinggi sebesar 80,07% pada huruf [Ka]. Akan tetapi, nilai akurasi ini mengalami penurunan dibandingkan dengan performa JST-PB dengan 14 koefisien MFCC yang mampu menghasilkan akurasi tertinggi hingga 99,36% pada huruf [Ha]. Hal ini disebabkan akurasi segmentasi sebesar 83,12% mempengaruhi hasil deteksi kesalahan secara keseluruhan. Adanya *insertion* sebesar 6,88% dan *deletion* sebesar 3,46% pada segmentasi ikut serta dalam memperburuk akurasi deteksi kesalahan secara keseluruhan. Deteksi kesalahan pengucapan huruf Jawa pada JST-PB tidak mampu mengantisipasi pergeseran pengucapan yang terjadi akibat kesalahan segmentasi pada tahap sebelumnya.

Penggunaan ciri statistik dari 14 koefisien MFCC memberikan peningkatan pada keseluruhan kinerja karena penggunaan lebih banyak koefisien MFCC menghasilkan ciri yang lebih beragam pada koefisien yang merepresentasikan frekuensi rendah hingga menengah (Rana dan Miglani, 2014). Hal ini membuat lebih sensitif terhadap perubahan yang terjadi pada sinyal berfrekuensi rendah, di mana sinyal pengucapan banyak berada.

Deteksi kesalahan pengucapan huruf Jawa yang dibangun telah mampu mendeteksi *phoneme substitution* pada pengucapan dengan baik, tetapi kurang baik dalam mendeteksi *redundant phoneme* maupun *omitted phoneme* pada suatu sinyal pengucapan. Hal ini menyebabkan deteksi kesalahan pada beberapa huruf Jawa, yaitu: [Na], [Da], dan [Ta] yang mengalami *redundant phoneme*, dan [Tha], [Dha], dan [Nga] yang mengalami *omitted phoneme* memberikan kinerja yang rendah. Selain itu, deteksi hanya terbatas pada 20 fonem konsonan dari huruf Jawa *carakan*, sehingga tidak mempertimbangkan perubahan fonem vokal pada ucapan. Deteksi kesalahan ucapan dengan jumlah fonem kecil cenderung memperoleh akurasi lebih tinggi karena arsitektur yang lebih sederhana. Penelitian ini memiliki akurasi tinggi dikarenakan masih sangat sederhana dan terbatas ruang lingkup dari data yang digunakan. Pada penelitian-penelitian sebelumnya, deteksi kesalahan ucapan telah diimplementasikan pada kalimat efektif di suatu kalimat, sementara deteksi pada penelitian ini diterapkan pada data yang sederhana dan tidak efektif dalam komunikasi.

4. KESIMPULAN

Deteksi kesalahan pengucapan huruf Jawa *carakan* menggunakan sinyal ucapan dilakukan sampai pada penerapan secara *offline*.Segmentasi memberikan akurasi sebesar 83,17%, *insertion* sebesar 6,88%, dan *deletion* sebesar 3,46%. Deteksi kesalahan pengucapan huruf Jawa *carakan* dengan 7 koefisien MFCC menghasilkan akurasi tertinggi sebesar 80,07% pada huruf [Ka]. Deteksi kesalahan pengucapan huruf Jawa *carakan* dengan 14 koefisien MFCC menghasilkan akurasi tertinggi sebesar 82,63% pada huruf [Ha]. Deteksi kesalahan pengucapan huruf Jawa tidak mampu mengantisipasi pergeseran pengucapan huruf Jawa apabila terjadi *insertion* atau *addition* pada hasil segmentasi. Deteksi kesalahan pengucapan huruf Jawa *carakan* sulit mendeteksi *abundant phoneme* pada huruf [Na], [Ta], dan [Da] serta *omitted phoneme* pada huruf [Nga], [Tha], dan [Dha].

5. SARAN

Deteksi tidak memberikan hasil yang baik dalam mendeteksi *abundant* atau *omitted phoneme*, sehingga dibutuhkan seleksi ciri tambahan selain MFCC. Kemampuan subjek penutur perlu diuji terlebih dahulu dengan kriteria yang spesifik, selain wawancara. Deteksi kesalahan pengucapan huruf Jawa juga perlu dikembangkan agar dapat diterapkan secara *online*. Selain itu, jumlah fonem yang dapat diamati masih sangat terbatas, sementara terdapat banyak huruf Jawa yang belum digunakan. Deteksi juga perlu dikembangkan untuk mengantisipasi adanya *insertion* dan *deletion* pada hasil segmentasi. Terakhir, pengembangan sistem agar mendeteksi kesalahan pengucapan dalam suatu kalimat efektif dalam percakapan juga diperlukan.

DAFTAR PUSTAKA

[1] Tondo, F.H., 2009, Kepunahan Bahasa-bahasa Daerah, 11 (10), 277–296 [Online]. Available: http://jmb.lipi.go.id/index.php/jmb/article/view/245. [Accessed: 14-Okt-2018]

[2] G. Huang, J. Ye, Z. Sun, Y. Zhou, Y. Shen and R. Mo, English mispronunciation detection based on improved GOP methods for Chinese students, *2017 International Conference on Progress in Informatics and Computing (PIC)*, Nanjing, 2017, pp. 425-429.

[3] X. Li, J. Chen, M. Yao, D. Shen and F. Lin, English sentence pronunciation evaluation using rhythm and intonation, *The 2014 2nd International Conference on Systems and Informatics (ICSAI 2014)*, Shanghai, 2014, pp. 366-371.

[4] T. Dang and Kim-Giao Dang Thi, Automatic detection of common mispronunciations of Vietnamese speakers of English using SVMs, *2017 International Conference on System Science and Engineering (ICSSE)*, Ho Chi Minh City, 2017, pp. 231-234.s

[5] J. Yuan and M. Liberman, Automatic detection of "g-dropping" in American English using forced alignment, *2011 IEEE Workshop on Automatic Speech Recognition & Understanding*, Waikoloa, HI, 2011, pp. 490-493.

[6] L. Zhang, Research on English Pronunciation Recognition Based on Neural Network, *2018 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS)*, Xiamen, 2018, pp. 703-706.

[7] K. Yoshida, T. Nose and A. Ito, Analysis of English Pronunciation of Singing Voices Sung by Japanese Speakers, *2014 Tenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Kitakyushu, 2014, pp. 554-557.

[8] C. K. Dewa, Javanese vowels sound classification with convolutional neural network, *2016 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, Lombok, 2016, pp. 123-128.

[9] Suyanto & Putra, Agfianto. (2014). Automatic Segmentation of Indonesian Speech into Syllables using Fuzzy Smoothed Energy Contour with Local Normalization, Splitting, and Assimilation. *Journal of ICT Research and Applications*. 8. 97-112. 10.5614/itbj.ict.res.appl.2014.8.2.2.

[10] A. Charisma, M. R. Hidayat and Y. B. Zainal, Speaker recognition using mel-frequency cepstrum coefficients and sum square error, *2017 3rd International Conference on Wireless and Telematics (ICWT)*, Palembang, 2017, pp. 160-163.

[11] Sengupta, Nandini & Sahidullah, Md & Saha, Goutam. (2016). Lung sound classification using cepstral-based statistical features. *Computers in Biology and Medicine*. 75. 10.1016/j.compbiomed.2016.05.013.