

## INDIVIDUAL IDENTIFICATION SYSTEM DESIGN THROUGH VOICE USING LINEAR PREDICTIVE CODING METHOD AND K-NEAREST NEIGHBOR

Davita Nadia Fadhilah<sup>1</sup>, Rita Magdalena<sup>2</sup>, Sofia Sa'idah<sup>3</sup>

<sup>1,2,3</sup>Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom, Indonesia  
Email: : <sup>1</sup>[davitanf@student.telkomuniversity.ac.id](mailto:davitanf@student.telkomuniversity.ac.id), <sup>2</sup>[ritamagdalen@telkomuniversity.ac.id](mailto:ritamagdalen@telkomuniversity.ac.id),  
<sup>3</sup>[sofiasaidahsfi@telkomuniversity.ac.id](mailto:sofiasaidahsfi@telkomuniversity.ac.id)

(Naskah masuk: 18 Januari 2021, diterima untuk diterbitkan: 25 Januari 2021)

### Abstract

Humans have a variety of characteristics that are different from one another. Characteristics possessed by humans are genuine which can be used as a differentiator between one individual and another, one of which is sound. Voice recognition is called speech recognition. In this study, it was developed as an individual voice recognition system using a combination of the Linear Predictive Coding (LPC) method of feature extraction and K-Nearest Neighbor (K-NN) classification in the speech recognition process. Testing is done by testing changes in several parameters, namely the LPC order value, the number of frames, the K value, and different distance methods. The results of the parameter combination test showed a fairly good presentation of 73.56321839% with the combination parameter or LPC 8, the number of frames 480, the value of K 5, with the distance method used by Chebychev.

**Keywords:** K-Nearest Neighbor, Linear Predictive Coding, speech recognition

## SISTEM PENGENALAN INDIVIDU MELALUI SUARA DENGAN METODE LINEAR PREDICTIVE CODING DAN K-NEAREST NEIGHBOR

### Abstrak

Manusia memiliki berbagai macam ciri khas tersendiri yang berbeda satu sama lain. Ciri khas yang dimiliki manusia adalah karakteristik alami yang dapat dijadikan pembeda antara satu individu dengan yang lainnya, salah satunya adalah suara. Pengenalan suara disebut dengan *speech recognition*. Dalam penelitian ini, dibuat sistem pengenalan individu melalui suara dengan menggunakan kombinasi metode *Linear Predictive Coding* (LPC) sebagai ekstraksi ciri dan klasifikasi *K-Nearest Neighbor* (K-NN) dalam proses pengenalan suara. Pengujian dilakukan dengan menguji perubahan beberapa parameter yaitu nilai orde LPC, jumlah frame, nilai K, dan metode jarak yang berbeda-beda. Hasil pengujian kombinasi parameter terbaik menunjukkan presentasi akurasi yang cukup baik sebesar 73.56321839% dengan kombinasi parameter orde LPC 8, jumlah frame 480, nilai K 5, dengan metode *distance* yang digunakan *Chebychev*.

**Kata kunci:** K-Nearest Neighbor, Linear Predictive Coding, pengenalan suara

### 1. PENDAHULUAN

Setiap manusia diciptakan dengan berbagai macam ciri khas tersendiri yang dimiliki setiap individu. Ciri khas yang dimiliki setiap manusia ini menjadi dasar penting untuk dapat dikenali sebagai individu yang berbeda satu dengan yang lainnya. Ciri khas yang dimiliki manusia adalah karakteristik alami yang dapat dijadikan *person identification*, salah satunya adalah suara. Contoh lainnya adalah wajah, sidik jari, iris mata, dan tanda tangan[1]. Pengenalan individu sangat penting terutama pada era digital saat ini. Pengenalan individu baik melalui suara, sidik jari, wajah, dan lainnya dapat dikembangkan menjadi sebuah alat keamanan, absensi dan lain lain.

Pengenalan suara (*speech recognition*) disebut juga dengan *voice recognition*. *Voice recognition* adalah bagian dari ilmu komunikasi yang melibatkan pengolahan signal (*signal processing*). Sudah banyak dilakukan riset mengenai pengenalan ucapan, seperti robotika, sistem sekuriti dan lain lain[2]. *Voice recognition* merupakan pengenalan suara yang mengacu pada identitas pembicara yang memanfaatkan inputan sinyal suara yang masuk, mengambil informasi berupa ciri yang tersimpan dalam suara tersebut, mengolah suara, kemudian melakukan identifikasi terhadap suara masukan tersebut.

Dalam beberapa dekade ini sudah banyak penelitian yang memanfaatkan suara sebagai pengenalan individu. Sebuah penelitian dilakukan oleh Khoirul Anam menggunakan metode LPC dan *Fast Fourier Transform* (FFT) dengan hasil akurasi 70% dan waktu komputasi 5 detik[1]. Adapun penelitian yang dibuat menggunakan metode LPC dan Hidden Markov Model (HMM) memiliki nilai akurasi 53,34% dan waktu komputasi 51,27 detik[3]. Pada penelitian lain juga digunakan metode KNN dan MFCC untuk suara artikulasi P memiliki hasil 69%[4]. Sementara pada penelitian lain yang menggunakan metode MFCC dan KNN sebagai voice recognition mendapatkan hasil yang sangat baik yaitu 97,97%[5]. Adapun penelitian dengan menggunakan kombinasi LPC dan euclidean distance menghasilkan akurasi yang sangat baik yaitu 92%[6]. Penelitian-penelitian tersebut menjadi latar belakang penelitian ini untuk membuat sistem pengenalan individu melalui suara dengan menggunakan metode yang berbeda.

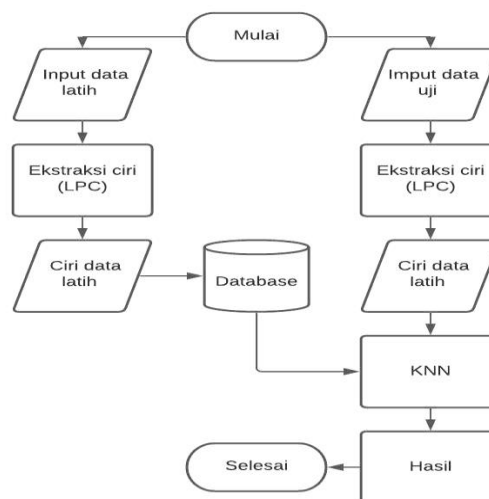
Untuk membuat perancangan sistem pengenalan individu melalui suara dibutuhkan beberapa metode untuk dapat memperoleh hasil yang baik. Pada penelitian ini dibuat sebuah sistem pengenalan individu melalui suara dengan metode *Linear Predictive Coding* untuk ekstraksi ciri dan metode *K-Nearest Neighbor* untuk pembelajaran pola sehingga suara dapat dikenali. Dilakukan juga analisis kombinasi parameter yang baik untuk sistem pengenalan individu melalui suara dengan metode LPC dan KNN ini.

Metode ekstraksi ciri LPC merupakan Teknik analisis suara yang kuat dan memberikan ekstraksi ciri yang memiliki kualitas yang baik dan hasil yang efisien untuk waktu komputasi[7]. Sementara KNN memiliki beberapa keunggulan yaitu menghasilkan pelatihan sangat cepat, merupakan metode klasifikasi yang sederhana dan mudah dipelajari, tahan terhadap data pelatihan yang memiliki derau, serta efektif jika dilakukan pada data yang banyak[8]. Maka dari itu pemilihan kombinasi metode LPC sebagai ekstraksi ciri dan KNN sebagai klasifikasi menjadi menarik untuk diteliti sebagai sistem pengenalan individu melalui suara.

## 2. METODE PENELITIAN

Penelitian ini terbagi menjadi dua tahap yaitu ekstraksi ciri dengan menggunakan metode LPC dan metode KNN untuk klasifikasi. Gambar 1 merupakan diagram sistem dalam penelitian ini.

Data ucapan yang telah diinputkan kemudian diekstraksi menggunakan metode ekstraksi ciri LPC sehingga dihasilkan ciri berupa nilai koefisien, selanjutnya proses pengenalan suara dilakukan dengan menggunakan nilai koefisien tersebut, metode K-NN berfungsi untuk mencari nilai yang terdekat dari nilai koefisien data uji dengan nilai koefisien data latih.



Gambar 1. Diagram Alir Sistem

### 2.1 Speech Recognition

*Speech recognition* merupakan teknologi yang memungkinkan suatu perangkat dapat mengenali dan memahami kata-kata yang diucapkan dengan cara digitalisasi kata dan mencocokkan sinyal digital tersebut dengan suatu pola tertentu yang tersimpan dalam suatu perangkat[2].

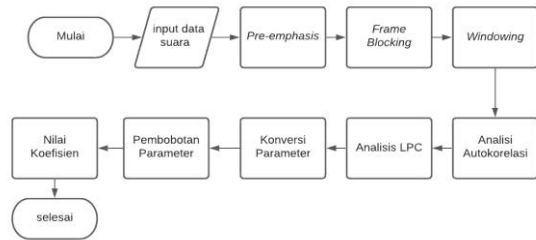
*Speech recognition* (pengenalan suara) merupakan proses mengidentifikasi ucapan seseorang untuk dapat mengenali orang tersebut. Ada empat dasar cara untuk melakukan *speech recognition* adalah sebagai berikut[9]:

- Template based approaches*, proses untuk mendapatkan hasil yang paling cocok dengan cara membandingkan input speech dibandingkan dengan database yang sudah ada.
- Knowledge based approaches*, untuk mengenali ucapan dengan proses pembelajaran pada sistem.
- Stochastic approaches*, melihat data statistika dari suatu individual *speech*.
- Connected approaches*, menghubungkan seluruh node yang di *training* untuk mengenali ucapan yang ingin diidentifikasi dengan menggunakan jaringan dari sejumlah contoh data.

Semua sistem identifikasi suara akan melalui dua proses penting yaitu *feature extraction* dan *feature matching*. *Feature extraction* akan melakukan proses ekstraksi data hasil akuisisi sehingga akan menghasilkan data yang berdimensi lebih kecil, kemudian nantinya digunakan untuk merepresentasikan tiap-tiap pembicara. *Feature matching* merupakan prosedur yang mengidentifikasi pembicara yang tidak dikenal kemudian akan dibandingkan fitur ekstraksi suara yang dimasukkan dengan salah satu dari database pembicara yang telah dikenal[10].

## 2.2 Linear Predictive Coding

Tahapan yang perlu dilakukan adalah ekstraksi ciri ucapan untuk mendapatkan nilai koefisien setiap data ucapan. Diketahui bahwa data awal berbentuk sebuah sinyal ucapan yang kemudian harus dikonversi menjadi bentuk angka untuk didapatkan ciri berupa nilai koefisien[11]. Tahapan ekstraksi ciri dapat dilihat pada Gambar 2.



Gambar 2. Proses Ekstraksi Ciri

Adapun langkah-langkah konversi tersebut yaitu

### 1. Preemphasis

Preemphasis Filter diperlukan setelah proses sampling dalam proses pengolahan sinyal ucapan. Proses ini bertujuan untuk mendapatkan bentuk spectral frekuensi ucapan yang lebih halus[12]. proses *preemphasis* ditunjukkan oleh persamaan (1).

$$\tilde{s}(n) = s(n) - \tilde{a}s(n - 1) \quad 0.9 \leq \tilde{a} \leq 1 \quad (1)$$

dimana  $S(n)$  adalah sinyal suara,  $\tilde{a}$  adalah koefisien preemphasis.

### 2. Frame Blocking

Proses ini dilakukan agar tidak ada sedikitpun sinyal yang hilang maka pada proses ini sinyal ucapan disegmentasi menjadi beberapa frame yang saling tumpang tindih.

### 3. Windowing

Dilakukan windowing setelah sinyal analog sudah diubah menjadi sinyal digital pada setiap frame, ini bertujuan untuk meminimalisir ketidakberlanjutan pada awal dan akhir setiap frame[13]. Windowing yang digunakan adalah *hamming window*, yang ditunjukkan oleh persamaan (2).

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

dimana  $N$  adalah jumlah *sample* tiap frame dan  $n$  adalah bilangan bulat dari 0 hingga  $N-1$ .

Akan didapat sinyal ucapan terbingkai yang telah dijendelakan menggunakan jendela *hamming window* menggunakan urutan seperti pada persamaan (3).

$$S_w(n) = \tilde{s}(n) \cdot w(n), \quad 0 \leq n \leq N - 1 \quad (3)$$

dimana  $N$  adalah sinyal yang diblok menjadi beberapa bagian dengan jumlah sampel  $N$ ,  $S_w(n)$

adalah nilai  $\tilde{s}(n)$  yang sudah melalui proses windowing, dan  $w(n)$  adalah *hamming window*.

### 4. Analisa Autokorelasi

Pada setiap frame hasil windowing dilakukan Analisa korelasi untuk mendapatkan nilai koefisien autokorelasi. Kegunaan dari proses analisa autokorelasi adalah untuk mengoreksi bentuk gelombang dengan dirinya sendiri[14]. Nilai auto korelasi dapat dicari dengan menggunakan persamaan (4) dan (5).

$$R_n(i - k) = \sum_{m=0}^{N-1-(i-k)} s_n(m)s_n(m + i - k) \quad (4)$$

$$\sum_{k=1}^p R = (|i - k|) \hat{\alpha} k = R_n(i), \quad 0 \leq i \leq p \quad (5)$$

dimana  $R_n$  adalah nilai auto korelasi,  $S_w(n)$  adalah sinyal yang telah melalui proses *windowing*,  $m$  adalah bilangan bulat dari 1 sampai  $p$ ,  $a_k$  adalah koefisien LPC,  $i$  adalah bilangan dari 1 sampai  $p$ ,  $R_n(i)$  adalah nilai auto korelasi ke  $i$ , dan  $p$  adalah orde dari analisa LPC yang akan dilakukan. Untuk orde analisa LPC nilai yang umum adalah antara 8 sampai 16.

### 5. Analisa LPC

Proses ini dilakukan untuk mendapatkan himpunan parameter LPC dengan cara mengkonversi setiap frame dari  $p+1$  autokorelasi menjadi himpunan parameter LPC, dengan menggunakan persamaan (6)

$$\begin{bmatrix} R_n(0) & R_n(1) & R_n(2) & \dots & R_n(p-1) \\ R_n(1) & R_n(0) & R_n(1) & \dots & R_n(p-2) \\ R_n(2) & R_n(1) & R_n(0) & \dots & R_n(p-3) \\ \dots & \dots & \dots & \dots & \dots \\ R_n(p-1) & R_n(p-1) & R_n(p-1) & \dots & R_n(0) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \dots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_n(1) \\ R_n(2) \\ R_n(3) \\ \dots \\ R_n(p) \end{bmatrix} \quad (6)$$

### 6. Konversi Parameter LPC ke Koefisien Cepstral

Proses ini dilakukan untuk mendapatkan hasil yang tahan terhadap *noise* dan mendapatkan kinerja yang lebih baik dengan menggunakan persamaan (7) dan (8).

$$c_m = \alpha_m + \sum_{k=1}^{m-1} \binom{k}{m} c_k a_{m-k}, \quad 0 \leq i \leq p \quad (7)$$

$$c_m = \sum_{k=1}^{m-1} \binom{k}{m} c_k a_{m-k}, \quad 0 \leq i \leq p \quad (8)$$

dimana  $c_m$  adalah koefisien *cepstral* dan  $\alpha_m$  adalah koefisien LPC.

## 2.3 K-Nearest Neighbor

Klasifikasi merupakan proses yang dilakukan untuk menemukan model atau fungsi yang dapat membedakan kelas data dengan tujuan agar model tersebut dapat digunakan untuk memprediksi kelas yang belum diketahui dari suatu objek pengamatan[15]. Klasifikasi *Nearest Neighbor* (K-NN) didasarkan pada pembelajaran (*learning*) dengan membandingkan sebuah data uji dengan sejumlah data latih[9]. Data pembelajaran akan diproyeksikan ke ruang berdimensi banyak, masing-masing dari dimensi tersebut merepresentasikan ciri

dari data. Ruang ini dibagi menjadi bagian-bagian berdasarkan klasifikasi data pembelajaran.

Teknik K-Nearest Neighbor dilakukan dengan beberapa langkah yaitu input data training, memberikan label pada data training, input data testing, mencari tetangga terdekat sejumlah k, memasukan data testing ke dalam kelas berdasarkan jarak terdekatnya[16]. Dekat atau jauhnya data biasanya dihitung berdasarkan jarak. Ada beberapa rumus untuk menghitung jarak terdekat antara data latih dan data uji.

1. *Euclidean Distance*, dengan rumus umum seperti pada persamaan (9)

$$d(x, y) = \sqrt{\sum_{i=1}^n |x_i - y_i|^2} \quad (9)$$

dimana  $d(x,y)$  adalah jarak antara  $x$  dan  $y$ ,  $x_i$  vektor bobot ke- $i$ , dan  $y_i$  adalah vector input ke- $i$ .

2. *CityBlock Distance*, dengan rumus seperti pada persamaan (10)

$$d(x, y) = \sum_{i=1}^n |x_i - y_i| \quad (10)$$

3. *Minkowski Distance*, dengan rumus seperti pada persamaan (11)

$$d(x, y) = (\sum_{i=1}^n |x_i - y_i|^r)^{\frac{1}{r}} \quad (11)$$

dimana  $r$  tidak boleh angka 1 dan 2, karena jika 1 akan menjadi rumus manhattan distance dan jika 2 akan menjadi rumus euclidean distance.

4. *Chebyshev Distance*, dengan rumus seperti pada persamaan (12)

$$d(x, y) = \max\{|x_i - y_i|\} \quad (12)$$

### 3. HASIL DAN PEMBAHASAN

Pada penelitian ini didapat 435 sample suara dari 29 orang speaker independent yang mengucapkan nama masing-masing sebanyak 15 kali. Data tersebut dibagi menjadi dua buah jenis data yaitu data latih dan data uji. Untuk data latih didapat sejumlah 348 sample suara, sedangkan untuk data uji terdapat 87 sample suara. Semua data rekaman tersebut disimpan dalam format file \*.wav.

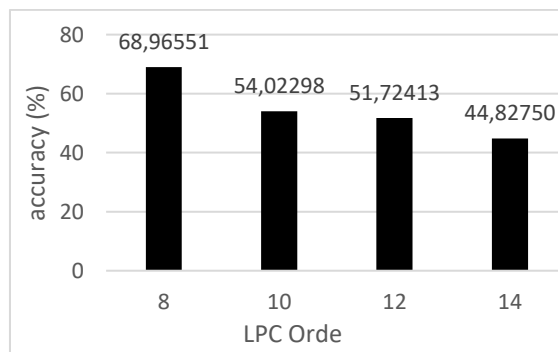
#### 3.1. Pengujian Nilai Orde LPC

Pada pengujian ini, sistem diuji dengan mengubah nilai parameter orde LPC dengan nilai 8, 10, 12, dan 14. Sementara untuk nilai parameter lain dibuat tetap. Jumlah frame bernilai 480, nilai K yang digunakan adalah 5, dan metode jarak yang digunakan adalah *Euclidean*. Setelah itu diambil nilai orde LPC yang memiliki nilai akurasi yang paling tinggi untuk digunakan pada pengujian selanjutnya. Tabel 1 merupakan hasil pengujian nilai orde LPC

Tabel 1. Hasil Pengujian Nilai Orde LPC

Orde LPC	Parameter yang Diuji			Hasil	
	Jumlah Frame	Nilai K	Distance	Jumlah Benar	Jumlah Salah
8				60	27
10	480	5	Euclidean	47	40
12				45	42
14				39	48

Dengan menguji nilai LPC yang diubah-ubah maka akan diketahui respon frekuensi yang akan dihasilkan secara lebih mendetail. Pemilihan orde LPC bergantung kepada frekuensi sampling yang digunakan. Hasil uji menunjukkan bahwa orde LPC 8 merupakan nilai orde yang baik untuk frekuensi sampling data yang digunakan sehingga jumlah data yang benar lebih banyak bila dibandingkan dengan penggunaan orde LPC 10, 12, dan 14. Pengaruh perubahan orde LPC yang digunakan dapat dilihat dari Gambar 3 yang menunjukkan akurasi dari sistem berdasarkan nilai orde yang diujikan.



Gambar 3. Hasil Akurasi Pengujian Orde LPC

Semakin besar orde LPC maka akan semakin banyak matriks koefisien LPC yang dihasilkan. Hasil dari pengujian menunjukkan bahwa orde LPC 8 menghasilkan matriks koefisien LPC yang dapat memberikan hasil pengenalan yang lebih baik dengan akurasi 68,965517%. Pada pengujian orde LPC dengan nilai yang lain hasil akurasi mengalami penurunan yang cukup jauh ketika dilakukan pengujian dengan orde 10 dan 12 yaitu sebesar 54,022989% dan 51,724138% dan kembali mengalami penurunan akurasi menjadi 44,827586% Ketika dilakukan pengujian pada orde LPC 14.

#### 3.2. Pengujian Jumlah Frame

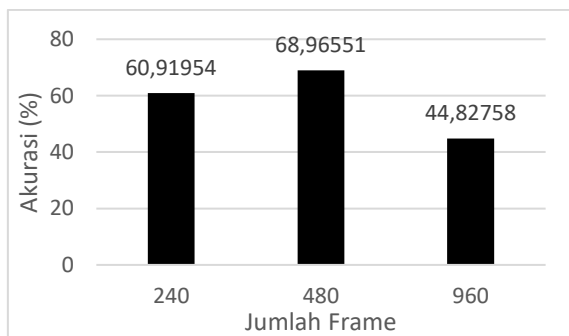
Pada pengujian kedua ini, sistem diuji dengan mengubah jumlah frame yang digunakan yaitu 240, 480, dan 960. Sementara untuk nilai parameter lain dibuat tetap. Nilai K yang digunakan adalah 5, metode jarak yang digunakan adalah *Euclidean*, dan nilai orde LPC ditentukan berdasarkan hasil pengujian sebelumnya yaitu 8. Setelah itu diambil nilai jumlah frame yang memiliki nilai akurasi yang paling tinggi untuk digunakan pada pengujian selanjutnya. Dapat dilihat pada Tabel 2 dari hasil pengujian jumlah frame.

Tabel 2. Hasil Pengujian Jumlah Frame

Orde LPC	Parameter yang Diuji			Hasil	
	Jumlah Frame	Nilai K	Distance	Jumlah Benar	Jumlah Salah
8	240	5	Euclidean	53	34
	480			60	27
	960			39	48

Nilai jumlah frame dapat mempengaruhi panjang frame yang didapat. Panjang frame ini berpengaruh terhadap pengambilan ciri sinyal suara yang diucapkan. Jumlah frame 480 memiliki hasil jumlah benar yang lebih banyak dikarenakan panjang frame yang didapat bisa menghasilkan ciri yang lebih optimal pada data suara yang diujikan.

Semakin banyak jumlah frame maka semakin kecil panjang frame yang dihasilkan. Dapat dilihat pada pengujian jumlah frame 960, jumlah benar yang dihasilkan tidak sebanyak jumlah benar yang dihasilkan ketika jumlah frame 480, hal ini menunjukkan panjang frame yang terlalu kecil dapat mempengaruhi hasil pengenalan. Pada Gambar 4 dapat dilihat pengaruh dari jumlah frame yang digunakan terhadap akurasi sistem. Akurasi sistem didapat dari perbandingan jumlah data yang benar terhadap 87 data yang diujikan.



Gambar 4 Hasil Akurasi Pengujian Jumlah Frame

Pada pengujian jumlah frame 240 dan orde LPC yang digunakan 8, nilai K 5, dan metode jarak K-NN yang digunakan adalah euclidean, sistem menunjukkan akurasi cukup baik yaitu sebesar 60,919%. Ketika jumlah frame diubah menjadi 480, dengan parameter lain dibuat sama, akurasi menunjukkan kenaikan yang cukup besar sekitar 8% menjadi 68,965% yang berarti jumlah frame 480 lebih baik dibanding jumlah frame 240. Dilakukan pula pengujian dengan jumlah frame 960 dengan parameter lain dibuat sama, hasil menunjukkan penurunan yang cukup jauh yaitu lebih dari 24% menjadi 44,827586% dimana hasil tersebut lebih kecil dibanding hasil pengujian pada jumlah frame 240. hal ini menunjukkan bahwa hasil terbaik didapat ketika menggunakan jumlah frame 480.

### 3.3 Pengujian Nilai K dan Metode Jarak K-NN

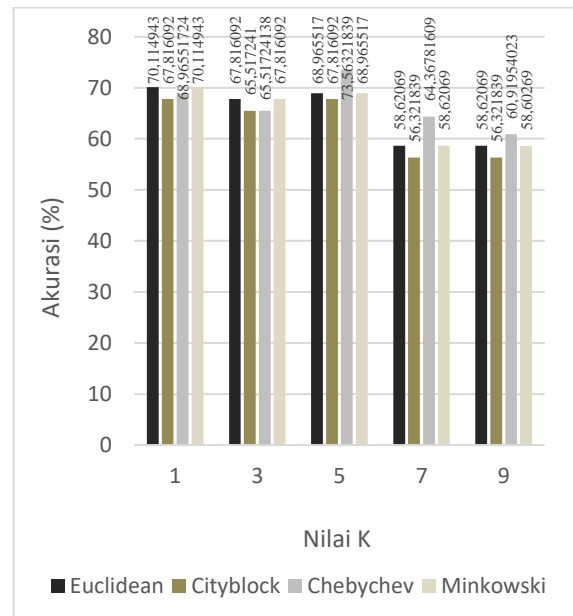
Pada pengujian ini, sistem diuji dengan mengubah nilai K dan metode jarak yang digunakan. Untuk nilai K yang diuji adalah 1, 3, 5, 7, dan 9.

Sementara metode jarak yang diuji adalah *Euclidean*, *CityBlock*, *Chebychev*, dan *Minkowski*. Sementara untuk nilai orde LPC ditentukan berdasarkan hasil pengujian pertama yaitu 8 dan jumlah frame ditentukan berdasarkan hasil pengujian kedua yaitu 480. Hasil dari pengujian ini dapat dilihat pada Tabel 3.

Tabel 3. Hasil Pengujian Nilai K dan Metode Jarak K-NN

Nilai K	Jumlah Benar			
	Euclidean	CityBlock	Chebychev	Minkowski
1	61	59	60	61
3	59	57	57	59
5	60	59	64	60
7	51	49	56	51
9	51	49	53	51

Hasil pengujian Nilai K dan metode jarak K-NN dibuat dalam bentuk diagram yang menunjukkan akurasi sistem pada pengujian ini dapat dilihat pada Gambar 5.



Gambar 5 Hasil Akurasi Pengujian Nilai K dan Metode Jarak

Dari pengujian skenario ketiga ini dapat dilihat bahwa perubahan metode jarak pada nilai K yang diuji mempengaruhi hasil akurasi sistem meskipun tidak berbeda cukup jauh. Sementara untuk pengujian pada Nilai K terdapat perbedaan yang cukup signifikan ketika nilai K 1, 3, maupun 5 hasil akurasi ditiap metode jarak cenderung tinggi, tetapi untuk nilai K 7 dan 9 terjadi penurunan akurasi yang cukup besar yaitu sekitar 10% pada metode jarak *Euclidean* dan *Minkowski*.

Secara umum, penambahan jumlah K pada metode jarak yang berbeda menghasilkan akurasi yang menurun. Meskipun dibebberapa pengujian penambahan nilai K mengalami peningkatan dan hasil terbaik diperoleh saat K bernilai 5 dengan metode *chebychev*. Hal ini menandakan penambahan jumlah K membuat misklasifikasi. Ketika nilai K

dinaikkan, maka ciri data uji tersebut justru lebih banyak dekat ke kelas lainnya. Dapat dilihat dari hasil akurasi pada setiap pengujian, ketika nilai K dinaikkan maka jumlah hasil yang salah mengalami peningkatan berarti ciri data latih tersebut tidak hanya dekat ke kelasnya sendiri, namun juga dekat oleh kelas lainnya.

#### 4. KESIMPULAN

Penelitian ini mengusulkan sistem pengenalan individu melalui suara dengan menggunakan metode LPC dan KNN. Sistem dapat mengidentifikasi sampel suara dari 29 *speaker independent* yang mengucapkan nama masing masing sebanyak 15 kali dengan akurasi 73.56321839% yang didapat dari kombinasi parameter orde LPC 8, jumlah frame 480, Nilai K 5, dan metode jarak K-NN adalah *Chebychev*. Hasil menunjukkan bahwa menentukan parameter yang tepat seperti nilai orde LPC, jumlah frame, nilai K, dan metode jarak yang digunakan untuk sistem yang dirancang dapat meningkatkan performa sistem pengenalan individu melalui suara,

#### DAFTAR PUSTAKA

- [1] K. Anam, "Pengenalan suara manusia menggunakan metode LPC," 2013.
- [2] "SPEECH RECOGNITION – MTI." [Online]. Available: <https://mti.binus.ac.id/2019/05/08/speech-recognition/>. [Accessed: 10-Feb-2020].
- [3] R. R. Juniansyah, "PERANCANGAN SISTEM PENGENALAN SUARA DENGAN METODE LINEAR PREDICTIVE CODING," *e-Proceeding Eng.*, vol. 4, no. 1, pp. 404–411, 2017.
- [4] A. Anggoro, S. Herdjunanto, and R. Hidayat, "MFCC dan KNN untuk Pengenalan Suara Artikulasi P," *Avitec*, vol. 2, no. 1, pp. 13–19, 2020, doi: 10.28989/avitec.v2i1.605.
- [5] Ranny, "Voice recognition using k nearest neighbor and double distance method," *ICIMSA 2016 - 2016 3rd Int. Conf. Ind. Eng. Manag. Sci. Appl.*, 2016, doi: 10.1109/ICIMSA.2016.7504045.
- [6] H. Saputra, "Sistem Pengenalan Kata dengan Menggunakan Linear Predictive Coding dan Nearest Neighbor Classifier," *J. Tek. Elektro Univ. Kristen Petra*, vol. 5, no. 1, pp. 19–24, 2005, doi: 10.9744/jte.5.1.
- [7] W. S. Mada Sanjaya, D. Anggraeni, and I. P. Santika, "Speech Recognition using Linear Predictive Coding (LPC) and Adaptive Neuro-Fuzzy (ANFIS) to Control 5 DoF Arm Robot," *J. Phys. Conf. Ser.*, vol. 1090, no. 1, 2018, doi: 10.1088/1742-6596/1090/1/012046.
- [8] A. Bode, "K-Nearest Neighbor Dengan Feature Selection Menggunakan Backward Elimination Untuk Prediksi Harga Komoditi Kopi Arabika," *Ilk. J. Ilm.*, vol. 9, no. 2, pp. 188–195, 2017, doi: 10.33096/ilkom.v9i2.139.188-195.
- [9] L. Rabiner and B. Juang, "Fundamentals of speech recognition," *Robust Automatic Speech Recognition*. pp. 9–40, 2016, doi: 10.1016/b978-0-12-802398-3.00002-7.
- [10] M. Yassir, "Penerapan Metode Algoritma Genetika Dalam Pengenalan Pola Sinyal Suara Manusia," *Inspiration*, vol. 2, pp. 28–35, 2012.
- [11] M. Walid and A. K. Darmawan, "Pengenalan Ucapan Menggunakan Metode Linear Predictive Coding (LPC) dan K-Nearest Neighbor (KNN)," *Energy, Univ. Panca Marga*, vol. 7, no. 1, pp. 13–22, 2017.
- [12] Thiang and H. Saputra, "Sistem Pengenalan Kata dengan Menggunakan Linear Predictive Coding dan Nearest Neighbor Classifier," 2005.
- [13] S. Rachman, "Visualisasi pengenalan ucapan vokal bahasa indonesia dengan metode lpc-dtw," 2011.
- [14] A. Y. P. Idwal, Y. I. Nurhasanah, and D. B. Utami, "Sistem Pengenalan Suara Bahasa Indonesia Untuk Mengenali Aksen Daerah," *J. Tek. Inform. dan Sist. Inf.*, vol. 3, no. 3, pp. 461–471, 2017, doi: 10.28932/jutisi.v3i3.661.
- [15] P. R. Sihombing and A. M. Arsani, "COMPARISON OF MACHINE LEARNING METHODS IN CLASSIFYING POVERTY IN INDONESIA IN 2018 PERBANDINGAN METODE MACHINE LEARNING DALAM KLASIFIKASI KEMISKINAN DI INDONESIA TAHUN 2018," *J. Tek. Inform.*, vol. 2, no. 1, pp. 51–56, 2021.
- [16] A. Rohman, "MODEL ALGORITMA K-NEAREST NEIGHBOR (K-NN) UNTUK PREDIKSI KELULUSAN MAHASISWA," *J. Ilm. Teknol.*, vol. 1, p. 236 hal, 2015.