

## OTOMASI MUSIC PLAYER BERBASIS SPEECH RECOGNITION PADA PLATFORM ANDROID

Nurullaili Fadhilah<sup>1</sup>, Bambang Hidayat<sup>2, 3</sup>

<sup>1</sup>Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

---

### Abstrak

Perkembangan teknologi komunikasi terkini yang diimplementasikan pada smartphone membuat jenis telepon genggam ini semakin menjadi perangkat mobile yang user friendly dan telah menjadikannya kebutuhan tersendiri bagi masyarakat. Salah satu aplikasi yang sering diakses adalah music player karena hampir setiap pengguna smartphone mendengarkan musik di sela aktivitasnya yang padat. Saat ini pengaturan music player masih secara manual yaitu dengan mengklik fungsi-fungsi tombol yang ada pada tampilannya. Cara ini dianggap masih belum efektif karena saat mengakses aplikasi ini dibutuhkan tingkat perhatian yang tinggi sehingga konsentrasi dalam melakukan aktivitas utamanya menjadi terganggu. Oleh karena itu dicari suatu cara yang dapat dengan mudah dan praktis melakukan pengaturan music player pada smartphone yaitu dengan mengimplementasikan Automatic Speech Recognition (ASR) yang menjadikan input suara sebagai perintah untuk menjalankan fungsi play, stop, next, previous, pause, (volume) up dan (volume) down.

Metode ekstraksi ciri yang digunakan dalam penelitian ini adalah Mel Frequency Cepstral Coefficient (MFCC). Metode MFCC mampu menangkap karakteristik suara yang penting bagi pengenalan sinyal suara. K-Nearest Neighbors (KNN) digunakan sebagai metode klasifikasi ciri. Metode KNN memiliki ketangguhan terhadap training data yang memiliki banyak noise. Bahasa pemrograman yang digunakan dalam penelitian ini adalah Java Android. Android merupakan salah satu basis platform perangkat mobile yang memiliki sifat open source sehingga penulis dapat mengembangkan aplikasi dan melakukan konfigurasi sistem sesuai kebutuhan penelitian.

Sistem yang telah dibuat memperoleh akurasi tertinggi sebesar 76% saat dioperasikan pada lingkungan tanpa noise. Apabila sistem diberikan noise, akurasi yang didapat mengalami penurunan yaitu 70% untuk level noise rendah, 67.14% untuk level noise sedang, dan 60% untuk level noise tinggi sehingga sistem dikatakan belum dapat bekerja sama baiknya dengan ketika sistem dioperasikan tanpa noise.

Kata Kunci : Speech Recognition, Music Player, Mel Frequency Cepstral Coefficient, K-Nearest Neighbors, Android

---

Telkom  
University

### **Abstract**

The latest communication technology embedded on a smart phone has made the mobile handheld more users friendly and become a necessity in majority of people. One of the applications that are frequently accessed is the music player because almost every smart phone users listen to music on the sidelines of their activities. The current settings for the music player still manually by clicking the functions button on the appearance. This way is still not effective because when accessing this application required a high level attention so users main activity being disturbed. Therefore sought a way that can be easily and practically doing the music player settings by implementing the Automatic Speech Recognition (ASR), which makes the voice input as a command to perform play, stop, next, previous, pause, (volume) up and down.

A feature extraction method used in this research was the Mel Frequency Cepstral Coefficient (MFCC). MFCC method capable of capturing important sound characteristic for speech recognition system. K-Nearest Neighbors (KNN) was used as a classification method. KNN having toughness to classify the training data that has a lot of noise. The programming language used in this research was Java Android. Android is one of platform for mobile devices that have open source capability so that author can develops applications and configure the system according to the needs of the research.

The best accuracy of this system is 76%.If noise added, the system"s accuracy goes down, which is 70% on low noise environment, 67.14% on medium noise environment, and 60% on high noise environment so the system is said not quite robust against noise.

**Keywords :** Speech Recognition, Music Player, Mel Frequency Cepstral Coefficient, K-Nearest Neighbors, Android

---

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Perkembangan teknologi komunikasi terkini yang diimplementasikan pada *smartphone* membuat jenis telepon genggam ini semakin menjadi perangkat *mobile* yang *user friendly* dan telah menjadikannya kebutuhan tersendiri bagi masyarakat. Salah satu aplikasi yang sering diakses adalah *music player* karena hampir setiap pengguna *smartphone* mendengarkan musik di sela aktivitasnya yang padat. Saat ini pengaturan *music player* masih secara manual yaitu dengan mengklik fungsi-fungsi tombol yang ada pada tampilannya. Cara ini dianggap masih belum efektif karena saat mengakses aplikasi ini dibutuhkan tingkat perhatian yang tinggi sehingga konsentrasi dalam melakukan aktivitas utamanya menjadi terganggu. Oleh karena itu dicari suatu cara yang dapat dengan mudah dan praktis melakukan pengaturan *music player* pada *smartphone* yaitu dengan mengimplementasikan *Automatic Speech Recognition* (ASR) yang menjadikan input suara sebagai perintah untuk menjalankan fungsi *play*, *stop*, *next*, *previous*, *pause*, *(volume) up* dan *(volume) down*.

Metode ekstraksi ciri yang digunakan dalam penelitian ini adalah *Mel Frequency Cepstral Coefficient* (MFCC). Metode MFCC mampu menangkap karakteristik suara yang penting bagi pengenalan sinyal suara<sup>[7]</sup>. *K-Nearest Neighbors* (KNN) digunakan sebagai metode klasifikasi ciri. Metode KNN memiliki ketangguhan terhadap *training data* yang memiliki banyak *noise*<sup>[13]</sup>. Bahasa pemrograman yang digunakan dalam penelitian ini adalah Java Android. Android merupakan salah satu basis platform perangkat *mobile* yang memiliki sifat *open source* sehingga penulis dapat mengembangkan aplikasi dan melakukan konfigurasi sistem sesuai kebutuhan penelitian.

### 1.2 Tujuan Penelitian

Tujuan dari tugas akhir ini adalah:

1. Membuat aplikasi berbasis *speech recognition* dengan metode ekstraksi MFCC dan klasifikasi KNN untuk otomasi pengaturan *music player* pada *smartphone* Android.

2. Menganalisis akurasi yang dihasilkan oleh sistem pengenalan suara yang diimplementasikan sebagai aplikasi *speech command* pada *smartphone* Android.

### 1.3 Rumusan Masalah

Permasalahan pada pembuatan Tugas Akhir ini adalah:

1. Bagaimana cara membuat suatu sistem pengenalan suara manusia untuk otomasi pengaturan *music player* dengan metode ekstraksi MFCC dan klasifikasi KNN pada *smartphone* Android?
2. Bagaimana cara membuat *database* hasil ekstraksi ciri perintah suara untuk disimpan pada perangkat yang digunakan?
3. Bagaimana cara mencari nilai *threshold* untuk dijadikan batasan sistem dalam mengklasifikasi perintah suara yang menjadi masukan?
4. Bagaimana performansi sistem *speech recognition* yang telah dibuat dalam mengenali dan menjalankan perintah suara yang terdeteksi?

### 1.4 Batasan Masalah

Untuk mempermudah dan membatasi cakupan pembahasan masalah pada Tugas Akhir ini, maka disimpulkan batasan-batasan sebagai berikut:

1. Aplikasi yang dibuat menangani fungsi “*play*”, “*stop*”, “*pause*”, “*next*”, “*previous*”, “*(volume) up*” dan “*(volume) down*”.
2. Data latihan merupakan data non-real time berupa rekaman \*.wav dengan frekuensi *sampling* 8000 Hz, 16 bit dan mono yang diambil menggunakan Adobe Audition CS5.5.
3. Metode ekstraksi ciri yang digunakan adalah *Mel Frequency Cepstral Coefficient* dan *k-Nearest Neighbors* sebagai metode klasifikasi ciri.
4. Metode perhitungan jarak terdekat saat klasifikasi adalah *Euclidean Distance* dan tidak membahas analisis pengaruh jenis metode pencarian jarak terhadap akurasi sistem.
5. Jenis *window* yang digunakan adalah *Hamming* dan tidak membahas analisis pengaruh jenis *window* terhadap akurasi sistem.
6. Aplikasi dibangun menggunakan bahasa pemrograman Java dengan bantuan Android *Software Development Kit* (SDK) pada Sony Xperia J-ST26i versi 4.1.2 (*Jelly Bean*).

7. *Integrated Development Environment* (IDE) yang digunakan adalah Eclipse 4.2 (Juno).
8. Aplikasi yang dibuat hanya dapat digunakan oleh *mobile phone* Android yang mendukung fitur *speech recognition* yaitu versi 2.2 (*Froyo*) ke atas.
9. *Database* yang disimpan pada perangkat berupa matriks ciri dari data latih dalam format \*.txt dengan menggunakan NetBeans IDE 7.3.

### 1.5 Metodologi Penelitian

Tugas Akhir ini dibagi menjadi beberapa tahap pengerjaan sebagai berikut:

a. Studi literatur

Pada tahap ini dilakukan pencarian referensi yang dapat mendukung pembuatan sistem dan aplikasi, serta berdiskusi dengan pihak-pihak yang berkompetensi

b. Proses Pengumpulan Data

Pada tahap ini dilakukan rekaman suara dengan format \*.wav untuk selanjutnya dilakukan proses ekstraksi ciri. Ciri yang didapat kemudian diubah menjadi teks untuk dijadikan *database* sistem pengenalan suara.

c. Perancangan dan Implementasi Sistem

Pada tahap ini dilakukan pembuatan aplikasi *mobile* dengan menerapkan sistem pengenalan suara berdasarkan perancangan yang telah dilakukan.

d. Pengujian Sistem dan Analisis

Pada tahap ini dilakukan pengujian dan analisis terhadap aplikasi yang telah dibuat.

### 1.6 Sistematika Penulisan

Sistematika penulisan laporan Tugas Akhir ini terdiri atas lima bab yang disusun sebagai berikut:

#### BAB I Pendahuluan

Berisi latar belakang, tujuan penelitian, rumusan masalah, batasan masalah, metodologi penelitian dan sistematika penulisan.

#### BAB II Dasar Teori

Berisi teori-teori yang mendukung dan mendasari penulisan laporan Tugas Akhir.

### **BAB III Perancangan dan Implementasi Sistem**

Berisi urutan proses perancangan dan implementasi sistem pengenalan suara untuk otomatisasi pengaturan *music player* pada perangkat bergerak berbasis Android dengan metode ekstraksi ciri *Mel Frequency Cepstral Coefficient* dan metode klasifikasi ciri *k-Nearest Neighbors*.

### **BAB IV Pengujian Sistem dan Analisis**

Berisi penjelasan tentang skenario pengujian sistem dan analisis terhadap hasil penelitian yang telah dilakukan.

### **BAB V Kesimpulan dan Saran**

Berisi kesimpulan dari analisa yang telah dilakukan dan saran untuk penelitian selanjutnya.



## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Dari hasil pengujian dan analisis sistem yang telah dilakukan pada sistem, dapat diambil beberapa kesimpulan sebagai berikut:

1. Pengujian nilai Euclidean *distance* sebagai nilai *threshold* sistem dalam mengklasifikasikan kelas perintah suara yang digunakan mendapatkan hasil 0.61956 sebagai batas bawah dan 2.96823 sebagai batas atas. Idealnya, batas bawah daripada nilai *threshold distance* pada setiap perintah suara lebih mendekati nol lebih baik, namun nilai yang didapat pada pengujian ini dapat terjadi akibat adanya *noise* yang ikut terekam saat akuisisi data secara *real time*. Pengujian tidak dilakukan pada kondisi ideal atau ruangan tanpa *noise* (kedap suara) karena melihat pembuatan program ini bertujuan untuk dapat digunakan pada kondisi *real*. Sehingga hasil dari pengujian ini yang kemudian digunakan sebagai *threshold distance* sistem pengenalan yang dibuat.
2. Pengujian sistem tanpa *noise* menghasilkan akurasi rata-rata sebesar 76%. Hal ini menunjukkan bahwa sistem yang telah dibuat dapat mengenali perintah suara dengan cukup baik.
3. Pengujian sistem dengan menggunakan *noise* menunjukkan adanya pengaruh *noise* terhadap akurasi dari sistem yang dibuat. Semakin besar *noise* yang ada, akurasi sistem semakin menurun yaitu dengan akurasi rata-rata 70% pada lingkungan dengan *noise* rendah, 67.14% pada lingkungan dengan *noise* sedang, 60% pada lingkungan dengan *noise* tinggi.

#### 5.2 Saran

Pengembangan lebih lanjut yang dapat dilakukan terhadap tugas akhir ini adalah sebagai berikut:

1. Penambahan fitur pada aplikasi *music player* seperti perintah suara *lyric*. Perintah suara ini mempermudah pengguna untuk melakukan pencarian tanpa harus membuka *browser* terlebih dahulu.
2. Perbaiki antarmuka sistem sehingga lebih menarik untuk digunakan.

3. Gunakan metode ekstraksi ciri lain seperti MFCC berbasis fonem. Dengan melakukan ekstraksi ciri untuk setiap *fonem*, ciri yang dihasilkan akan lebih unik sehingga performansi dari sistem diyakini akan meningkat.
4. Gunakan metode klasifikasi ciri lain seperti Hidden Markov Model. Saat ini telah ada *tools* dengan metode HMM yang membantu pengembang untuk membuat sistem pengenalan suara dengan metode ini secara *open source* dengan menggunakan basis *operating system* Linux yaitu *sphinx*.





## DAFTAR PUSTAKA

- [1] Akhmad, A. A. *Proses Pembentukan dan Karakteristik Sinyal Ucapan*. Bandung: Institut Teknologi Bandung.
- [2] Axelsson, A., & Bjorhall, E. (2003). *Real Time Speech Driven Face Animation*. Department of Electrical Engineering, Linkoping University.
- [3] Feng, L. *An Automatic Speaker Recognition System*. Laussane, Switzerland: Audio Visual Communication Laboratory, Swiss Federal Institute of Technology.
- [4] Hidayah, A. N. (2012). *Mengenal Sistem Kontrol atau Sistem Pengendalian*. Retrieved Februari 10, 2013, from <http://affinh.blogspot.com/2012/10/mengenal-sistem-kontrol-atau-sistem>
- [5] Irfansyah, M. (2011). *Pengukuran Kinerja K-Nearest Neighbors dan Self Organizing Maps Menggunakan Fast Fourier Transform untuk Identifikasi Penyakit Tanaman*. Bogor: Institut Pertanian Bogor.
- [6] Kinnunen, T. (2004). *Spectral Feature for Automatic Text Independent Speaker Recognition*. University of Joensuu Departement of Computer Science.
- [7] Manunggal, H. S. (2005). *Perancangan dan Pembuatan Perangkat Lunak Pengenalan Suara Pembicara dengan Menggunakan Analisa MFCC*. Surabaya: Universitas Kristen Petra.
- [8] Nazruddin, S. (2011). *Pemrograman Aplikasi Smartphone dan Tablet PC Berbasis Android*. Bandung: Penerbit Informatika.
- [9] NN. *Pengenalan Ucapan*. Retrieved Desember 3, 2012, from Wikipedia: [http://id.wikipedia.org/wiki/Pengenalan\\_ucapan#Proses\\_kerja\\_alat\\_pengenalan\\_ucapan](http://id.wikipedia.org/wiki/Pengenalan_ucapan#Proses_kerja_alat_pengenalan_ucapan)
- [10] NN. (2013). *Wikipedia Indonesia*. Retrieved Januari 10, 2013, from [id.wikipedia.org/wiki/KNN](http://id.wikipedia.org/wiki/KNN)
- [11] Putri, W. E. (2013). *Perancangan Aplikasi Capture Gambar dalam Posisi Diam dengan Metoda Frame Difference Berbasis Android*. Bandung: Institut Teknologi Telkom.
- [12] Rabiner, L., & Biing - Hwang, J. (1993). *Fundamental of Speech Recognition*. London: Prentice-Hall International Inc.
- [13] Rizal, A. (2011). Retrieved Januari 14, 2013, from <http://arl.blog.ittelkom.ac.id/blog/2011/07/k-nearest-neighbor-k-nn/>
- [14] Santoso, T. B., & Huda, M. (2008). *Modul Praktikum Pengolahan Informasi Wicara*.
- [15] Tychtl, Z. K., & Psutka, J. (1999). *Speech Production Based on the Mel Frequency Cepstrum Coefficients*. Retrieved Januari 7, 2013, from [http://www.isca-speech.org/archive/eurospeech\\_1999\\_2335](http://www.isca-speech.org/archive/eurospeech_1999_2335)