



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang Masalah

Aplikasi pengenalan suara (*speech recognizer*) adalah sebuah aplikasi yang memungkinkan komputer dapat mengenali kata-kata yang diucapkan dengan cara mendigitalisasi kata yang diucapkan dan mencocokkannya dengan pola yang sesuai (Business Dictionary, 2014).

Alat pengenalan suara ini sendiri sebenarnya sudah berkembang sejak dekade 1940an oleh perusahaan American Telephone and Telegraph Company (AT&T). Kemudian pada dekade 1960an, para peneliti dari AT&T berhasil membuat perangkat identifikasi kata-kata terpisah, kemudian pada tahun 1970an, mereka berhasil menciptakan perangkat pengidentifikasi kata-kata kontinu. Alat pengenalan ucapan ini menjadi sangat fungsional sejak tahun 1980an, dan masih dikembangkan hingga sekarang.

Seiring dengan perkembangan teknologi, aplikasi pengenalan suara ini kemudian berkembang lagi menjadi aplikasi yang lebih luas penggunaannya. Contohnya adalah aplikasi pengenalan ucapan yang mampu mendeteksi suara pembicara, dan aplikasi pengidentifikasi lagu (*song identifier / music identifier*) yang berguna untuk memberikan informasi judul lagu yang didengarkan oleh suatu perangkat.

Contoh aplikasi pengenalan lagu jenis ini adalah Shazam, SoundHound (atau sebelumnya disebut Midomi), Spotsearch, dan musiXmatch. Aplikasi tersebut

juga dapat dijalankan di berbagai macam *smartphone* (khususnya *platform* Android) dan terintegrasi dengan internet.

Munculnya alat pengenalan suara ini tidak lepas pula dari adanya algoritma pemroses sinyal seperti *Fast Fourier Transform* (FFT), *Fast Folding*, Algoritma Gerchberg-Saxton dan Algoritma Goertzel. Algoritma tersebut memiliki rumus penghitungan yang berbeda dalam mengolah sampel suara yang didapatkan.

Proses analisa ini dilakukan pada domain frekuensi, karena besaran yang paling menentukan suatu sinyal adalah frekuensi. Jadi, untuk mendapatkan nilai domain frekuensi dari suatu sampel, algoritma tersebut memerlukan beberapa rumus dan definisi yang beragam ketika diterapkan.

Selain pengolahan sinyal, aplikasi pengenalan lagu juga harus memiliki suatu metode untuk melakukan pencocokannya. Proses pencocokan tersebut bisa berupa penggunaan *hash function*, pencocokan *keypoint* dan metode-metode pencocokan lainnya.

Maka, adanya tren aplikasi pengenalan lagu tersebut menyebabkan penulis tertarik untuk mengetahui cara kerja alat identifikasi lagu tersebut. Penulis juga terdorong dengan usaha untuk mengetahui apa judul lagu dari lagu yang biasanya didengarkan oleh penulis dari tempat umum atau media komunikasi, namun tidak tahu judulnya.

Dalam penelitian ini, penulis akan menerapkan algoritma Goertzel yang biasa digunakan dalam aplikasi DTMF (*Dual Tone Multi Frequency*). DTMF ini biasa digunakan untuk pengolahan sinyal telekomunikasi antar telepon dengan frekuensi suara antara gagang telepon dan alat komunikasi lainnya.

Sedangkan pencocokan dalam penelitian ini dilakukan dengan mengimplementasikan MD5 *hash* - yang umumnya diterapkan dalam bidang kriptografi - dan *string matching*, sebuah proses yang cukup sederhana untuk diterapkan dalam pencocokan nilai suatu data, dimana data-data audio yang telah diolah dan dibuat *hash*-nya akan dicocokkan nilai atau karakternya.

Algoritma Goertzel ini tidak perlu memperhitungkan frekuensi spektrum secara keseluruhan, serta membutuhkan performa CPU yang tidak sebesar FFT (Kevin Banks, 2002). Maka itu, banyak pihak yang menganggap algoritma Goertzel ini lebih efisien dibandingkan dengan algoritma FFT yang sangat umum digunakan dalam aplikasi yang sejenis.

Maka, bermula dari fakta tersebut, penulis akan mencoba mengimplementasikan Algoritma Goertzel dan metode *string matching* ini ke dalam sebuah aplikasi pengidentifikasi lagu yang terdengar cukup familiar bagi beberapa orang.

## 1.2. Rumusan Masalah

Penelitian ini akan mengkaji masalah berikut.

- Bagaimana Algoritma Goertzel dapat diimplementasikan dan bekerja secara efektif, baik dalam segi kecepatan pemrosesan, dan kecocokan nada dalam proses pengenalan nada dan lirik lagu?
- Apakah metode pencocokan *string* dan fungsi *hash* kriptografik seperti MD5 cocok untuk diimplementasikan kedalam aplikasi pengenalan lagu?

### 1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk menerapkan dan mengetahui kinerja dari Algoritma Goertzel ketika diimplementasikan sebagai pengganti *fourier transform* dengan mencoba menerapkan algoritma tersebut dalam sebuah aplikasi pengenalan lagu. Selain itu, penelitian ini juga bertujuan untuk mengimplementasikan fungsi *hash* MD5 metode pencocokan *string* untuk mengetahui seberapa cocok metode tersebut ketika diimplementasikan pada aplikasi yang bersangkutan.

### 1.4. Manfaat Penelitian

Dengan adanya penelitian ini, maka penulis dapat mengetahui apakah Algoritma Goertzel, fungsi *hash* kriptografi dan metode pencocokan *string* cocok untuk diterapkan dalam aplikasi pengenalan lagu. Dengan demikian, para pengembang aplikasi pengenalan lagu atau suara dapat mengetahui apakah Algoritma Goertzel dan metode-metode pencocokan sederhana seperti pencocokan *string* cocok untuk diterapkan pada aplikasi yang akan dikembangkan.

Penelitian ini juga bermanfaat bagi kelompok pengembang aplikasi agar dapat mengetahui seberapa fungsionalkah metode-metode yang diterapkan dalam penelitian ini ketika diterapkan pada suatu aplikasi tertentu.

### 1.5. Batasan Penelitian

Aplikasi ini dapat memproses *input* dari lingkungan sekitar untuk dibandingkan dengan sampel lagu yang tersimpan dalam *database*. Namun jika

rekaman sampel *input* tersebut tidak terdengar jelas baik karena gangguan bunyi dari tempat lain ataupun kualitas sumber bunyi yang tidak baik, maka proses pengolahan sinyal akan mengalami gangguan sehingga ada kemungkinan hasil berupa judul lagu yang tidak atau kurang sesuai dengan sampel input. Maka, penelitian ini dibatasi pada kondisi dimana lagu disetel tanpa suara berisik yang berlebihan.

Aplikasi ini juga mengandalkan sistem *database* yang bisa menyimpan berbagai sampel lagu yang akan disesuaikan dengan sampel suara yang dimasukkan. Namun karena kapasitas memori untuk penyimpanan ke dalam *database* yang terbatas, maka tidak semua lagu bisa diketahui judulnya, melainkan hanya mengetahui lagu-lagu yang tersimpan dalam *database*.

Format data audio yang bisa dibaca dan diproses dalam penelitian ini hanya wav yang merupakan salah satu format audio paling umum digunakan dalam analisa sinyal digital. Audio yang diproses ini berupa lagu-lagu yang dinyanyikan oleh artis aslinya, tidak berlaku untuk versi *cover* atau tiruan yang dimainkan oleh orang lain.

Aplikasi ini juga mampu merekam sumber suara, namun tidak bisa secara langsung mengolah sampel rekaman setelah berhenti merekam. Untuk melakukan pencocokan, harus cari dahulu sampel audio yang tersimpan di *file* atau direktori tertentu dari suatu perangkat sebelum pencocokan dilakukan.

Aplikasi yang dibuat ini merupakan aplikasi yang berbasis *desktop* yang hanya dapat berjalan pada perangkat statis, seperti komputer *desktop* dan *laptop*, serta tidak bisa berjalan pada perangkat *mobile*. Agar tidak memberatkan kinerja

perangkat yang menjalankannya, maka dalam proses pencocokan, rekaman yang dijadikan sampel maksimal hanya berdurasi 10 detik.

## **1.6. Sistematika Penulisan**

Sistematika penulisan yang digunakan dalam penyajian laporan skripsi ini adalah:

### **BAB I      PENDAHULUAN**

Bab ini berisi latar belakang masalah, rumusan masalah, tujuan dan manfaat penelitian, batasan penelitian, metode penelitian, dan sistematika penulisan.

### **BAB II      LANDASAN TEORI**

Bab ini menjelaskan teori dan konsep dasar yang berhubungan dengan masalah yang dibahas dalam penelitian ini.

### **BAB III      METODE DAN PERANCANGAN APLIKASI**

Bab ini menjelaskan metode penelitian dan pembahasan dari aplikasi yang dibuat sehubungan dengan penelitian yang dilakukan

### **BAB IV      IMPLEMENTASI DAN HASIL PENELITIAN**

Bab ini berisi implementasi aplikasi dan pembahasan dari hasil penelitian yang dilakukan.

### **BAB V      SIMPULAN DAN SARAN**

Bab ini berisi kesimpulan yang diambil dari hasil penelitian yang dilakukan, dan saran untuk pengembangan penelitian lebih lanjut

