

**UJI KETEPATAN PENGENALAN SIDIK JARI DENGAN  
METODE LEVENSHTTEIN DISTANCE DAN HAMMING  
DISTANCE**

Skripsi



Oleh :

**GREGORIUS SAKTI GINANTAKA**

**71190463**

**PROGRAM STUDI INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA**

2023

**UJI KETEPATAN PENGENALAN SIDIK JARI DENGAN  
METODE LEVENSHTTEIN DISTANCE DAN HAMMING  
DISTANCE**

Skripsi



Diajukan kepada Program Studi Informatika Fakultas Teknologi Informasi  
Universitas Kristen Duta Wacana  
Sebagai Salah Satu Syarat dalam Memperoleh Gelar  
Sarjana Komputer

Disusun Oleh :

**GREGORIUS SAKTI GINANTAKA**

**71190463**

**PROGRAM STUDI INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS KRISTEN DUTA WACANA**

2023

## PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul :

### UJI KETEPATAN PENGENALAN SIDIK JARI DENGAN METODE LEVENSHTEIN DISTANCE DAN HAMMING DISTANCE

yang saya kerjakan untuk melengkapi sebagian persyaratan menjadi Sarjana Komputer pada pendidikan Sarjana Program Studi Informatika Fakultas Teknologi Informasi Universitas Kristen Duta Wacana, bukan merupakan tiruan atau duplikasi dari skripsi kesarjanaan di lingkungan Universitas Kristen Duta Wacana maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Jika dikemudian hari didapati bahwa hasil skripsi ini adalah hasil plagiasi atau tiruan dari skripsi lain, saya bersedia dikenai sanksi yakni pencabutan gelar kesarjanaan saya.

Yogyakarta, 26 April 2023



GREGORIUS SAKTI GINANTAKA

71190463

## HALAMAN PERSETUJUAN

Judul Skripsi : UJI KETEPATAN PENGENALAN SIDIK JARI  
DENGAN METODE LEVENSHTTEIN DISTANCE DAN  
HAMMING DISTANCE

Nama Mahasiswa : GREGORIUS SAKTI GINANTAKA

NIM : 71190463

Mata Kuliah : Skripsi (Tugas Akhir)

Kode : TI0366

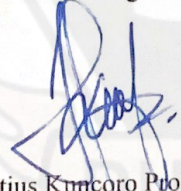
Semester : Genap


Tahun Akademik : 2022/2023

Telah diperiksa dan disetujui di  
Yogyakarta,  
Pada tanggal 26 April 2023

Dosen Pembimbing I

Dosen Pembimbing II

  
Laurentius Kuncoro Probo Saputra, S.T.,M.Eng.

  
Dr. Ir. Sri Suwarno, M.Eng.

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI  
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS  
SECARA ONLINE  
UNIVERSITAS KRISTEN DUTA WACANA YOGYAKARTA**

Saya yang bertanda tangan di bawah ini :

NIM : 71190463  
Nama : GREGORIUS SAKTI GINANTAKA  
Prodi / Fakultas : Teknologi Informasi / Informatika  
Judul Tugas Akhir : UJI KETEPATAN PENGENALAN SIDIK JARI  
DENGAN METODE LEVENSHTTEIN  
DISTANCE DAN HAMMING DISTANCE

bersedia menyerahkan Tugas Akhir kepada Universitas melalui Perpustakaan untuk keperluan akademis dan memberikan **Hak Bebas Royalti Non Eksklusif** (*Non-exclusive Royalty-free Right*) serta bersedia Tugas Akhirnya dipublikasikan secara online dan dapat diakses secara lengkap (*full access*).

Dengan Hak Bebas Royalti Non Eksklusif ini Perpustakaan Universitas Kristen Duta Wacana berhak menyimpan, mengalihkan / formatkan, mengelola dalam bentuk *database*, merawat dan mempublikasikan Tugas Akhir saya selama tetap mencantumkan nama saya sebagai penulis / pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenar-benarnya.

Yogyakarta, 26 April 2023

Yang menyatakan,

  
(71190463-GREGORIUS SAKTI GINANTAKA)

## HALAMAN PENGESAHAN

### UJI KETEPATAN PENGENALAN SIDIK JARI DENGAN METODE LEVENSHTEIN DISTANCE DAN HAMMING DISTANCE

Oleh : GREGORIUS SAKTI GINANTAKA / 71190463

Dipertahankan di depan Dewan Penguji Skripsi  
Program Studi Informatika Fakultas Teknologi Informasi  
Universitas Kristen Duta Wacana – Yogyakarta  
Dan dinyatakan diterima untuk memenuhi salah satu syarat memperoleh gelar  
Sarjana Komputer  
pada tanggal 29 Maret 2023

Yogyakarta, 26 April 2023

Mengesahkan,

Dewan Penguji :

1. Laurentius Kuncoro Probo Saputra, S.T.,M.Eng.


2. Dr.Ir. Sri Suwarno, M.Eng.

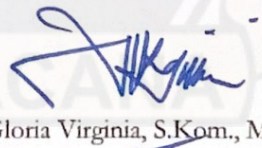
3. Yuan Lukito, S.Kom., M.Cs.

4. Lucia Dwi Krisnawati, Dr. Phil.

Dekan

Ketua Program Studi

  
(Resyandito, S.Kom., MSIS., Ph.D.)

  
(Gloria Virginia, S.Kom., MAI., Ph.D.)

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI  
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS  
SECARA ONLINE  
UNIVERSITAS KRISTEN DUTA WACANA YOGYAKARTA**

Saya yang bertanda tangan di bawah ini :

NIM : 71190463  
Nama : GREGORIUS SAKTI GINANTAKA  
Prodi / Fakultas : Teknologi Informasi / Informatika  
Judul Tugas Akhir : UJI KETEPATAN PENGENALAN SIDIK JARI  
DENGAN METODE LEVENSHTTEIN  
DISTANCE DAN HAMMING DISTANCE

bersedia menyerahkan Tugas Akhir kepada Universitas melalui Perpustakaan untuk keperluan akademis dan memberikan **Hak Bebas Royalti Non Eksklusif** (*Non-exclusive Royalty-free Right*) serta bersedia Tugas Akhirnya dipublikasikan secara online dan dapat diakses secara lengkap (*full access*).

Dengan Hak Bebas Royalti Non Eksklusif ini Perpustakaan Universitas Kristen Duta Wacana berhak menyimpan, mengalihkan / formatkan, mengelola dalam bentuk *database*, merawat dan mempublikasikan Tugas Akhir saya selama tetap mencantumkan nama saya sebagai penulis / pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenar-benarnya.

Yogyakarta, 26 April 2023

Yang menyatakan,

  
(71190463-GREGORIUS SAKTI GINANTAKA)

## KATA PENGANTAR

Segala puji dan syukur kepada Tuhan yang Maha Kasih, karena atas segala rahmat, bimbingan, dan bantuan-Nya maka akhirnya Skripsi dengan judul **UJI KETEPATAN PENGENALAN SIDIK JARI DENGAN METODE LEVENSHTAIN DISTANCE DAN HAMMING DISTANCE** ini telah selesai disusun.

Penulis memperoleh banyak bantuan dari kerja sama baik secara moral maupun spiritual dalam penulisan Skripsi ini, untuk itu tak lupa penulis ucapkan terimakasih yang sebesar-besarnya kepada :

1. Tuhan Yesus Kristus yang Maha Pengasih dan Penyayang.
2. Orang tua yang selama ini telah sabar membimbing dan mendoakan penulis tanpa kenal lelah untuk selama-lamanya.
3. Bapak Resyandito, S.Kom., MSIS., Ph.D. selaku Dekan Fakultas Teknologi Informasi yang membantu dalam kelancaran penulisan skripsi.
4. Ibu Gloria Virginia, S.Kom, MAI., Ph.D. selaku Kepala Program Studi Informatika yang membantu dalam kelancaran penulisan skripsi.
5. Bapak Laurentius Kuncoro Probo Saputra, S.T., M.Eng. selaku Dosen Pembimbing I yang telah memberikan ilmunya dan dengan penuh kesabaran membimbing dan membantu penulis.
6. Bapak Dr.Ir. Sri Suwarno, M.Eng. selaku Dosen Pembimbing II yang telah memberikan ilmunya dan dengan penuh kesabaran membimbing dan membantu penulis.
7. Keluarga tercinta yang senantiasa mendoakan kelancaran penulisan skripsi.
8. Papa dan Mama yang telah membantu dan mendukung segala kebutuhan baik materil dan non-materil dalam penyelesaian penulisan skripsi.
9. Serta seluruh pribadi yang terlibat dengan penulis dalam penyelesaian penulisan skripsi.



Laporan Skripsi ini tentunya tidak lepas dari segala kekurangan dan kelemahan untuk itu segala kritikan dan saran yang bersifat membangun guna kesempurnaan skripsi ini sangat diharapkan. Semoga Skripsi ini dapat bermanfaat bagi pembaca semua dan lebih khusus lagi bagi pengembangan ilmu komputer dan teknologi informasi.

Yogyakarta, 26 April 2023



Gregorius Sakti Ginantaka

71190463



## DAFTAR ISI

PERNYATAAN KEASLIAN SKRIPSI.....	ii
HALAMAN PERSETUJUAN.....	iii
HALAMAN PENGESAHAN.....	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	v
KATA PENGANTAR .....	vii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xi
DAFTAR GAMBAR .....	xii
INTISARI.....	xiii
ABSTRACT.....	xv
BAB I PENDAHULUAN .....	1
1.1. Latar Belakang Masalah.....	1
1.2. Perumusan Masalah.....	2
1.3. Batasan Masalah.....	2
1.4. Tujuan Penelitian.....	3
1.5. Manfaat Penelitian.....	3
1.6. Sistematika Penulisan.....	3
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI.....	5
2.1. Tinjauan Pustaka .....	5
2.2. Landasan Teori .....	7
2.2.1. Biometrik .....	7
2.2.1.1. Sidik Jari .....	7
2.2.2. Approximate String Matching .....	8
2.2.1.2. Levenshtein Distance .....	8
2.2.1.3. Hamming Distance.....	9
2.2.3. Rumus Nilai Kemiripan ( <i>Similarity</i> ) .....	10
2.2.4. Rumus Perhitungan Threshold .....	10
2.2.5. Rumus Perhitungan Presisi .....	10
2.2.6. Rumus Perhitungan Akurasi .....	11
BAB III METODOLOGI PENELITIAN.....	12

3.1.	Analisis Kebutuhan Sistem .....	12
3.1.1.	Kebutuhan Non Fungsional.....	12
3.1.2.	Kebutuhan Fungsional .....	12
3.2.	Perancangan Penelitian.....	13
3.3.	Blok Diagram Penelitian .....	14
<b>BAB IV IMPLEMENTASI .....</b>		<b>15</b>
4.1.	Implementasi Awal.....	15
4.2.	Implementasi Sistem .....	15
4.3.	Proses Perhitungan Metode Levenshtein Distance .....	24
4.3.1.	Perhitungan Threshold.....	28
4.3.2.	Analisis Perhitungan .....	28
4.4.	Proses Perhitungan Metode Hamming Distance .....	30
4.4.1.	Perhitungan Threshold.....	34
4.4.2.	Analisis Perhitungan .....	34
<b>BAB V KESIMPULAN DAN SARAN.....</b>		<b>37</b>
5.1.	Kesimpulan.....	37
5.2.	Saran .....	37
<b>DAFTAR PUSTAKA .....</b>		<b>39</b>
<b>LAMPIRAN .....</b>		<b>41</b>
LAMPIRAN I – CODING PROGRAM VB NET .....		41
LAMPIRAN II – KARTU KONSULTASI DOSEN PEMBIMBING I .....		65
LAMPIRAN III – KARTU KONSULTASI DOSEN PEMBIMBING II.....		66

## DAFTAR TABEL

Tabel 4. 1 Raw Dataset Karyawan.....	16
Tabel 4. 2 Dataset Hasil Preprocessing.....	18
Tabel 4. 3 Hasil Encode Dataset .....	19
Tabel 4. 4 Tabel Hasil Perhitungan Data Metode Levenshtein .....	27
Tabel 4. 5 Tabel Hasil Perhitungan Threshold Metode Levenshtein.....	28
Tabel 4. 6 Tabel Sebaran Perhitungan Data Uji Metode Levenshtein.....	29
Tabel 4. 7 Tabel Perhitungan Confusion Matrix Metode Levenshtein.....	30
Tabel 4. 9 Tabel Hasil Perhitungan Data Metode Hamming .....	33
Tabel 4. 10 Tabel Hasil Perhitungan Threshold Metode Hamming .....	34
Tabel 4. 11 Tabel Perhitungan Presisi Metode Hamming .....	35
Tabel 4. 12 Tabel Perhitungan Akurasi Metode Hamming .....	36



## DAFTAR GAMBAR

Gambar 3. 1 Blok Diagram Penelitian .....	14
Gambar 4. 1 Contoh Data Gambar .....	15
Gambar 4. 2 Gambar setelah di resize .....	17
Gambar 4. 3 Gambar setelah di compress.....	18
Gambar 4. 4 Tampilan Utama Program .....	22
Gambar 4. 5 Menu Input Data Karyawan .....	22
Gambar 4. 6 Contoh Inputan Data .....	23
Gambar 4. 7 Data Sidik Jari Pada Database.....	24
Gambar 4. 8 Data Sidik Jari Pada Database.....	24
Gambar 4. 9 Tampilan Menu Analisis Data Metode Levenshtein.....	25
Gambar 4. 10 Tampilan Hasil Perhitungan Data Dengan Metode Levenshtein ...	25
Gambar 4. 11 Tampilan Hasil Perhitungan Data Dengan Metode Levenshtein ...	26
Gambar 4. 12 Tampilan Hasil Perhitungan Data Dengan Metode Levenshtein ...	26
Gambar 4. 13 Tampilan Hasil Perhitungan Data Dengan Metode Levenshtein ...	27
Gambar 4. 14 Tampilan Menu Analisis Data Metode Hamming .....	31
Gambar 4. 15 Tampilan Hasil Perhitungan Data Dengan Metode Hamming.....	31
Gambar 4. 16 Tampilan Hasil Perhitungan Data Dengan Metode Hamming.....	32
Gambar 4. 17 Tampilan Hasil Perhitungan Data Dengan Metode Hamming.....	32
Gambar 4. 18 Tampilan Hasil Perhitungan Data Dengan Metode Hamming.....	33



## **INTISARI**

### **UJI KETEPATAN PENGENALAN SIDIK JARI DENGAN METODE LEVENSHTTEIN DISTANCE DAN HAMMING DISTANCE**

Oleh :

**GREGORIUS SAKTI GINANTAKA**

71190463

Keberadaan atau bukti hadir sangat penting dalam memantau kehadiran setiap orang yang bekerja di bidang tertentu. Pembangunan sistem presensi karyawan dengan menggunakan sidik jari atau fingerprint dapat mempercepat proses pengolahan data karyawan yang telah hadir atau belum hadir. Salah satu merek mesin yang digunakan sebagai alat presensi sidik jari adalah Fingerspot Flexcode. Data yang diperoleh dari mesin tersebut berupa gambar bitmap yang diubah menjadi string menggunakan encoding.

Meskipun deretan string yang dihasilkan berbeda, ada kemungkinan terdapat kesamaan data sidik jari antara karyawan karena sistem tidak dapat membedakan data dengan tepat. Oleh karena itu, perbandingan antara metode Levenshtein Distance dan Hamming Distance digunakan untuk mengetahui metode mana yang memiliki akurasi paling tinggi dalam pemrosesan perhitungan sistem. Metode dengan akurasi tertinggi akan menentukan tingkat kecocokan metode terhadap alat yang diuji. Sebagai contoh, 6 data sidik jari diambil dari masing-masing 7 karyawan yang berbeda sehingga terkumpul sebanyak 42 data sebagai data uji.

Hasil perhitungan menunjukkan bahwa akurasi metode Levenshtein Distance adalah 80,76 % dengan presisi sebesar 46,43 % dan metode Hamming

Distance adalah 78,34 % dengan presisi 30,50 % dalam memproses kemiripan string pada data sidik jari. Berdasarkan hasil tersebut, dapat disimpulkan bahwa metode Levenshtein Distance lebih baik dalam menghitung kemiripan pada data sidik jari dibandingkan dengan metode Hamming Distance karena mempunyai tingkat keakuratan dan juga presisi yang lebih tinggi dibandingkan dengan metode hamming distance.

**Kata-kata kunci :** Presensi, Sidik Jari, Levenshtein, Hamming, Akurasi.



## **ABSTRACT**

### **TESTING THE ACCURACY OF FINGERPRINT RECOGNITION USING LEVENSHTEIN DISTANCE AND HAMMING DISTANCE METHODS**

By :

GREGORIUS SAKTI GINANTAKA

71190463

The presence or evidence of attendance is crucial in monitoring the presence of every individual working in a particular field. Developing an employee attendance system using fingerprints can expedite the processing of data of employees who have or have not attended. One brand of machine used as a fingerprint attendance tool is Fingerspot Flexcode. The data obtained from the machine comes in the form of bitmap images that are converted into strings using encoding.

Although the resulting string sequences are different, there is a possibility of similarity in fingerprint data among employees because the system cannot distinguish data precisely. Therefore, the comparison between the Levenshtein Distance and Hamming Distance methods is used to determine which method has the highest accuracy in processing the system's calculation. The method with the highest accuracy will determine the level of compatibility of the method with the tested tool. For example, 6 fingerprint data are taken from each of the 7 different employees, resulting in a total of 42 data as test data.

The calculation results show that the accuracy of the Levenshtein Distance method is 80,76 % with a precision of 46,43 %, while the Hamming Distance method is 78,34 % with a precision of 30,50 % in processing string similarity in



fingerprint data. Based on these results, it can be concluded that the Levenshtein Distance method is better in calculating similarity in fingerprint data compared to the Hamming Distance method because it has a higher level of accuracy and precision compared to the Hamming Distance method.

**Keywords :** Attendance, Fingerprint, Levenshtein, Hamming, Accuracy.



# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang Masalah

Presensi atau bukti kehadiran merupakan hal yang penting untuk mengetahui dan memantau kehadiran dari setiap orang yang bekerja disuatu bidang tertentu. Salah satunya adalah jenis pekerjaan seperti karyawan perusahaan. Pentingnya presensi atau bukti kehadiran ini dikarenakan setiap perusahaan pasti akan memberikan gaji kepada karyawannya dan presensi ini juga digunakan sebagai pertimbangan dalam pemberian gaji karyawan. Seperti halnya jika karyawan lembur atau tidak hadir, hal tersebut tentu akan memberi pengaruh terhadap gaji yang diterima.

Bukti dari cepatnya perkembangan dalam bidang teknologi dan informasi yang ada pada zaman modern ini salah satunya adalah biometrik, sidik jari atau *fingerprint* yang merupakan salah satu identifikasi pribadi manusia dapat dimanfaatkan untuk mempermudah sistem dalam membedakan data tiap karyawan yang ada.

Pembangunan sistem presensi karyawan dengan menggunakan sidik jari atau *fingerprint* ini tentunya dapat mempercepat proses pengolahan data karyawan yang sudah tercatat hadir dan yang belum hadir. Pengembangan sistem ini tentu akan memudahkan perusahaan yang mempunyai karyawan dalam jumlah banyak untuk melakukan proses presensi kehadiran secara digital atau melalui sistem sehingga tidak menggunakan cara lama atau cara konvensional secara satu persatu.

Salah satu merk dari mesin yang digunakan sebagai alat presensi sidik jari atau *fingerprint* ini adalah *Fingerspot Flexcode*. Data yang didapatkan dari mesin tersebut adalah berupa gambar berformat *bitmap* yang kemudian akan diubah dengan *encoding* menjadi bentuk *string*. Hasil dari masing-masing deret *string* tersebut tentunya akan berbeda, tetapi tidak menutup kemungkinan bahwa akan ada kesamaan data sidik jari antar karyawan.

Hal itu terjadi karena sistem tidak dapat membedakan data secara tepat, oleh karena itu perbandingan penggunaan metode Levenshtein Distance dan Hamming Distance digunakan untuk mengetahui manakah dari kedua metode tersebut yang memiliki nilai akurasi paling tinggi dalam pemrosesan perhitungan yang ada pada sistem. Hal ini akan menghasilkan nilai akurasi tertinggi dan menentukan tingkat kecocokan metode terhadap alat yang akan diuji. Sebagai sampel dari pengujian ini penulis akan mengambil sebanyak 6 data sidik jari dari masing-masing 7 orang karyawan yang berbeda sehingga akan terkumpul sebanyak 42 data untuk dijadikan sebagai data uji.

### 1.2. Perumusan Masalah

Perumusan mengenai masalah yang dapat ditemukan dari latar belakang yang ada adalah sebagai berikut :

1. Seberapa besar tingkat akurasi sistem terhadap deteksi sidik jari atau *fingerprint* dari karyawan pada masing-masing metode ?
2. Metode manakah yang memiliki akurasi maksimal untuk diterapkan dalam algoritma presensi sidik jari karyawan ?

### 1.3. Batasan Masalah

Pembatasan masalah yang ada di penelitian yang dilakukan kali ini adalah sebagai berikut :

1. Alat presensi sidik jari atau *fingerprint* yang digunakan bermerk "*Fingerspot Flexcode*".
2. Data yang diolah untuk dilakukan pengujian merupakan data yang diambil dari alat *fingerprint* tersebut.
3. Format data yang diambil berupa gambar atau citra sidik jari dengan format bitmap yang kemudian diubah menjadi *string*.
4. Pengubahan data dari gambar atau citra sidik jari menjadi *string* menggunakan penyandian *code base64* sebagai acuan.
5. Metode pembandingan yang digunakan sebagai acuan dalam penelitian kali ini adalah metode *Levenshtein Distance* dan *Hamming Distance*.

#### **1.4. Tujuan Penelitian**

Tujuan utama penulis dalam mengadakan penelitian ini adalah untuk melakukan pengujian tingkat akurasi fitur presensi pada alat presensi sidik jari atau *fingerprint* dengan merk “*Fingerspot Flexcode*” dengan menggunakan dua metode untuk membandingkan manakah metode yang terbaik untuk diterapkan pada algoritma presensi sidik jari karyawan.

#### **1.5. Manfaat Penelitian**

Hasil penelitian yang telah didapatkan dari pengujian ketepatan fitur sidik jari atau *fingerspint* ini diharapkan dapat dipergunakan sebagai pembanding dan referensi dalam pengujian-pengujian sejenis yang menggunakan fitur dari alat yang berbeda serta menggunakan metode-metode yang berbeda juga.

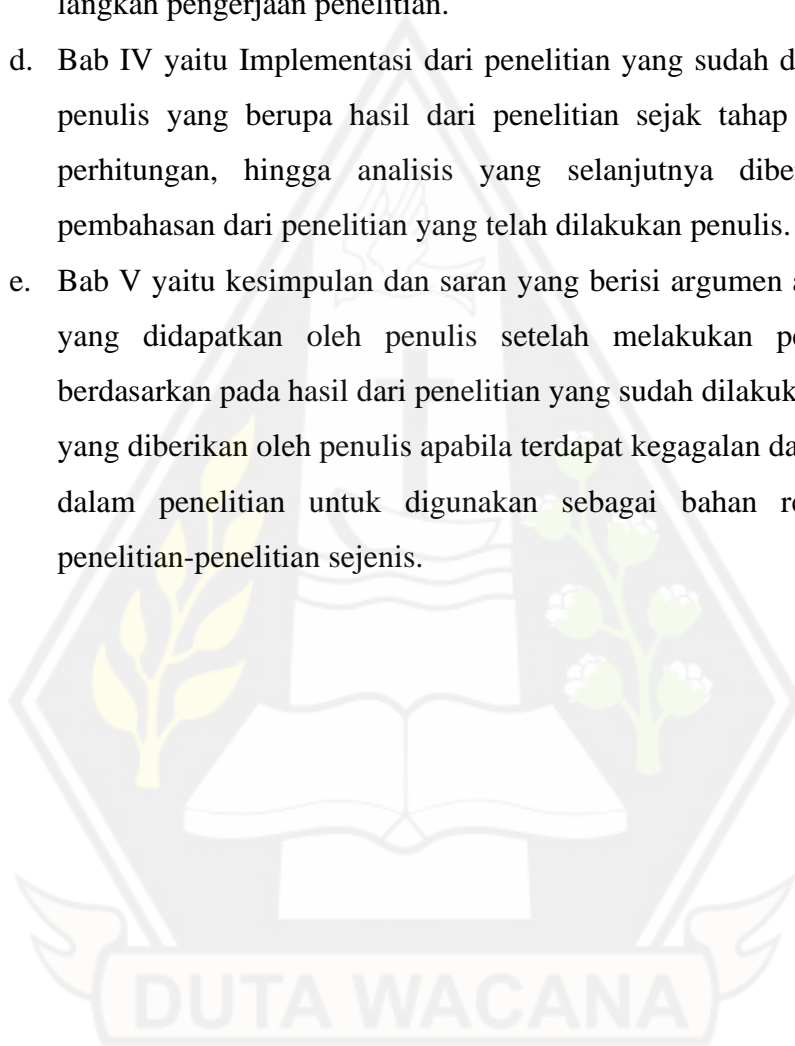
Apabila hasil penelitian yang berupa akurasi dinilai cukup baik, maka dapat diterapkan dalam sistem presensi karyawan di sebuah perusahaan dengan metode yang memiliki tingkat keakuratan yang tertinggi dari metode-metode yang ada.

#### **1.6. Sistematika Penulisan**

Sistematika yang digunakan oleh penulis pada penulisan penelitian mengenai Uji Ketepatan Pengenalan Sidik Jari Dengan Metode *Levenshtein Distance* dan *Hamming Distance* sebagai judul dari skripsi ini telah dirincikan sebagai berikut :

- a. Bab I yaitu pendahuluan yang berisi mengenai latar belakang masalah dari penelitian, rumusan masalah serta batasannya dari penelitian, tujuan dan manfaat penelitian, dan yang terakhir pernyataan keaslian penulisan penelitian.
- b. Bab II yaitu tinjauan pustaka dan dasar atau landasan teori yang berisi mengenai penelitian-penelitian yang terkait dengan judul dari penelitian yang dilakukan oleh penulis pada tinjauan pustaka serta dasar atau landasan teori yang berisi penjelasan teori dari berbagai macam istilah atau alat serta metode-metode yang digunakan oleh penulis pada penulisan penelitian skripsi ini.

- c. Bab III berisi metodologi penelitian yang merangkum langkah-langkah serta alur penelitian yang dilakukan oleh penulis, seperti analisa kebutuhan sistem, baik fungsional maupun non fungsional, perancangan penilitian mulai dari tahap studi pustaka, pengumpulan data, pengolahan data, hingga proses evaluasi, dan blok diagram sistem sebagai acuan langkah pengerjaan penelitian.
- d. Bab IV yaitu Implementasi dari penelitian yang sudah dilakukan oleh penulis yang berupa hasil dari penelitian sejak tahap awal, proses perhitungan, hingga analisis yang selanjutnya diberikan sedikit pembahasan dari penelitian yang telah dilakukan penulis.
- e. Bab V yaitu kesimpulan dan saran yang berisi argumen atau pendapat yang didapatkan oleh penulis setelah melakukan penelitian dan berdasarkan pada hasil dari penelitian yang sudah dilakukan serta saran yang diberikan oleh penulis apabila terdapat kegagalan dan kekurangan dalam penelitian untuk digunakan sebagai bahan referensi bagi penelitian-penelitian sejenis.



## **BAB II**

### **TINJAUAN PUSTAKA DAN DASAR TEORI**

#### **2.1. Tinjauan Pustaka**

Kumar, dkk. (2021) melakukan sebuah penelitian mengenai biometric fusion system yang dimana tujuan dilakukannya penelitian adalah untuk mengetahui apakah terjadi peningkatan akurasi fungsi dari biometric yang menggunakan pengenalan terhadap wajah dan juga sidik jari atau fingerprint. Dengan menggunakan teknik pendekatan Whale Optimized Algorithm (WOA) bersama dengan Maximally Stable Extremal Region (MSER) untuk mengekstraksi point fitur dari gambar mereka melakukan pendeteksian pola menggunakan SVM untuk mendapatkan tingkat akurasi yang diukur berdasarkan gambar yang digunakan sebagai data penelitian. Dari penelitian yang dilakukan didapatkan bahwa tingkat akurasi yang dihasilkan oleh biometric fusion system sebesar 91,6 % yang dimana tingkat presisinya sebesar 12,3 %.

Pengembangan sebuah sistem deteksi gender dan perhitungannya berdasarkan dari sidik jari atau fingerprint ini jarang dijumpai. Abdullah, dkk. (2016) melakukan penelitian dimana mereka membuat sebuah klasifikasi pada fingerprint atau sidik jari untuk mendeteksi gender menggunakan metode fingerprint global feature. Dari penelitian yang dilakukan oleh mereka, hasil yang didapatkan yaitu sebesar 74,5 % pendeteksian klasifikasi pada gender laki-laki dan perempuan berhasil dilakukan. Berdasarkan dari penelitian ini pula metode yang digunakan sangat direkomendasikan untuk digunakan kembali apabila akan melakukan penelitian serupa yang berupa klasifikasi.

Hicham, dkk. (2012) melakukan penelitian untuk pengenalan error yang mungkin saja terjadi dalam pengejaan dalam Bahasa Arab. Pengenalan error terhadap pengejaan Bahasa Arab ini berfokus pada inserting, deleting, dan permutation. Penelitian ini dilakukan karena terinspirasi dari algoritma levenshtein dimana digunakan untuk pendeteksian error yang serupa pada sebuah kata atau rangkaian huruf. Hasil yang diperoleh berdasarkan penelitian

yang dilakukan berupa penurunan tingkat error dari percobaan pertama yang sebesar 10 % menjadi 1,57 %.

Pratama dan Pamungkas (2016), membangun sebuah sistem yang mampu mendeteksi tingkat kemiripan antar dokumen teks menggunakan algoritma Levenshtein distance dengan menambahkan proses seperti *case folding*, *tokenization*, *stop word removal*, *stemming* dan *sorting*. Proses pencocokan string dari algoritma ini dapat menghasilkan nilai jarak yang menentukan persen kemiripan terbobot. Analisis stop word removal, derivasi, dan pengurutan yang digunakan dilakukan untuk melihat pengaruhnya terhadap kinerja algoritma Levenshtein distance. Simulasi algoritma ini dilakukan dengan dua dataset dan

satu dataset real. Hasil simulasi menunjukkan bahwa sorting berpengaruh besar terhadap algoritma Levenshtein distance. Hasil terbaik pada dataset 1 diperoleh untuk proses yang menggunakan stopwords deletion, forwarding dan sorting secara bersamaan. Hasil terbaik pada dataset 2 terlihat pada proses yang menggunakan stopwords dan wire yang dikombinasikan dengan sorting. Hasil terbaik dari data real diperoleh dalam proses stemming-sorting.

Pengujian algoritma yang dilakukan oleh Julian Tannga, dkk. (2017) dengan menggunakan aplikasi pendeteksi plagiarisme untuk menghitung skor kemiripan dan waktu proses yang dihasilkan oleh kedua algoritma tersebut. Tujuan dari penelitian ini adalah untuk mengukur perbandingan kinerja dari algoritma *Levenshtein Distance* dan algoritma *Jaro Winkler* untuk mendeteksi plagiarisme dalam dokumen teks. Data uji yang digunakan dalam penelitian ini terdiri dari dua data uji, yaitu data uji untuk mengukur kesamaan algoritma dan data uji untuk mengukur waktu pemrosesan algoritma. Setelah dilakukan pengujian, hasil dari kedua pengujian tersebut dirata-ratakan kemudian dianalisis perbandingannya. Hasil analisis perbandingan rata-rata kesamaan algoritma *Jaro-Winkler* adalah 80,92%, sedangkan nilai kesamaan rata-rata algoritma *Levenshtein-Distance* adalah 49,43%. Kemudian dilakukan analisis perbandingan waktu proses rata-rata algoritma *Jaro-Winkler* adalah 0,054 detik, sedangkan waktu proses rata-rata algoritma *Levenshtein-Distance* adalah

0,138 detik. Berdasarkan analisis komparatif, algoritme *Jaro Winkler* terbukti memiliki akurasi kesamaan yang lebih tinggi dan waktu pemrosesan yang lebih cepat daripada algoritme *Levenshtein Distance* dalam deteksi plagiarisme dokumen.

Berdasarkan penelitian yang dilakukan oleh Anggelina, dkk. (2022) didapatkan bahwa proses daur ulang teks yang telah didefinisikan sebagai penggunaan sumber tulisan yang ada merujuk untuk menulis teks baru. Oleh karena itu, diperlukan sistem deteksi kemiripan teks untuk mengenali teks mana yang tumpang tindih dan mana yang hampir tumpang tindih. Oleh karena itu, penelitian ini berfokus pada pendeteksian kesamaan teks menggunakan algoritma simhash. Algoritma ini membuat sidik jari dokumen yang bertindak sebagai fitur dokumen untuk membandingkan kesamaan tekstual dengan pengukuran jarak menggunakan metode hamming. Dalam pengujian yang berfokus pada dokumen duplikat dan hampir duplikat, tingkat pemulihan dokumen cukup tinggi yaitu sebesar 80%.

## **2.2. Landasan Teori**

### **2.2.1. Biometrik**

Sebagai acuan terbaru yang ditulis Thales (2020). *Biometric* merupakan sebuah alat yang berkembang saat ini dibidang teknologi dan informasi, yang mana kegunaan dari alat ini adalah sebagai alat untuk mengetahui ukuran dan juga melakukan perhitungan pada tubuh manusia yang setelah itu digunakan sebagai patokan dalam menunjukan dan menentukan karakteristik dari manusia itu. Dalam penggunaannya *biometric* membutuhkan fase verifikasi dan juga identifikasi.

#### **2.2.1.1. Sidik Jari**

Sidik jari berarti suatu pemeriksaan dengan suatu alat untuk mengetahui garis rekaman dari jari. Biasanya jari yang digunakan adalah ibu jari. Sidik jari dapat ditinggalkan pada suatu benda yang pernah tersentuh dengan telapak tangan atau pun telapak kaki menurut Iskandar, dkk. (2017).



Untuk mengetahui pemilik dari sidik jari bisa digunakan sebuah alat yaitu *fingerprint*. *Fingerprint* ini adalah suatu alat yang memiliki sensor yang bisa mengetahui sidik jari seseorang. Hal tersebut sering digunakan sebagai verifikasi identitas. *Fingerprint* sudah banyak digunakan di banyak tempat bahkan di alat elektronik seperti *smartphone*, pintu masuk, dan juga alat absensi karyawan, dsb.

## 2.2.2. Approximate String Matching

Menurut Kopelowitz dan Porat (2018). *Approximate String Matching* ialah sebuah algoritma proses untuk menemukan kemiripan atau persamaan yang ada pada dua buah *string* berbeda yang berdasarkan pada segi penulisan baik dalam susunan atau jumlah dari karakternya. Persentase kemiripan *string* ini dilihat dari jauh tidaknya perbedaan antara *string* satu dengan yang lainnya.

### 2.2.1.2. Levenshtein Distance

Menurut Azhri, dkk. (2019). *Levenshtein Distance* adalah metrik *string* yang digunakan melakukan pengukuran perbedaan antara dua buah barisan kata (*string*). Secara informal, *Levenshtein Distance* yang ada antara dua buah *string* merupakan jumlah minimum dalam melakukan perubahan sebuah karakter baik *insert*, *delete*, maupun *substitution* yang dibutuhkan dalam melakukan perubahan sebuah *string* menjadi *string* lain.

Berikut ini adalah *pseudo-code* dari algoritma *levenshtein distance* :

```
function levenshteinDistance(string1, string2)
    // Hitung panjang string
    len1 = length(string1)
    len2 = length(string2)
    // Buat matriks dengan ukuran len1 + 1 dan len2 + 1
    matrix = new array(len1+1, len2+1)
    // Inisialisasikan baris dan kolom pertama pada matriks
    for i from 0 to len1
        matrix[i][0] = i
```

```

for j from 0 to len2
    matrix[0][j] = j
// Isi matriks
for i from 1 to len1
    for j from 1 to len2
        // Hitung perubahan
        if string1[i-1] = string2[j-1]
            cost = 0
        else
            cost = 1
        // Kalkulasikan perubahan
        matrix[i][j] = min(matrix[i-1][j]+1, matrix[i][j-1]+1,
matrix[i-1][j-1]+cost)
        // Return nilai final
    return matrix[len1][len2]
end function

```

### 2.2.1.3. Hamming Distance

Menurut Anggelina, dkk. (2022). *Hamming Distance* merupakan sebuah metode yang digunakan untuk melakukan perhitungan pada dua buah *string* yang memiliki panjang sama, dimana setiap daripada *string* tersebut dikodekan terlebih dahulu dalam bentuk *biner* lalu dihitung nilai minimum substitusi yang dibutuhkan untuk mengubah sebuah *string* menjadi *string* yang lainnya.

Berikut ini adalah *pseudo-code* dari algoritma *hamming distance* :

```

function hammingDistance(string1, string2)
    // Hitung panjang string
    len1 = length(string1)
    len2 = length(string2)
    // Pastikan panjang string sama
    if len1 ≠ len2
        return "Error: strings must be the same length"

```

```

// Inisialisasikan jarak
distance = 0
// Kalkulasikan jarak hamming
for i from 0 to len1-1
    if string1[i] ≠ string2[i]
        distance = distance + 1
// Return jarak
return distance
end function

```

### 2.2.3. Rumus Nilai Kemiripan (*Similarity*)

$$Similarity = \left( 1 - \left( \frac{Distance}{MaxLenghtString} \right) \right) * 100 \% \quad [1]$$

*Similarity* = Nilai kemiripan *string*.  
*Distance* = Jarak perbedaan antar *string*.  
*MaxLenghtString* = Panjang Maksimal *string*.

### 2.2.4. Rumus Perhitungan Threshold

$$Threshold = \left( \frac{x1+x2+x3+x4+x5}{Total Similarity} \right) * 100 \% \quad [2]$$

Threshold = Nilai batas *similarity*.  
 X = Nilai *similarity*.  
 Total Similarity = Banyaknya nilai *similarity*.

### 2.2.5. Rumus Perhitungan Presisi

$$Precicion = \left( \frac{TP}{TP+FP} \right) * 100 \% \quad [3]$$

*Precicion* = Nilai presisi.  
 True Positif (TP) = Data benar diprediksi benar.  
 False Positif (FP) = Data salah diprediksi benar.

### 2.2.6. Rumus Perhitungan Akurasi

$$Accuracy = \left( \frac{TP + TN}{Total\ Dataset} \right) * 100 \% \quad [4]$$

*Accuracy* = Nilai Akurasi.

True Positif (TP) = Data benar diprediksi benar.

True Negatif (TN) = Data benar diprediksi salah.

Total Dataset = Total Jumlah Data Uji.



## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1. Analisis Kebutuhan Sistem**

Analisis kebutuhan sistem yang dilakukan oleh penulis pada penelitian kali ini berguna untuk mengetahui apa saja yang diperlukan dalam melakukan penelitian khususnya kebutuhan akan perangkat keras maupun perangkat lunak yang ada dan mendukung dalam berjalannya proses penelitian. Analisis kebutuhan sistem ini terbagi menjadi 2 (dua) yaitu kebutuhan non fungsional dan kebutuhan fungsional.

##### **3.1.1. Kebutuhan Non Fungsional**

Untuk melakukan proses penelitian ini penulis menggunakan bantuan perangkat keras yang sesuai dengan rincian dibawah :

1. Perangkat keras laptop merk MSI Seri GF63 11SC dengan spesifikasi :
  - a. Processor : Tiger Lake i5 11400H
  - b. RAM : DDR IV 8GB (3200 Mhz)
  - c. Storage : 512 GB NVMe PCIe Gen 3x4 SSD
  - d. Operating System : Windows 10 (64 bit)
2. Perangkat keras fingerprint merk Flexcode seri 4500 dengan spesifikasi :
  - a. Support Sistem : U.are.U 4500
  - b. Library : ActiveX DLL
  - c. Sample Code : VB6, Delphi 7, VB.Net, C#, PHP
  - d. Operating System : Windows XP/7/8/10 (32/64 bit)

##### **3.1.2. Kebutuhan Fungsional**

Berikut adalah rincian dari kebutuhan fungsional :

- a. Perangkat Lunak Aplikasi Microsoft Visual Studio.
- b. Perangkat Lunak Aplikasi Heidi Sql.
- c. Perangkat Lunak Aplikasi My SQL Workbench.
- d. Bahasa Pemrograman VB.Net.
- e. Bahasa Pemrograman SQL.

### 3.2. Perancangan Penelitian

Pada penelitian ini penulis melakukan beberapa penatalaksanaan yang sudah diurutkan berdasarkan pada kebutuhan dalam pelaksanaan penelitian seperti berikut :

#### 1 Studi Literatur

Pada tahap ini penulis melakukan pengumpulan pustaka atau informasi secara teoritis yang berkaitan dengan penelitian yang dilakukan oleh penulis. Pustaka atau informasi yang sudah terkumpul tersebut kemudian dikaji berdasarkan pada sumber-sumber yang valid seperti tesis, artikel, maupun jurnal yang resmi.

#### 2 Pengumpulan Data

Data sampel yang digunakan oleh penulis sebanyak jumlah populasi yang ada pada sebuah kantor di Yogyakarta, dimana karyawan yang berkerja di kantor tersebut sebanyak 7 orang dengan rincian pengambilan data sidik jari pada masing-masing karyawan adalah sebanyak 6 kali dengan pengambilan pertama digunakan sebagai acuan.

Berdasarkan pengambilan data tersebut didapatkan sebanyak 42 data sidik jari yang terkumpul dan akan digunakan sebagai sampel pada penelitian yang dilakukan oleh penulis. Data yang diambil oleh penulis merupakan data mentah atau *raw data* yang berupa gambar sidik jari dengan format bitmap.

#### 3 Pengolahan Data

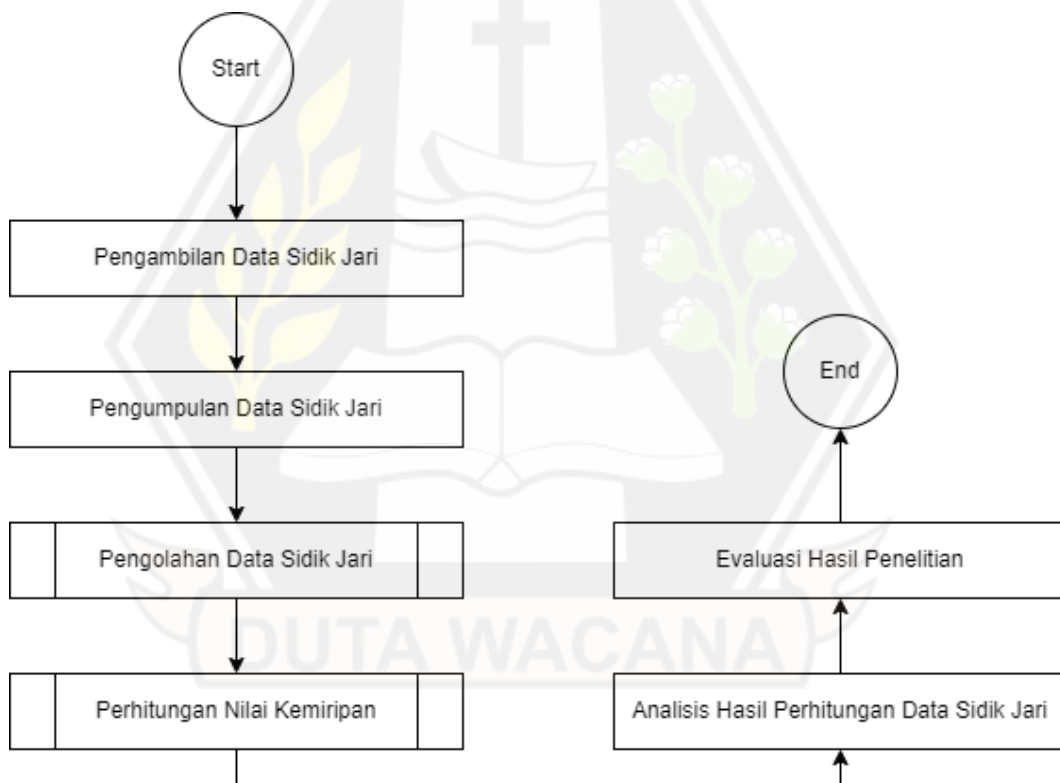
Setelah data selesai dikumpulkan maka dilakukan perhitungan berapa jarak perbedaan antar data sidik jari yang ada untuk kemudian setelah ditemukan jarak yang dihitung, dilakukan perhitungan kembali untuk menentukan nilai kemiripan pada setiap data yang telah dibandingkan dengan levenshtein distance dan juga hamming distance.

#### 4 Evaluasi

Pada evaluasi kali ini, dari setiap hasil yang telah didapatkan setelah proses perhitungan yang dilakukan maka hasil tersebut akan dievaluasi kembali untuk mengetahui dan menemukan apakah terdapat kesalahan dalam proses perhitungan dan pengolahan data.

Apabila ditemukan adanya kesalahan dalam melakukan proses pengolahan data dan perhitungan maka akan dilakukan perhitungan kembali dan apabila tidak ditemukan adanya kesalahan yang berarti pada proses perhitungan maka data yang diperoleh dapat dinyatakan sebagai data yang valid atas dasar perhitungan yang dilakukan.

### 3.3. Blok Diagram Penelitian



Gambar 3. 1 Blok Diagram Penelitian

## **BAB IV**

### **IMPLEMENTASI**

#### **4.1. Implementasi Awal**

Pada penelitian ini, yang dilakukan oleh penulis di tahap implementasi awal adalah pengambilan serta pengumpulan data untuk penelitian. Data yang diambil merupakan data gambar yang memiliki format bitmap. Penggunaan format bitmap dengan alasan bahwa format bitmap tersebut mempunyai keunggulan apabila gambar yang disimpan mempunyai ukuran yang sama, maka setelah dilakukan proses encoding hasil string yang didapatkan akan mempunyai jumlah karakter yang sama dan tidak akan berbeda panjang atau jumlah stringnya.

#### **4.2. Implementasi Sistem**

Setelah data yang digunakan untuk penelitian terkumpul, maka langkah selanjutnya yang dilakukan oleh penulis adalah sebagai berikut :

a. Pengambilan dan Pengumpulan Dataset

Kegiatan penelitian dimulai dengan melakukan pengumpulan data sidik jari pada sebuah kantor di Yogyakarta dimana jumlah populasi karyawan berjumlah 7 orang dengan rincian pengambilan data sidik jari sebanyak 6x untuk setiap orangnya.




























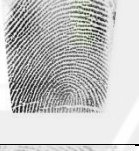
















Gambar 4. 1 Contoh Data Gambar



Dataset yang didapat berupa gambar berformat bitmap seperti pada Gambar 4. 1. Rincian data yang sudah diambil tertera pada rincian Tabel 4. 1.

Tabel 4. 1 Raw Dataset Karyawan

No	Kode	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6
1.	Karyawan 1						
2.	Karyawan 2						
3.	Karyawan 3						
4.	Karyawan 4						
5.	Karyawan 5						
6.	Karyawan 6						
7.	Karyawan 7						

b. Pengubahan Ukuran Dataset (*Resize*)

Pada saat pengambilan dan pengumpulan data didapatkan bahwa data gambar yang terkumpul mempunyai ukuran sebesar 411 KB yang

dimana ukuran tersebut terlalu besar untuk diolah ke dalam bentuk *string*. Oleh karena itu penulis melakukan proses pengubahan ukuran data atau *resize* data sebesar 50 %, yang dimana keputusan melakukan pengubahan gambar sebanyak 50 % dilakukan untuk menghindari rusaknya data.

Proses pengubahan data menggunakan bantuan website *resize* dengan domain *aspose.app* serta menggunakan ketentuan bahwa data yang diubah ukurannya tetap berformat bitmap. Data yang telah diubah ukurannya atau *resize* melalui website tersebut menghasilkan data bitmap berukuran 137 KB seperti pada Gambar 4. 2.



Gambar 4. 2 Gambar setelah di *resize*

c. Pemampatan Dataset (*Compress*)

Setelah dilakukan proses pengubahan ukuran data, tahapan selanjutnya yang dilakukan penulis dalam pengolahan data adalah pemampatan data atau *compress* data menggunakan bantuan *website* dengan domain yang sama yaitu *aspose.app* yang dimana ketentuan dalam pemampatan data adalah sebagai berikut :

1. Menggunakan *compression type* RGB.
2. Menggunakan palette bits sebesar 1.

Hasil yang didapatkan setelah proses pemampatan data dilakukan berupa data gambar dengan format yang sama yaitu bitmap dan ukuran file yang lebih kecil sebesar 5 kb seperti pada Gambar 4. 3.



Gambar 4. 3 Gambar setelah di compress

Berikut adalah rincian hasil dari dataset yang telah diolah sebagaimana tampil pada Tabel 4. 2.

Tabel 4. 2 Dataset Hasil Preprocessing

No	Kode	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6
1.	Karyawan 1						
2.	Karyawan 2						
3.	Karyawan 3						
4.	Karyawan 4						
5.	Karyawan 5						

No	Kode	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6
6.	Karyawan 6						
7.	Karyawan 7						

d. Pengubahan Dataset Menjadi String (*Encode*)

Proses pengubahan data gambar menjadi data *string* melalui proses *encoding*. Pada tahap ini dipilihnya proses *encoding* sebagai jembatan untuk mengubah data yang berupa gambar menjadi data berbentuk *string* untuk memudahkan proses perhitungan menggunakan metode *levenshtein* dan juga metode *hamming*.

Dalam proses pengubahannya penulis menggunakan bantuan website dengan domain *codebeautify.org*. Salah satu contoh dari data yang telah diubah ditampilkan pada Tabel 4. 3.

Tabel 4. 3 Hasil Encode Dataset

Data Gambar	Data String
	Qk2eEgAAAAAAD4AAAAoAAAAsgAAAMQAAAABAAEAAAAAGASAADEDgAAxA4AAAA AAAAAAAAAPz8//7+/v//////////wAD//////////wAD//////////wAD/// //////////wAD//////////wAD//////////wAD//////////wAD//////// //////////wAD//////////wAD//////////wAD//////////wAD//////// //////////wAD//////////wAD//////////wAD//////////wAD//////// //////////wAD//////////wAD//////////wAD//////////wAD//////// //////////wAD//////////wAD//////////wAD//////////wAD//////// ////////wAD///xz/n////////9/9/z////////wAD///xxxjv/3////////9/8973////////wAD///5xhzn5z////////9897zP////////wAD///5 xxz5x13///++/c85jP////////wAD//95xxz551z/3////uY5jf////////wAD//8wzzx/57z/3///v+Y5j////////wAD//8wxxw 485//3//nvOcx////////wAD//45xxw899/7////nnOc////////wAD//4www48c977////////t////////wAD//84ww4ccd/ ////////N////////wAD//+484OP//+9////////2P////////wAD//9/8/+P//+++////////85mf////////wAD//9//8P//++ef//8f3cxnd ////////wAD//8cf8D//+fP//+f3czn9////////wAD//+ePcF//fvv/+P93c7vf////////wAD//+PHOM5////3v//Pd3czO////

Data Gambar	Data String
	//wAD//+dHHOY13//7vvnno+ZzM7/////wAD//7pnHGM4x/////m+Z3+yf/////wAD//5onPHOc53/////nf53 dy/////wAD//5sjnnOc5xn/////ne7m5z/////wAD//53xhjHee5/P////vfzO53/////wAD//4jzxjnOeY7jG+//fze5v/ /////wAD//8jzxxjmec9zjm3f//nd7/////wAD//3hxw4znOf97jjnv//mZv/////wAD//3xx44zjGP87x 33/8/u/uff/////wAD//zw5449znHef57//d7/nc8f/////wAD//z888cdxznPfe9//z/uc4f/////wAD//58cO cc5xjnvOe//nfc5zf/////wAD//4+OOMO84znn/Pv//uf53/////wAD//88OHGHc45z33nnv88/zne/ /////wAD//+ePHHHOc953zxz4R9/nuc/////wAD//+eDD/nmce57755//93Oc5/////wAD//vPn//vO Of//+ef/nu+Z/P/////wAD//Hnz/znvOPffP/v/H857nP/////wAD//n3x455z/vHP/vv5+/Z5zPf/////w AD//jz48c85z/jj//3//87jnP/////wAD//z548Me95zxz95+//5/PP9/////wAD//8YcHeHc4544++fj+P 8ef/////wAD//8ecjvTuc8d8e/v8f/x8/z/////wAD//+fsjnZnOcf+f/3/j+Hx7nv/////wAD/////xzt3POPP n/8f/v/HnHP/////wAD/////xzuzjnHDh4/B//+f+f/////wAD//v9x45ndznzj58D8H/7+c/////wAD//jx4 Y9zOZzz/8Ph/8AH4j7/////wAD//z44887vc9+8eH7+f//jHn3/////wAD//88cec5nP88+Pj/HH/P/fn/ /////wAD//8cefOdznfvj4/z+UB//+f/////wAD//+ePHHPxznHzw/H5//+z/////wAD//+OHjnu5zzz58 fj+f//wH7/////wAD//P3Djznf/798/H+fH//A/n7/////wAD//vzzx7zeO8efPj/P5H4f+fP/////wAD//zx 9/znhfPPn9fz+Un/w+P/////wAD//54488/znn3z9/n8f//Hj8/////wAD//848+e/zz7798fz/H//+H/////w AD//8cefff/554/P5/h//4fH/////wAD//ePPPOec8ef/5+P8H4D+f7/////wAD//HDnHHOePnx8fz/w P/55/////wAD/////xzj3vHnz4/P9f8//H3/////wAD//v/95j73jj78fz/H//+8f/////wAD//zz//5585758/P4 f5/8H/5/////wAD//49//8+88//vz/H8D//P/////wAD//88ef//vPfv3+fx//P/f/////wAD//+eOP//3vfz// H4//+P+f/////wAD//+ffPnnv//775/Px+H/wAPx/////wAD//jzzzv9998fH8/j/9/v/////wAD//33z7xz3c c8/Px/P4H//v/z/////wAD//79845PPffn/fz/h//+P+f/////wAD//588ec93nvz/H8/8O8/4/////wAD//88+P P+7z/x/P5/H//+f/////wAD//fefnnnf894/9+fw//j/////wAD//fvJzznOe9+f//H+D/////wAD//ns 557nPPff/v//wA//3/////wAD//3/7sf9/znn3/////f/w/Xf/////wAD//375+Pc/7zz75//fz/////wAD//883H Oef/5//+f//8/////wAD//8e+znpPOe+//fP36+D/P/8/////wAD//P/zz3vvPPPvz8/P4XEAP/////wAD//f 3/55/73nv3/5/f5/n+Jf5/////wAD//94957/z59+P//4f//v/////wAD//58++89/98/fn/////wAD//+9 +eP8c//+fn4/w/////f/////wAD//+ePPP/P/vvn//n//+/////wAD//+efn33n3j3z8/////A//P/////wAD//Pz 5zzzx78+P/H8//x/////wAD//znz/955559/fH/48P/8D/////wAD//fxj79+98/8//vzj8PD//+/////wAD//5z8 8+eef//v9//H/Mf/3/////wAD//95+//fPvP/79//H//8/////wAD//9/P/Pn/n//f/x/n/wF/////wAD//+fOPH nnz3z7/vn7+fw7+/////wAD//HPn37z/7/8/55+fh/j//j/////wAD//D/z559/5//+89+/H4P/n4f/////wAD//x7 /88+8f/////vx/D/AD/////wAD//n858+e/eef//3z4fwZ//P/////wAD//x+c+efvPO/7978+D8Av/5/////wAD/ /w/fPnnnz3//88+ff4//B/B/////wAD//8fHnnnz5//+ffP//+B//oH/////wAD//8PDjjn55//fHn35/gf/B/////w AD//Hxw5588f//3zz7/H8Ajv/P/////wAD//j5588eePP/549+fh//+8f/////wAD//x8/c+fPPH98+e+P4f//J/ /////wAD//4/HOfPn33//P5/Gf4D/////wAD//38fDuPvz77+f3/n/f+CAH/z/////wAD//8fj/Hz54+/n5/4

Data Gambar	Data String
	/j/AD/4H///wAD//8+Pz/n598f/j/f9/4f///Af///wAD///nP/fw8//f39/3/j+A//AH/////wAD///nh/P4+f/vz+f 8f5/wBgA/////wAD///z5/n8fP/39/////+P/gDP/n///wAD///x8/4+Pv///7///g///+P///wAD///w8c8eH/9+/P 4f+f4B//7/////wAD///5eccfD//+fD/fPn/AHQB/////wAD//88/OPP3//fz/nn4f8AAf/////wAD//+f3vHj8//vv //z/+P///+P///wAD//+Pf/jh+f/n//P8/fgf//8/////wAD///3/7x53/zz///fD4H/8N/////wAD///z44447z5///4/3x /MAC/////wAD///z488cb59///+P74/8AH/////wAD///w/+ef/8/////n8/f//A3+///wAD///8/vfP/+f///7/f5//+ A/+///wAD///+v/nj//P//P+Pj+A/+//+///wAD///nfj1//3//j/jx/wEH//+///wAD///vjHz9/7/////w9//ADHu c///wAD///vzn9/979///+98P/nhnvv8f///wAD/////39//9+////vTz///8f///wAD///7/6/P28///7/P3/x/+H//8// //wAD/////+fH/////D+AQA8///wAD///n+/3//v///5/D/wH+EP8f///wAD///t/////vw/+A//8f///wAD /////7/f/////38f/8H//8f///wAD/////7/P///n//+4//jh54f///wAD/////H+X/3jR5f///wAD///+fj/////fj /5H//7/f///wAD/////x//8/////n5/5h/////wAD/////+/v/////3//84H/////wAD/////fv/////P9//kAH/////wA D///8/f/////j/n//4B///wAD///8v/+/////n//gf/////wAD/////9/v//v9A/////wAD/////9//+/ //+A/////wAD/////x//+f3///3/f/4AHf///wAD///n5///f///8/PP/wDf///wAD///j//v/////PH7//P///w AD/////fjx/H/////wAD/////3/////H5//gf/////wAD///f/7/////w8/j+G/////wAD/////z8 efx/mP///wAD/////5/GP+P/v///wAD/////7+fzD/B/////wAD/////n/z/////wAD/ /////P/////wAD/////v///4/8/////wAD/////v5f//4X/////wAD/////3/////8P//+D/// ///wAD/////3/////7/3P///wAD/////+/5P/P/////wAD/////b/v///7///wAD/////f7//+///7///wAD/////8/////x///7///wAD/////9/////7/w///7///wAD/////+/7//f///wAD/// /////+/+P/////wAD/////v/////wAD/////wAD/////wAD/////wAD/ /////+//////wAD/////f/////wAD/////wAD/////wAD/////wAD/ /////wAD/////f/////wAD/////wAD/////wAD/////wAD/ /////wAD/////wAD/////wAD/////wAD/////wAD/////wAA=

e. Entry Dataset ke Database

Pada tahap ini, setelah proses pengubahan *dataset* menjadi *string* selesai maka yang dilakukan penulis adalah proses penginputan data ke dalam *database* untuk membantu proses perhitungan dengan masing-masing metode yang ada. Proses penginputan data pada *database* menggunakan bantuan dari program yang telah penulis buat.

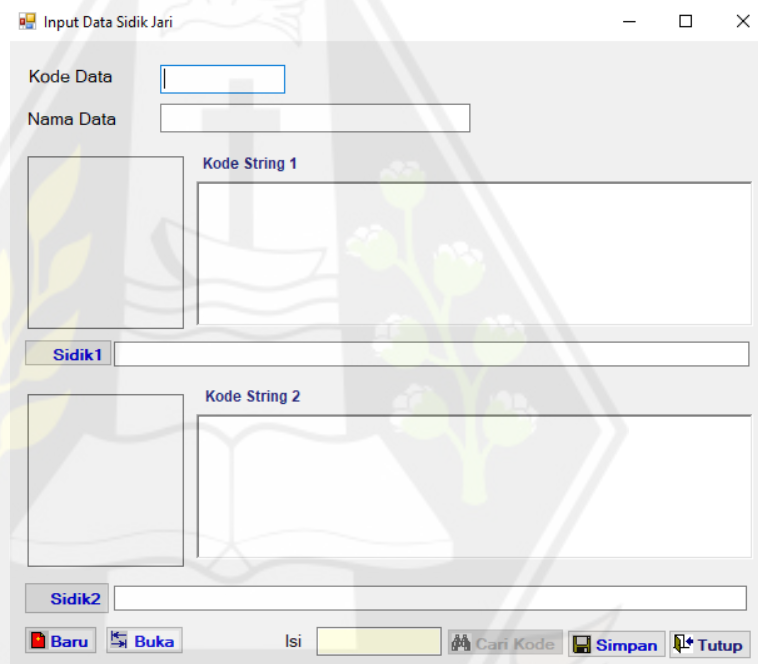
Berikut ini adalah langkah-langkah dalam proses penginputan data ke dalam *database* :

1. Tampilan Utama dari program seperti pada Gambar 4. 4.



Gambar 4. 4 Tampilan Utama Program

2. Pilih menu “Entry Data Sidik Jari” dan akan muncul tampilan seperti pada Gambar 4. 5.



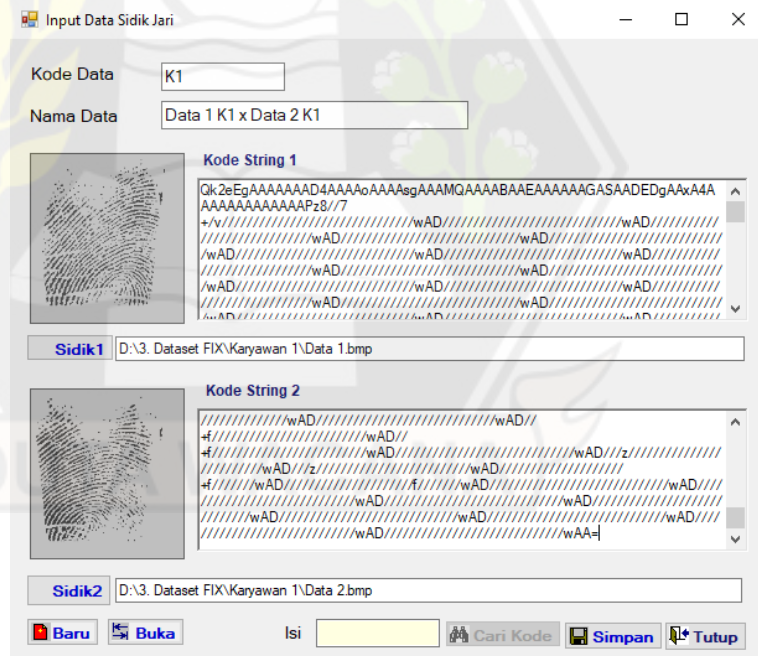
Gambar 4. 5 Menu Input Data Karyawan

3. Tahapan selanjutnya adalah proses penginputan data sidik jari yang terlihat pada Gambar 4. 6 yang dimana proses ini mempunyai ketentuan seperti :

- a. Kode data digunakan sebagai inisial pada proses penginputan data, contohnya penginputan data untuk karyawan 1, maka kode data yang digunakan adalah “K1” apabila penginputan data digunakan untuk

karyawan 1 dan karyawan 2 maka kode data yang digunakan adalah “K1K2” dan seterusnya.

- b. Nama data digunakan sebagai penanda dari data acuan dan data pembanding yang diinputkan, contohnya data yang diinputkan adalah data 1 sidik jari dan data 2 sidik jari dari karyawan 1, maka nama data yang digunakan adalah “Data 1 K1 x Data 2 K1”.
- c. Sidik 1 digunakan sebagai inputan gambar bagi data acuan.
- d. Text Box Kode String 1 digunakan sebagai inputan data string dari data sidik jadi acuan.
- e. Sidik 2 digunakan sebagai inputan gambar bagi data pembanding.
- f. Text Box Kode String 2 digunakan sebagai inputan data string dari data sidik jadi pembanding.



Gambar 4. 6 Contoh Inputan Data



Setelah data diinput dan disimpan maka data akan tertampil dalam database seperti pada Gambar 4. 7 dan Gambar 4. 8 berikut :

id_data	kode_d	nama_data	gamparsdk1	gamparsdk2	kodesdk1	kodesdk2
1008	K1	Data 1 K1 x Data 2 K1	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1009	K1	Data 1 K1 x Data 3 K1	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1010	K1	Data 1 K1 x Data 4 K1	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1011	K1	Data 1 K1 x Data 5 K1	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1012	K1	Data 1 K1 x Data 6 K1	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1015	K2	Data 1 K2 x Data 2 K2	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1016	K2	Data 1 K2 x Data 3 K2	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1017	K2	Data 1 K2 x Data 4 K2	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1018	K2	Data 1 K2 x Data 5 K2	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1019	K2	Data 1 K2 x Data 6 K2	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1020	K3	Data 1 K3 x Data 2 K3	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1021	K3	Data 1 K3 x Data 3 K3	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1022	K3	Data 1 K3 x Data 4 K3	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1023	K3	Data 1 K3 x Data 5 K3	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1024	K3	Data 1 K3 x Data 6 K3	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1025	K4	Data 1 K4 x Data 2 K4	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1026	K4	Data 1 K4 x Data 3 K4	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1027	K4	Data 1 K4 x Data 4 K4	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1028	K4	Data 1 K4 x Data 5 K4	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...

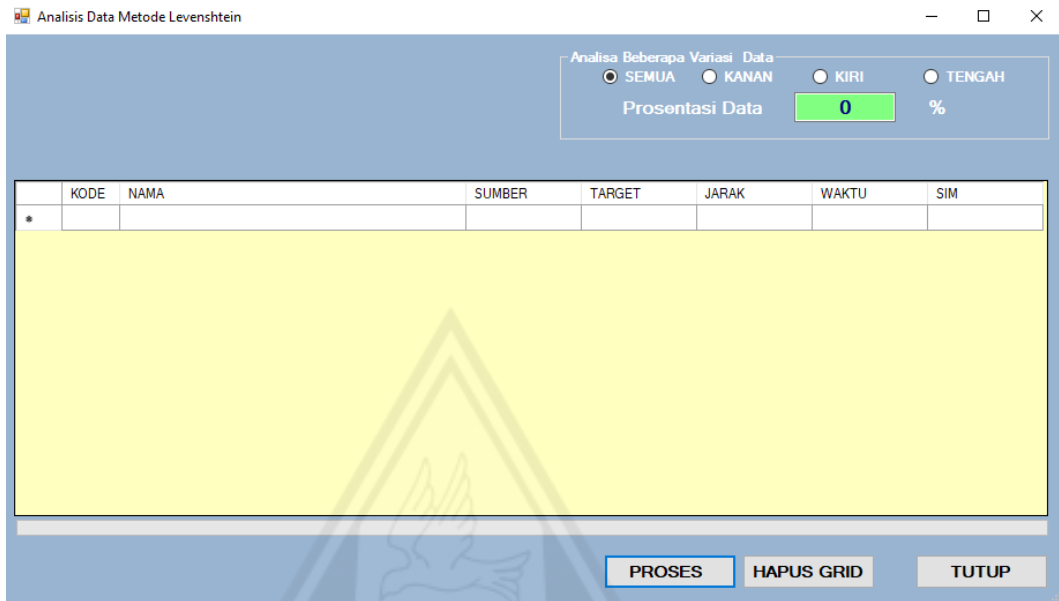
Gambar 4. 7 Data Sidik Jari Pada Database

1029	K4	Data 1 K4 x Data 6 K4	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1030	K5	Data 1 K5 x Data 2 K5	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1031	K5	Data 1 K5 x Data 3 K5	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1032	K5	Data 1 K5 x Data 4 K5	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1033	K5	Data 1 K5 x Data 5 K5	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1034	K5	Data 1 K5 x Data 6 K5	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1035	K6	Data 1 K6 x Data 2 K6	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1036	K6	Data 1 K6 x Data 3 K6	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1037	K6	Data 1 K6 x Data 4 K6	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1038	K6	Data 1 K6 x Data 5 K6	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1039	K6	Data 1 K6 x Data 6 K6	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1040	K7	Data 1 K7 x Data 2 K7	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1041	K7	Data 1 K7 x Data 3 K7	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1042	K7	Data 1 K7 x Data 4 K7	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1043	K7	Data 1 K7 x Data 5 K7	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
1044	K7	Data 1 K7 x Data 6 K7	BLOB	BLOB	Ok2eEaAA...	Ok2eEaAAAA...
	NULL	NULL	NULL	NULL	NULL	NULL

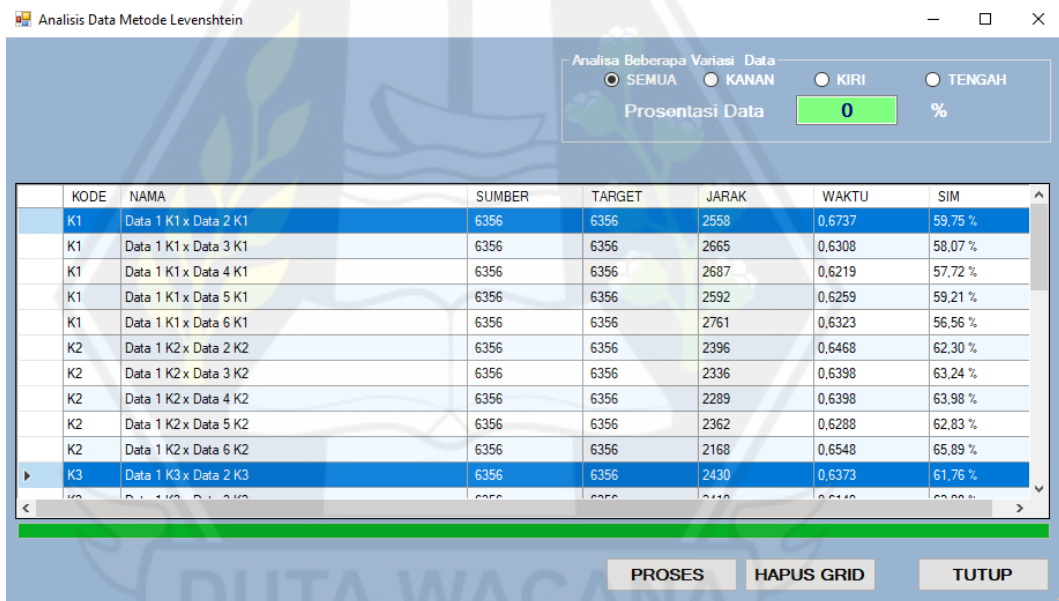
Gambar 4. 8 Data Sidik Jari Pada Database

### 4.3. Proses Perhitungan Metode Levenshtein Distance

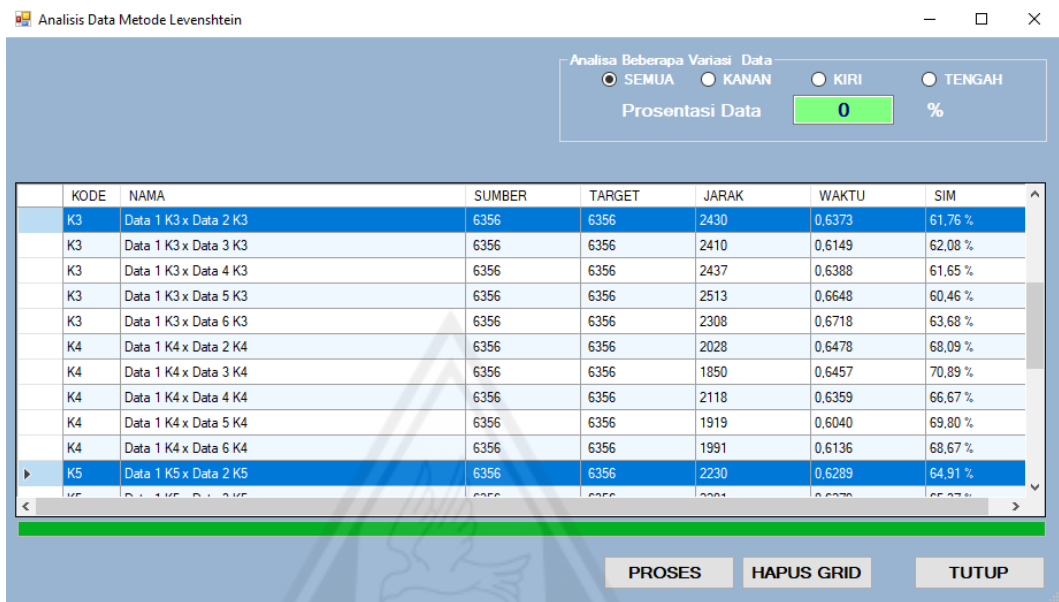
Pada tahap proses perhitungan dengan metode *levenshtein distance* penulis menggunakan program pada menu “Analisis Data Metode Levenshtein” yang telah dibuat untuk memudahkan proses perhitungan menggunakan rumus persamaan ke 1 seperti pada Gambar 4. 9.



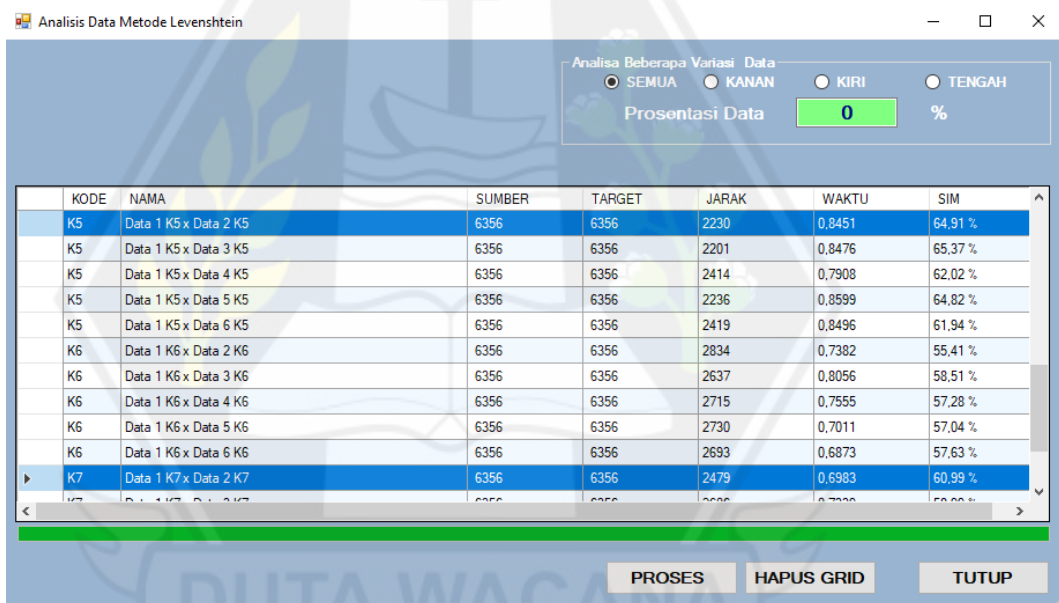
Gambar 4. 9 Tampilan Menu Analisis Data Metode Levenshtein



Gambar 4. 10 Tampilan Hasil Perhitungan Data Dengan Metode Levenshtein



Gambar 4. 11 Tampilan Hasil Perhitungan Data Dengan Metode Levenshtein



Gambar 4. 12 Tampilan Hasil Perhitungan Data Dengan Metode Levenshtein

Analisa Data Metode Levenshtein

Analisa Beberapa Variasi Data  
 SEMUA  KANAN  KIRI  TENGAH  
 Prosentasi Data  %

KODE	NAMA	SUMBER	TARGET	JARAK	WAKTU	SIM
K6	Data 1 K6 x Data 2 K6	6356	6356	2834	0.6424	55,41 %
K6	Data 1 K6 x Data 3 K6	6356	6356	2637	0.6316	58,51 %
K6	Data 1 K6 x Data 4 K6	6356	6356	2715	0.6259	57,28 %
K6	Data 1 K6 x Data 5 K6	6356	6356	2730	0.6693	57,04 %
K6	Data 1 K6 x Data 6 K6	6356	6356	2693	0.6946	57,63 %
K7	Data 1 K7 x Data 2 K7	6356	6356	2479	0.8023	60,99 %
K7	Data 1 K7 x Data 3 K7	6356	6356	2606	1.0366	58,99 %
K7	Data 1 K7 x Data 4 K7	6356	6356	2547	1.0286	59,92 %
K7	Data 1 K7 x Data 5 K7	6356	6356	2518	1.1992	60,38 %
K7	Data 1 K7 x Data 6 K7	6356	6356	2545	0.9979	59,95 %

PROSES HAPUS GRID TUTUP

Gambar 4. 13 Tampilan Hasil Perhitungan Data Dengan Metode Levenshtein

Hasil dari perhitungan yang ada pada Gambar 4. 10 sampai dengan Gambar 4. 13 dirincikan pada Tabel 4. 4.

Tabel 4. 4 Tabel Hasil Perhitungan Data Metode Levenshtein

No	Kode Data	Similarity (%)					Total (%)
		Hasil 1	Hasil 2	Hasil 3	Hasil 4	Hasil 5	
1.	K1	59,75	58,07	57,72	59,21	56,56	291,31
2.	K2	62,30	63,24	63,98	62,83	65,89	318,24
3.	K3	61,76	62,08	61,65	60,46	63,68	309,63
4.	K4	68,09	70,89	66,67	69,80	68,67	344,12
5.	K5	64,91	65,37	62,02	64,82	61,94	319,06
6.	K6	55,41	58,51	57,28	57,04	57,63	285,87
7.	K7	60,99	58,99	59,92	60,38	59,95	300,23

#### 4.3.1. Perhitungan Threshold

Berdasarkan pada hasil dari proses perhitungan dengan rumus persamaan ke 1 yang sudah dilakukan penulis dengan menggunakan metode *levenshtein distance* dan bantuan program pada menu “Analisis Data Metode Levenshtein” didapatkan nilai *similarity* dari masing-masing data setelah dibandingkan yang tertera pada Tabel 4. 4.

Tahap selanjutnya yang dilakukan adalah perhitungan dengan menggunakan rumus persamaan ke 2 untuk menemukan nilai *threshold* dari data yang sudah diolah diatas berdasarkan kode data seperti pada rincian Tabel 4. 5.

Tabel 4. 5 Tabel Hasil Perhitungan Threshold Metode Levenshtein

No	Kode Data	Total Similarity (%)	Threshold (%)
1.	K1	291,31	58,26
2.	K2	318,24	63,65
3.	K3	309,63	61,93
4.	K4	344,12	68,82
5.	K5	319,06	63,81
6.	K6	285,87	57,17
7.	K7	300,23	60,05

#### 4.3.2. Analisis Perhitungan

Perhitungan yang dilakukan selanjutnya adalah mencari presisi dan akurasi dengan rumus persamaan 3 dan persamaan 4 dari metode *levenshtein distance* yang digunakan dengan melakukan perbandingan data menggunakan Data 1 sebagai acuan dan data 2, 3, 4, 5, dan 6 sebagai pembanding.

Apabila hasil perhitungan yang didapatkan melebihi threshold yang ditentukan maka data dinyatakan mirip dengan tanda warna hijau dan apabila

hasil yang didapatkan kurang dari threshold yang ditentukan maka data dinyatakan berbeda tidak berwarna seperti pada Tabel 4. 6.

Tabel 4. 6 Tabel Sebaran Perhitungan Data Uji Metode Levenshtein

		Data Acuan (%)							Data Dikenali	
		K1	K2	K3	K4	K5	K6	K7		
		Data 1	Data 1	Data 1	Data 1	Data 1	Data 1	Data 1		
Data Uji (%)	K1	Data 2	59,75	64,36	61,89	61,72	62,49	55,45	61,13	Salah
		Data 3	58,07	61,84	59,39	59,53	60,96	54,56	59,01	Null
		Data 4	57,72	59,12	60,50	60,33	59,91	54,87	59,81	Null
		Data 5	59,21	60,11	58,93	59,06	60,71	55,19	58,08	Benar
		Data 6	56,56	60,85	59,58	60,02	58,84	53,05	59,81	Null
	K2	Data 2	56,29	62,30	58,19	58,59	58,48	52,86	59,55	Null
		Data 3	56,29	63,24	59,32	59,83	58,48	52,89	60,06	Salah
		Data 4	58,13	63,98	60,76	61,84	60,49	55,47	62,03	Benar
		Data 5	56,26	62,83	59,62	60,85	59,50	53,52	61,21	Salah
		Data 6	57,64	65,89	61,24	62,35	60,57	53,24	62,13	Benar
	K3	Data 2	57,33	56,85	61,76	61,10	60,35	54,90	59,28	Null
		Data 3	56,24	60,69	62,08	63,02	58,77	53,53	60,90	Benar
		Data 4	57,88	57,53	61,65	60,21	60,96	56,18	58,99	Null
		Data 5	60,11	60,80	60,46	60,38	62,83	56,54	59,31	Salah
		Data 6	56,62	60,91	63,68	63,57	59,97	53,53	61,78	Benar
	K4	Data 2	58,41	62,27	64,86	68,09	62,41	55,83	64,28	Salah
		Data 3	59,86	62,91	66,33	70,89	63,56	56,08	65,71	Benar
		Data 4	56,19	58,70	63,23	66,67	60,19	55,09	63,38	Salah
		Data 5	56,76	59,20	64,22	69,80	60,32	54,98	64,28	Benar
		Data 6	56,04	58,37	63,01	68,67	59,75	54,51	63,84	Salah
	K5	Data 2	60,17	61,53	64,94	65,85	64,91	56,74	64,34	Salah
		Data 3	60,98	61,45	62,93	64,47	65,37	55,66	62,14	Benar
		Data 4	58,55	60,39	59,01	58,95	62,02	55,85	58,40	Salah
		Data 5	60,84	62,25	62,71	63,48	64,82	56,26	62,39	Benar
		Data 6	58,93	58,02	59,18	59,37	61,94	54,98	58,59	Salah
	K6	Data 2	54,70	55,45	57,48	57,67	57,22	55,41	58,18	Null
		Data 3	55,85	56,32	57,85	58,07	58,63	58,51	57,85	Benar
		Data 4	55,53	56,73	57,89	58,02	57,63	57,28	57,80	Benar
Data 5		55,20	56,34	58,07	58,43	58,22	57,04	58,48	Null	
Data 6		56,84	57,81	58,59	59,14	59,75	57,63	58,73	Benar	
K7	Data 2	58,29	58,54	59,97	62,06	60,74	55,50	60,99	Benar	
	Data 3	57,15	59,18	57,64	58,65	58,79	55,09	58,99	Null	
	Data 4	58,21	59,83	58,41	59,58	59,73	55,20	59,92	Null	

			Data Acuan (%)					Data Dikenali		
			K1	K2	K3	K4	K5		K6	K7
			Data 1	Data 1	Data 1	Data 1	Data 1		Data 1	Data 1
	<b>Data 5</b>	57,37	60,25	59,56	61,56	59,72	54,79	60,38	Benar	
	<b>Data 6</b>	56,63	59,15	58,74	60,17	58,74	55,05	59,95	Null	

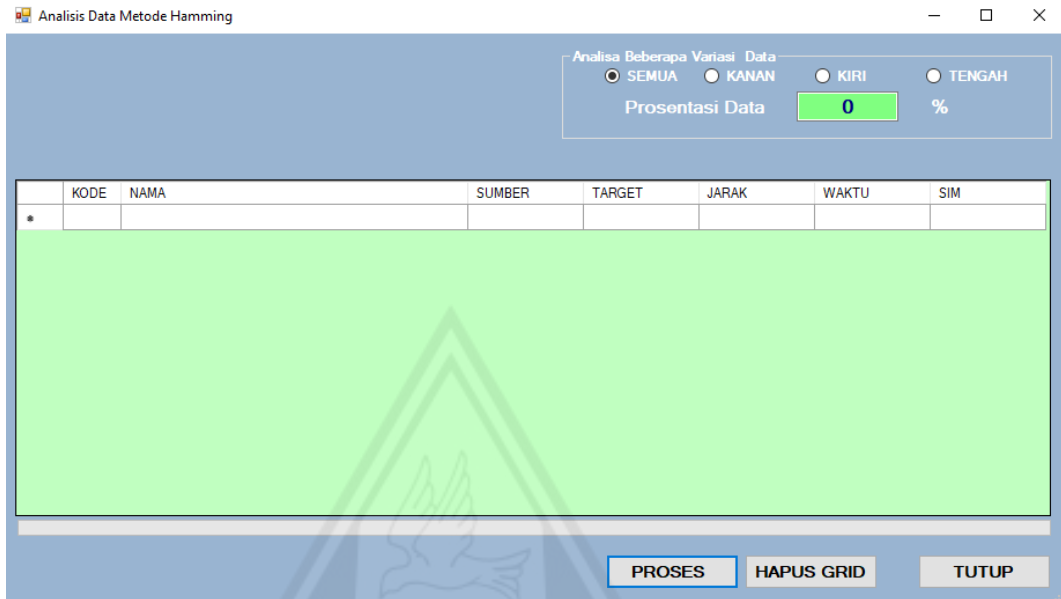
Berdasarkan perhitungan pada Tabel 4. 6 didapatkan hasil True Positif , True negatif, False positif, dan False negatif. Dengan didapatkannya hasil tersebut maka dapat dilakukan proses perhitungan presisi dan akurasi sesuai dengan rumus persamaan ke 3 dan persamaan ke 4 dan didapatkan hasil yang terlihat pada Tabel 4. 7.

Tabel 4. 7 Tabel Perhitungan Confusion Matrix Metode Levenshtein

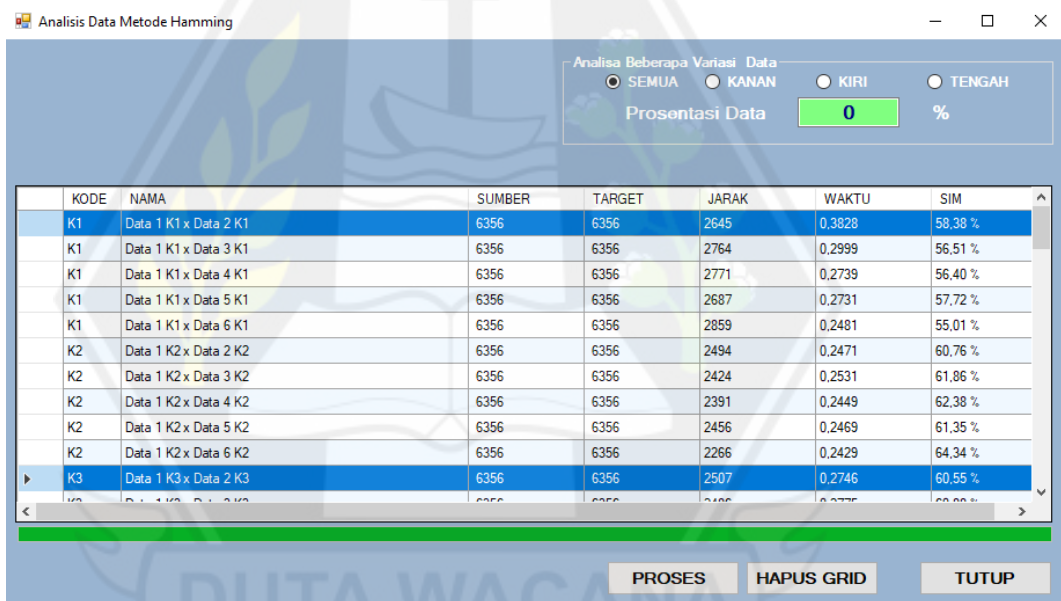
		Data Acuan							Rata-rata
		K1	K2	K3	K4	K5	K6	K7	
Data Uji	K1	2	1	0	0	0	0	1	
	K2	0	2	0	0	0	0	4	
	K3	1	0	2	0	0	0	2	
	K4	2	0	5	2	0	0	5	
	K5	5	0	3	0	3	0	3	
	K6	0	0	0	0	0	3	0	
	K7	1	0	0	0	0	0	2	
	<b>Presisi</b>	50,00%	33,00%	40,00%	14,00%	21,00%	100,00%	67,00%	46,43%
	<b>Akurasi</b>	77,55%	89,80%	77,55%	75,51%	77,55%	100,00%	67,35%	80,76%

#### 4.4. Proses Perhitungan Metode Hamming Distance

Pada tahap proses perhitungan dengan metode hamming distance penulis menggunakan program pada menu “Analisis Data Metode Hamming” yang telah dibuat untuk memudahkan proses perhitungan menggunakan rumus persamaan ke 1 seperti pada Gambar 4. 14.

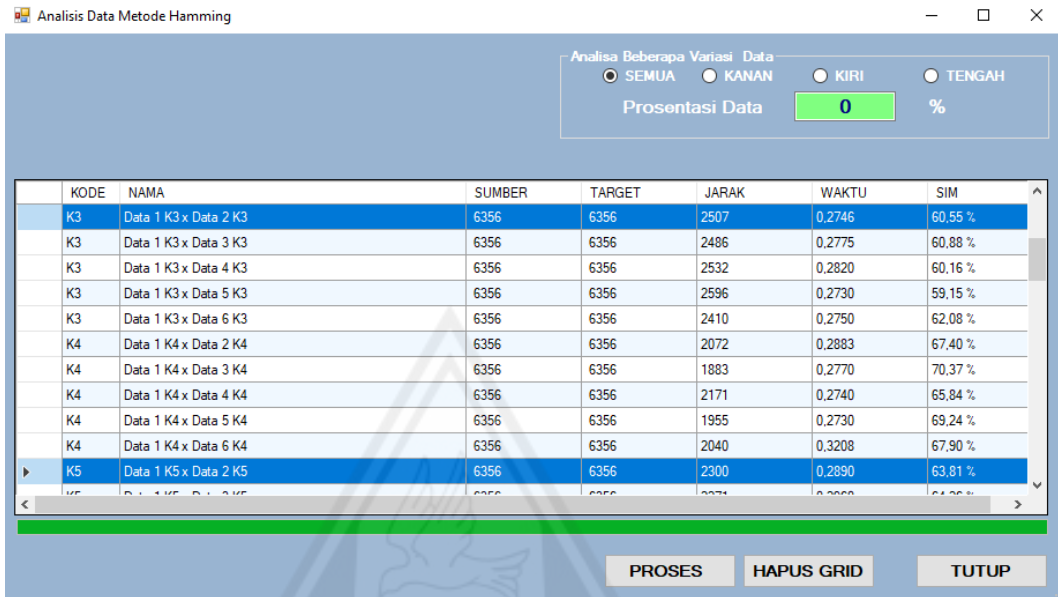


Gambar 4. 14 Tampilan Menu Analisis Data Metode Hamming

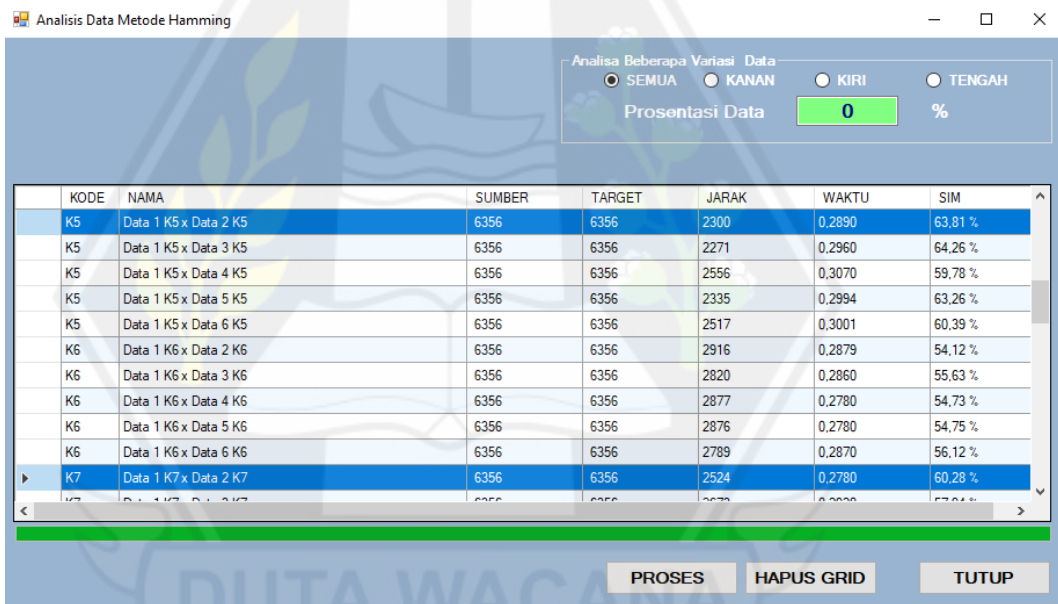


Gambar 4. 15 Tampilan Hasil Perhitungan Data Dengan Metode Hamming





Gambar 4. 16 Tampilan Hasil Perhitungan Data Dengan Metode Hamming



Gambar 4. 17 Tampilan Hasil Perhitungan Data Dengan Metode Hamming

KODE	NAMA	SUMBER	TARGET	JARAK	WAKTU	SIM
K5	Data 1 K5 x Data 6 K5	6356	6356	2517	0,3001	60,39 %
K6	Data 1 K6 x Data 2 K6	6356	6356	2916	0,2879	54,12 %
K6	Data 1 K6 x Data 3 K6	6356	6356	2820	0,2860	55,63 %
K6	Data 1 K6 x Data 4 K6	6356	6356	2877	0,2780	54,73 %
K6	Data 1 K6 x Data 5 K6	6356	6356	2876	0,2780	54,75 %
K6	Data 1 K6 x Data 6 K6	6356	6356	2789	0,2870	56,12 %
K7	Data 1 K7 x Data 2 K7	6356	6356	2524	0,2780	60,28 %
K7	Data 1 K7 x Data 3 K7	6356	6356	2673	0,2920	57,94 %
K7	Data 1 K7 x Data 4 K7	6356	6356	2617	0,2529	58,82 %
K7	Data 1 K7 x Data 5 K7	6356	6356	2574	0,2771	59,50 %
K7	Data 1 K7 x Data 6 K7	6356	6356	2617	0,2740	58,82 %

Gambar 4. 18 Tampilan Hasil Perhitungan Data Dengan Metode Hamming

Hasil dari perhitungan menggunakan rumus persamaan ke 1 tertera pada Gambar 4. 15 sampai dengan Gambar 4. 18 serta telah dirincikan pada Tabel 4. 8.

Tabel 4. 8 Tabel Hasil Perhitungan Data Metode Hamming

No	Kode Data	Similarity (%)					Total (%)
		Hasil 1	Hasil 2	Hasil 3	Hasil 4	Hasil 5	
1.	K1	58,38	56,51	56,40	57,72	55,01	284,02
2.	K2	60,76	61,86	62,38	61,35	64,34	310,69
3.	K3	60,55	60,88	60,16	59,15	62,08	302,82
4.	K4	67,40	70,37	65,84	69,24	67,90	340,75
5.	K5	63,81	64,26	59,78	63,26	60,39	311,50
6.	K6	54,12	55,63	54,73	54,75	56,12	275,35
7.	K7	60,28	57,94	58,82	59,50	58,82	295,36

#### 4.4.1. Perhitungan Threshold

Berdasarkan pada hasil dari proses perhitungan yang sudah dilakukan penulis dengan menggunakan metode *hamming distance* dan bantuan program pada menu “Analisis Data Metode Hamming” didapatkan nilai similarity dari masing-masing data setelah dibandingkan yang tertera pada Tabel 4. 8.

Tahap selanjutnya yang dilakukan adalah perhitungan dengan menggunakan rumus persamaan ke 2 untuk menemukan nilai *threshold* dari data yang sudah diolah diatas berdasarkan kode data seperti pada rincian Tabel 4. 9.

Tabel 4. 9 Tabel Hasil Perhitungan Threshold Metode Hamming

No	Kode Data	Total Similarity (%)	Threshold (%)
1.	K1	284,02	56,80
2.	K2	310,69	62,14
3.	K3	302,82	60,56
4.	K4	340,75	68,15
5.	K5	311,50	62,30
6.	K6	275,35	55,07
7.	K7	295,36	59,07

#### 4.4.2. Analisis Perhitungan

Perhitungan yang dilakukan selanjutnya adalah mencari presisi dan akurasi dengan menggunakan rumus persamaan ke 3 dan ke 4 dari metode hamming distance yang digunakan dengan melakukan perbandingan data menggunakan Data 1 sebagai acuan dan data 2, 3, 4, 5, dan 6 sebagai pembanding.

Apabila hasil perhitungan yang didapatkan melebihi threshold yang ditentukan maka data dinyatakan mirip dengan warna hijau dan apabila hasil

yang didapatkan kurang dari threshold yang ditentukan maka data dinyatakan berbeda tidak berwarna seperti pada Tabel 4. 10.

Tabel 4. 10 Tabel Perhitungan Presisi Metode Hamming

		Data Acuan (%)							Data Dikenali	
		K1	K2	K3	K4	K5	K6	K7		
		Data 1	Data 1	Data 1	Data 1	Data 1	Data 1	Data 1		
Data Uji (%)	K1	Data 2	58,38	62,69	60,68	61,21	61,32	54,51	60,36	Salah
		Data 3	56,51	60,49	58,38	59,31	59,45	53,55	58,19	Null
		Data 4	56,40	58,35	59,45	60,10	58,79	54,04	59,07	Null
		Data 5	57,72	58,79	57,66	58,60	59,28	54,18	57,25	Benar
		Data 6	55,01	59,64	58,48	59,47	57,74	52,24	58,77	Null
	K2	Data 2	55,01	60,76	57,03	58,10	57,28	51,99	58,38	Null
		Data 3	54,75	61,86	58,00	59,15	57,42	51,85	59,15	Salah
		Data 4	56,85	62,38	59,62	61,34	59,43	54,40	60,82	Benar
		Data 5	54,97	61,35	58,43	60,41	58,35	52,67	60,06	Salah
		Data 6	56,13	64,34	60,10	61,94	59,58	52,29	61,40	Benar
	K3	Data 2	57,94	55,66	60,55	60,79	59,45	53,82	58,63	Salah
		Data 3	59,48	59,70	60,88	62,63	58,16	52,64	60,35	Benar
		Data 4	55,78	56,71	60,16	59,91	60,03	54,95	58,49	Null
		Data 5	56,38	59,73	59,15	60,13	61,64	55,38	58,77	Salah
		Data 6	55,63	59,70	62,08	63,32	59,34	52,84	61,06	Benar
	K4	Data 2	58,85	61,72	64,58	67,40	61,95	55,53	63,87	Salah
		Data 3	59,47	62,39	65,87	70,37	63,18	55,77	65,35	Benar
		Data 4	57,22	58,19	62,72	65,84	59,94	54,74	62,88	Salah
		Data 5	59,29	58,66	63,82	69,24	60,11	54,68	63,86	Benar
		Data 6	57,42	58,08	62,60	67,90	59,61	54,21	63,49	Salah
	K5	Data 2	53,82	60,50	64,08	65,60	63,81	55,98	63,64	Salah
		Data 3	54,53	60,61	62,36	64,14	64,26	54,81	61,34	Benar
		Data 4	54,49	59,32	58,21	58,60	59,78	54,92	57,56	Null
		Data 5	54,07	61,12	61,84	63,02	63,26	55,49	61,48	Benar
		Data 6	55,61	56,84	58,02	59,06	60,39	53,88	57,86	Null
	K6	Data 2	57,09	54,64	56,79	57,48	56,48	54,12	57,39	Salah
		Data 3	55,78	55,39	56,74	57,85	57,42	55,63	57,26	Benar
		Data 4	57,07	55,41	56,93	57,70	56,54	54,73	57,01	Salah
Data 5		56,12	55,61	57,23	58,18	57,45	54,75	57,92	Null	
Data 6		55,31	56,73	57,74	58,65	58,55	56,12	58,08	Benar	
K7	Data 2	58,29	57,85	59,28	61,78	60,06	55,23	60,28	Benar	
	Data 3	57,15	58,02	56,79	58,21	57,89	54,18	57,94	Salah	
	Data 4	58,21	58,74	57,45	59,21	58,57	54,40	58,82	Salah	

		Data Acuan (%)							Data Dikenali
		K1	K2	K3	K4	K5	K6	K7	
		Data 1	Data 1	Data 1	Data 1	Data 1	Data 1	Data 1	
	Data 5	57,37	59,23	58,82	61,06	58,74	54,10	59,50	Benar
	Data 6	56,63	58,19	57,92	59,81	57,96	54,16	58,82	Null

Berdasarkan perhitungan pada Tabel 4. 10 didapatkan hasil True Positif , True negatif, False positif, dan False negatif. Dengan didaptkannya hasil tersebut maka dapat dilakukan proses perhitungan presisi dan akurasi sesuai dengan rumus persamaan ke 3 dan persamaan ke 4 dan didapatkan hasil yang terlihat pada Tabel 4. 11 Tabel Perhitungan Akurasi Metode Hamming Tabel 4. 11.

Tabel 4. 11 Tabel Perhitungan Akurasi Metode Hamming

		Data Acuan							Rata-rata
		K1	K2	K3	K4	K5	K6	K7	
Data Uji	K1	2	1	1	0	0	0	1	
	K2	1	2	0	0	0	0	4	
	K3	2	0	2	0	0	1	2	
	K4	5	1	5	2	1	2	5	
	K5	0	0	3	0	3	2	3	
	K6	2	0	0	0	0	2	0	
	K7	4	0	0	0	0	1	2	
Presisi		40,00%	29,00%	29,00%	9,50%	27,00%	50,00%	29,00%	30,50%
Akurasi		72,58%	88,71%	77,42%	69,35%	85,48%	87,10%	67,74%	78,34%

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1. Kesimpulan**

Pada hasil dari perhitungan yang telah dilakukan didapatkan bahwa akurasi dari metode *levenshtein distance* dalam mengukur kemiripan string pada data sidik jari yang ada dan telah diolah ke dalam bentuk string adalah sebesar 80,76% dengan rata-rata presisi data sebanyak 46,43 % dan untuk metode *hamming distance* dalam proses perhitungan kemiripan antara data string sidik jari satu dan sidik jari lainnya mempunyai nilai akurasi yang sebesar 78,34 % dengan rata-rata presisi data sebanyak 30,50 %.

Berdasarkan pada hasil yang didapatkan, dapat disimpulkan bahwa metode *levenshtein distance* merupakan metode yang lebih baik dalam perhitungan nilai kemiripan pada data sidik jari dibandingkan dengan metode *hamming distance* dikarenakan untuk deteksi kemiripan string yang dilakukan oleh keduanya, metode *levenshtein distance* mempunyai tingkat akurasi dan juga presisi yang lebih tinggi dibandingkan dengan metode *hamming distance*.

#### **5.2. Saran**

Berdasarkan dari penelitian yang telah dilakukan oleh penulis dan dengan menimbang beberapa kekurangan yang terdapat pada penelitian ini, maka berikut beberapa saran serta usulan dari penulis bagi peneliti berikutnya yang ingin melakukan penelitian serupa dengan metode yang berbeda untuk mendapatkan hasil yang lebih maksimal :

- 1 Perbanyak pengambilan data dan coba untuk melakukan *clening data* atau *preprosesing data* dengan cara atau proses yang lebih baik dan maksimal tanpa ada kecacatan pada data uji.
- 2 Mencari dan menambahkan lebih banyak referensi mengenai penggunaan metode yang tepat dalam proses perhitungan data yang ingin diuji.

- 3 Usahakan melakukan testing terlebih dahulu terhadap data real dan bukan data *dummy* dalam program yang digunakan sebagai alat bantu agar terhindar dari *errornya* program.



## DAFTAR PUSTAKA

- (n.d). Biometric Solution. Retrieved from Thales:  
<https://www.thalesgroup.com/en>
  - (n.d). Compress Images. Retrieved from Aspose:  
<https://products.aspose.app/imaging/image-compress>
  - (n.d). Image to Base64. Retrieved from CodeBeautify:  
<https://codebeautify.org/image-to-base64-converter>
  - (n.d). Resize Images. Retrieved from Aspose:  
<https://products.aspose.app/imaging/image-resize>
- Abdullah, S. F., Abas, Z. A., Rahman, A. F. N. A., & Saad, W. H. (2016). Development of a Fingerprint Gender Classification Algorithm Using Fingerprint Global Features. *International Journal of Advanced Computer Science and Applications*, 7(6), 275–279.  
<https://doi.org/10.14569/ijacsa.2016.070635>
- Adi, S. (2018). Penerapan Algoritma Rabin Karp Untuk Medeteksi Kemiripan Dua Dokumen Teks. *Jurnal Mantik Penusa*, 22(1), 125–130. <https://e-jurnal.pelitanusantara.ac.id/index.php/mantik/article/view/472/277>
- Anggelina, M., Dwi Krisnawati, L., & Sebastian, D. (2022). Penerapan Simhash dan Hamming distance dalam Deteksi kemiripan Teks Berita. *Jurnal Terapan Teknologi Informasi*, 6(2), 131–141.  
<https://doi.org/10.21460/jutei.2022.62.216>
- Azhri, M. F., Swanjaya, D., & Niswatin, R. K. (2019). Penerapan Algoritma Levenshtein Distance pada Aplikasi Asisten Guru Bahasa Inggris. *Seminar Nasional Inovasi Teknologi*, 155–160.
- B. P. Pratama, S. A. P. (2016). Analisis Kinerja Algoritma Levenshtein Distance Dalam Mendeteksi Kemiripan Dokumen Teks. *Jurnal LOGIKA*, 6(2), 131–143.
- Hartina, Sri Ayu. 2016. Pengaruh Metode Active Learning Tipe Question Students Have (QSH) Terhadap Hasil Belajar IPA Siswa Kelas V Gugus III Kecamatan Labuapi Lombok Barat Tahun Pelajaran 2018/2019. Skripsi. Universitas Mataram
- Iskandar, A., Muhajirin, & Lisah. (2017). Sistem Keamanan Pintu Berbasis Arduino Mega. *Jurnal Informatika Upgris*, 3(2), 99–104.  
<https://doi.org/10.26877/jiu.v3i2.1803>



Julian Tannga, M., Rahman, S., & Hasniati. (2017). Analisis Perbandingan Algoritma Levenshtein Distance dan Jaro Winkler Untuk Aplikasi deteksi Plagiarisme Dokumen Teks. *JTRISTE*, 4(1), 44–54.

Kopelowitz, T., & Porat, E. (2018). A Simple Algorithm for Approximating the Text-To-Pattern Hamming Distance. *OpenAccess Series in Informatics*, 61(683064), 1–5. <https://doi.org/10.4230/OASICS.SOSA.2018.10>

Kumar, T., Bhushan, S., & Jangra, S. (2021). An Improved Biometric Fusion System of Fingerprint and Face using Whale Optimization. *International Journal of Advanced Computer Science and Applications*, 12(1), 664–671. <https://doi.org/10.14569/IJACSA.2021.0120176>

