**

**EKSTRAKSI FITUR DINAMIS PADA GERAKAN TANGAN MENGGUNAKAN KINECT 2.0 UNTUK MENGENALI BAHASA ISYARAT INDONESIA**

TUGAS AKHIR - KI141502

Yahya Eka Nugyasa

NRP 5113100134

Dosen Pembimbing

Wijayanti Nurul Khotimah, S.Kom., M.Sc.

Dr.Eng. Nanik Suciati, S.Kom., M.Kom.

JURUSAN TEKNIK INFORMATIKA

Fakultas Teknologi Informasi

Institut Teknologi Sepuluh Nopember

Surabaya 2017

*[Halaman ini sengaja dikosongkan]*

**

EKSTRAKSI FITUR DINAMIS PADA GERAKAN TANGAN MENGGUNAKAN KINECT 2.0 UNTUK MENGENALI BAHASA ISYARAT INDONESIA

Yahya Eka Nugyasa

NRP 5113100134

Dosen Pembimbing

Wijayanti Nurul Khotimah, S.Kom., M.Sc.

Dr.Eng. Nanik Suciati, S.Kom., M.Kom.

JURUSAN TEKNIK INFORMATIKA

Fakultas Teknologi Informasi

Institut Teknologi Sepuluh Nopember

Surabaya 2017

TUGAS AKHIR - KI141502

*[Halaman ini sengaja dikosongkan]*

**

**EXTRACTION OF DYNAMIC FEATURES ON HAND GESTURES USING KINECT 2.0 TO RECOGNIZE INDONESIAN SIGN LANGUAGE**

YAHYA EKA NUGYASA

NRP 5113100134

Advisor

Wijayanti Nurul Khotimah, S.Kom., M.Sc.

Dr.Eng. Nanik Suciati, S.Kom., M.Kom.

INFORMATICS DEPARTMENT

Faculty of Information Technology

Institut Teknologi Sepuluh Nopember

Surabaya 2017

Su

FINAL PROJECT - KI141502

*[Halaman ini sengaja dikosongkan]*

# LEMBAR PENGESAHAN

**EKSTRAKSI FITUR DINAMIS PADA GERAKAN TANGAN MENGGUNAKAN KINECT 2.0 UNTUK MENGENALI BAHASA ISYARAT INDONESIA**

**TUGAS AKHIR**

Diajukan Untuk Memenuhi Salah Satu Syarat

Memperoleh Gelar Sarjana Komputer

pada

Bidang Studi Interaksi Grafis dan Seni

Program Studi S-1 Jurusan Teknik Informatika

Fakultas Teknologi Informasi

Institut Teknologi Sepuluh Nopember

Oleh:

**YAHYA EKA NUGYASA**

NRP. 5113100134

Disetujui oleh Pembimbing Tugas Akhir:

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | Wijayanti Nurul Khotimah, S.Kom., M.Sc. | | …………………… |
|  | NIP: 19860312 201212 2 004 | (pembimbing 1) | |
|  |  |  | |
| 2. | Dr.Eng. Nanik Suciati, S.Kom., M.Kom. | | …………………… |
|  | NIP: 19870213 201404 1 001 | (pembimbing 2) | |

**SURABAYA**

**DESEMBER, 2017**

*[Halaman ini sengaja dikosongkan]*

EKSTRAKSI FITUR DINAMIS PADA GERAKAN TANGAN MENGGUNAKAN KINECT 2.0 UNTUK MENGENALI BAHASA ISYARAT INDONESIA

Nama Mahasiswa : Yahya Eka Nugyasa

NRP : 5113100134

Jurusan : Teknik Informatika FTIf-ITS

Dosen Pembimbing I : Wijayanti Nurul Khotimah, S.Kom, M.Sc.

**Dosen Pembimbing II :** **Dr.Eng. Nanik Suciati, S.Kom., M.Kom.**

# ABSTRAK

*Bahasa isyarat adalah hal yang penting dalam komunikasi bagi orang yang menderita gangguan pendengaran. Kecepatan menguasai bahasa dan kemampuan mereka berinteraksi sangat dibutuhkan.**Mereka membutuhkan bahan pembelajaran yang tidak hanya berisi tentang komponen aural saja, namun juga secara visual karena lebih nyata.*

*Di sisi lain, teknologi berkembang pesat di segala aspek kehidupan. Berbagai macam terobosan teknologi baru telah diciptakan oleh manusia, salah satunya perangkat Kinect yang diciptakan untuk windows sekitar tahun 2012. Kinect dapat digunakan untuk mendeteksi gerakan isyarat yang diberikan dengan memanfaatkan fitur yang ada. Tugas akhir ini menggunakan fitur skeleton tracking yang ada pada Kinect untuk mendeteksi bahasa isyarat.*

*Hasil pengujian dari tugas akhir ini menunjukkan bahwa metode Back Propagation Genetic Algorithm Neural Network yang digunakan sebagai classifier gerakan isyarat memiliki akurasi yang baik yaitu sekitar 92,5 persen. Hasil tersebut masih dapat ditingkatkan dengan menambahkan data training yang diambil dari sample yang bervariasi.*

Kata kunci: Kinect 2, Fitur Dinamis, Bahasa Isyarat, SIBI.

*[Halaman ini sengaja dikosongkan]*

EXTRACTION OF DYNAMIC FEATURES ON HAND GESTURES USING KINECT 2.0 TO RECOGNIZE INDONESIAN SIGN LANGUAGE

Name : Yahya Eka Nugyasa

NRP : 5113100134

Major : Informatics Department, FTIf-ITS

Advisor I : Wijayanti Nurul Khotimah, S.Kom, M.Sc.

Advisor II : Dr.Eng. Nanik Suciati, S.Kom., M.Kom.

# ABSTRACT

Sign language is an important thing in the communication of deaf people. The speed of learning sign language and their skills to interact each other is very necessary. They need a media and the learning materials is not only about aural matter, but also visual.

On the other hand, technology growth very quickly. The new Many technology are invented, one of them is Kinect which invented by Microsoft in 2012. Kinect can be used to detect motion gesture by utilizing the existing features. This final project using the skeleton tracking feature that availavle on Kinect to detect the sign language.

The test results of this final project show that Back Propagation Genetic Algorithm Neural Network method used as classifier have good accuracy which is about 92,5 percent. These results can be improved by adding training data taken from different sample.

Keywords: : Kinect 2, Dynamic Feature, Sign Language, SIBI

*[Halaman ini sengaja dikosongkan]*

# KATA PENGANTAR

Puji dan syukur kepada Tuhan Yang Maha Esa atas rahmat dan kasih-Nya yang menyertai penulis selama proses pengerjaan tugas akhir ini sampai selesai.

Penulis ingin menyampaikan rasa hormat dan terima kasih yang setinggi-tingginya kepada pihak-pihak yang telah membantu penulis dalam penyelesaian tugas akhir ini, terutama kepada: Kedua orang tua penulis, yang selalu mendukung penulis mulai dari awal kuliah sampai lulus.

1. Ibu Wijayanti Nurul Khotimah, S.Kom., M.Sc. dan Ibu Dr.Eng. Nanik Suciati, S.Kom., M.Kom. yang telah bersedia untuk menjadi dosen pembimbing tugas akhir sehingga penulis dapat mengerjakan tugas akhir dengan arahan dan bimbingan yang baik dan jelas.
2. Teman-teman Mahasiswa Teknik Informatika 2012 yang telah berjuang bersama-sama selama menempuh pendidikan di Jurusan ini.
3. Kakak kelas Mahasiswa Teknik Informatika 2011 yang telah membantu dalam segala proses perkuliahan di TC ini.
4. Serta pihak-pihak lain yang turut membantu penulis baik secara langsung maupun tidak, yang namanya tidak penulis sebutkan disini.

Penulis telah berusaha sebaik-baiknya dalam menyusun tugas akhir ini, mohon maaf apabila ada kesalahan dan kata-kata yang dapat menyinggung perasaan. Penulis berharap tugas akhir ini dapat menjadi media pembelajaran bahasa isyarat Indonesia.

Surabaya, Mei 2017

Penulis

*[Halaman ini sengaja dikosongkan]*

# DAFTAR ISI

[LEMBAR PENGESAHAN vii](#_Toc457477452)

[ABSTRAK ix](#_Toc457477453)

[ABSTRACT xi](#_Toc457477454)

[KATA PENGANTAR xiii](#_Toc457477455)

[DAFTAR ISI xv](#_Toc457477456)

[DAFTAR GAMBAR xvii](#_Toc457477457)

[DAFTAR TABEL xix](#_Toc457477458)

[DAFTAR KODE SUMBER xxi](#_Toc457477459)

[BAB I PENDAHULUAN 1](#_Toc457477460)

[1.1 Latar Belakang 1](#_Toc457477461)

[1.2 Rumusan Masalah 2](#_Toc457477462)

[1.3 Batasan Masalah 2](#_Toc457477463)

[1.4 Tujuan 3](#_Toc457477464)

[1.5 Manfaat 3](#_Toc457477465)

[1.6 Metodologi 3](#_Toc457477466)

[1.7 Sistematika Penulisan 5](#_Toc457477467)

[BAB II DASAR TEORI 7](#_Toc457477468)

[2.1 Tunarungu 7](#_Toc457477471)

[2.2 Kinect 7](#_Toc457477472)

[2.3 Kinect SDK 9](#_Toc457477473)

[2.4 Bahasa Isyarat 10](#_Toc457477474)

[2.5 *Decision Tree* 11](#_Toc457477475)

[2.6 *Neural* *Network* 12](#_Toc457477476)

[2.7 *Back Propagation* 13](#_Toc457477477)

[2.8 *Back Propagation Genetic* *Algorithm* *Neural* *Network* 14](#_Toc457477478)

[BAB III ANALISIS DAN PERANCANGAN SISTEM 17](#_Toc457477479)

[3.1 Analisis Perangkat Lunak 17](#_Toc457477482)

[3.1.1 Deskripsi Umum Perangkat Lunak 18](#_Toc457477483)

[3.1.2 Spesifikasi Kebutuhan Perangkat Lunak 18](#_Toc457477484)

[3.1.3 Identifikasi Pengguna 19](#_Toc457477485)

[3.2 Perancangan Perangkat Lunak 19](#_Toc457477486)

[3.2.1 Model Kasus Penggunaan 20](#_Toc457477487)

[3.2.2 Definisi Aktor 20](#_Toc457477488)

[3.2.3 Definisi Kasus Penggunaan 21](#_Toc457477489)

[3.2.4 Arsitektur Umum Sistem 25](#_Toc457477490)

[3.2.5 Rancangan Antarmuka Aplikasi 25](#_Toc457477491)

[3.2.6 Rancangan Proses Aplikasi 26](#_Toc457477492)

[BAB IV IMPLEMENTASI 35](#_Toc457477493)

[4.1 Lingkungan Pembangunan 35](#_Toc457477497)

[4.1.1 Lingkungan Pembangunan Perangkat Keras 35](#_Toc457477498)

[4.1.2 Lingkungan Pembangunan Perangkat Lunak 35](#_Toc457477499)

[4.2 Implementasi Antarmuka 35](#_Toc457477500)

[4.3 Implementasi Aplikasi 36](#_Toc457477501)

[4.3.1 Implementasi Pendeteksian *Skeleton* Pengguna 36](#_Toc457477502)

[4.3.2 Implementasi Proses Ekstraksi Fitur 38](#_Toc457477503)

[4.3.3 Implementasi Proses Normalisasi Fitur 40](#_Toc457477504)

[4.3.4 Implementasi Proses Pembuatan *Rule* 42](#_Toc457477505)

[4.3.5 Implementasi Proses*, dan* *Testing* 43](#_Toc457477506)

[BAB V PENGUJIAN DAN EVALUASI 49](#_Toc457477507)

[5.1 Lingkungan Pembangunan 49](#_Toc457477508)

[5.2 Skenario Pengujian 49](#_Toc457477509)

[5.2.1 Pengujian Skenario A1 dan Analisis 50](#_Toc457477510)

[5.2.2 Pengujian Skenario A2 dan Analisis 52](#_Toc457477511)

[5.2.3 Pengujian Skenario B1 dan Analisis 53](#_Toc457477512)

[5.2.4 Pengujian Skenario B2 dan Analisis 54](#_Toc457477513)

[5.3 Evaluasi 55](#_Toc457477514)

[BAB VI KESIMPULAN DAN SARAN 57](#_Toc457477515)

[6.1 Kesimpulan 57](#_Toc457477516)

[6.2 Saran 58](#_Toc457477517)

[DAFTAR PUSTAKA 59](#_Toc457477518)

[LAMPIRAN A KODE SUMBER [13] 61](#_Toc457477519)

[LAMPIRAN B SCREENSHOT APLIKASI 75](#_Toc457477520)

[BIODATA PENULIS 79](#_Toc457477521)

# DAFTAR GAMBAR

[Gambar 2.1 Microsoft Kinect [6] 8](#_Toc457477522)

[Gambar 2.2 *Depth, Skeleton,* dan *VGA view* [7] 8](#_Toc457477523)

[Gambar 2.3 *Skeleton Joints* yang diketahui Kinect 9](#_Toc457477524)

[Gambar 2.4 *Skeleton Joints* yang Digunakan 10](#_Toc457477525)

[Gambar 2.5 Bahasa Isyarat yang Dipakai 11](#_Toc457477526)

[Gambar 2.6Contoh *Decision Tree* 12](#_Toc457477527)

[Gambar 2.7 Contoh *Neural Network* Dengan 1 *Hidden Layer* 13](#_Toc457477528)

[Gambar 3.1 Diagram Kasus Penggunaan Aplikasi 20](#_Toc457477529)

[Gambar 3.2 Arsitektur Umum Sistem 25](#_Toc457477530)

[Gambar 3.3 Rancangan Antarmuka Aplikasi 26](#_Toc457477531)

[Gambar 3.4 Diagram Alir Proses Ekstraksi Fitur 27](#_Toc457477532)

[Gambar 3.5 Diagram Alir Proses Normalisasi Fitur 29](#_Toc457477533)

[Gambar 3.6 Diagram Alir Proses Pembuatan *Rule* 31](#_Toc457477534)

[Gambar 3.7 Hasil Metode *Decision Tree* 31](#_Toc457477535)

[Gambar 3.8 Level 1 *Decision Tree* 31](#_Toc457477536)

[Gambar 3.9 Diagram Alir Proses *Training* 33](#_Toc457477537)

[Gambar 3.10 Diagram Alir Proses *Testing* 34](#_Toc457477538)

[Gambar 4.1 Antarmuka Aplikasi 36](#_Toc457477539)

[Gambar 5.1 Hasil Pengujian Tingkat Akurasi Model BPGANN 56](#_Toc457477540)

[Gambar B.1 Aplikasi Mendeteksi Bahasa Isyarat Hai 75](#_Toc457477541)

[Gambar B.2 Aplikasi Mendeteksi Bahasa Isyarat Ketua 75](#_Toc457477542)

[Gambar B.3 Aplikasi Mendeteksi Bahasa Isyarat Hormat 76](#_Toc457477543)

[Gambar B.4 Aplikasi Mendeteksi Bahasa Isyarat Bentuk 76](#_Toc457477544)

[Gambar B.5 Aplikasi Mendeteksi Bahasa Isyarat Wadah 77](#_Toc457477545)

[Gambar B.6 Aplikasi Mendeteksi Bahasa Isyarat Alquran 77](#_Toc457477546)

[Gambar B.7 Aplikasi Mendeteksi Bahasa Isyarat Gang 78](#_Toc457477547)

[Gambar B.8 Aplikasi Mendeteksi Bahasa Isyarat Hamba 78](#_Toc457477548)

*[Halaman ini sengaja dikosongkan]*

# DAFTAR TABEL

[Tabel 3.1 Definisi Kasus Penggunaan 20](#_Toc457477549)

[Tabel 3.2 Definisi Kasus Penggunaan 21](#_Toc457477550)

[Tabel 3.3 Spesifikasi Kasus Penggunaan Membuat Data Bahasa Isyarat Baru 21](#_Toc457477551)

[Tabel 3.4 Spesifikasi Kasus Penggunaan *Training Dataset* 23](#_Toc457477552)

[Tabel 3.5 Spesifikasi Kasus Penggunaan *Testing Dataset* 24](#_Toc457477553)

[Tabel 3.6 Fituryang Digunakan 28](#_Toc457477554)

[Tabel 3.7 *Rule* Pembagian Model BPGANN 32](#_Toc457477555)

[Tabel 5.1 Skenario Pengujian A1 51](#_Toc457477556)

[Tabel 5.2 Terjemahan Isyarat Kata Skenario Pengujian A1 51](#_Toc457477557)

[Tabel 5.3 Skenario Pengujian A2 52](#_Toc457477558)

[Tabel 5.4 Terjemahan Isyarat Kata Skenario Pengujian A2 53](#_Toc457477559)

[Tabel 5.5 Skenario Pengujian B1 53](#_Toc457477560)

[Tabel 5.6 Terjemahan Isyarat Kata Skenario Pengujian B1 54](#_Toc457477561)

[Tabel 5.7 Skenario Pengujian B2 54](#_Toc457477562)

[Tabel 5.8 Terjemahan Isyarat Kata Skenario Pengujian B2 55](#_Toc457477563)

*[Halaman ini sengaja dikosongkan]*

# DAFTAR KODE SUMBER

[Kode Sumber 4.1 Kode Sumber Integrasi Kinect 37](#_Toc457477564)

[Kode Sumber 4.2. Kode Sumber Deteksi *Skeleton* Pengguna 38](#_Toc457477565)

[Kode Sumber 4.3 Kode Sumber Ekstraksi Fitur *Skeleton* 40](#_Toc457477566)

[Kode Sumber 4.4 Fungsi Normalisasi Fitur 41](#_Toc457477567)

[Kode Sumber 4.5 Implementasi fungsi VectorFeatureNorm(), AngleFeatureNorm(), dan DistanceFeatureNorm() pada class Features 42](#_Toc457477568)

[Kode Sumber 4.6 Implementasi Proses Pembuatan Rule 43](#_Toc457477569)

[Kode Sumber 4.7 Implementasi Proses *Traning* 45](#_Toc457477570)

[Kode Sumber 4.8 Implementasi Proses Testing 47](#_Toc457477571)

[Kode Sumber A.1 Kelas BackPropagation.cs 63](#_Toc457477572)

[Kode Sumber A.2 Kelas Chromosom.cs 64](#_Toc457477573)

[Kode Sumber A.3 Kelas ClassificationClass.cs 65](#_Toc457477574)

[Kode Sumber A.4 Kelas DataSet.cs 66](#_Toc457477575)

[Kode Sumber A.5 Kelas FeedForward.cs 68](#_Toc457477576)

[Kode Sumber A.6 Kelas GeneticAlgorithm.cs 72](#_Toc457477577)

[Kode Sumber A.7 Kelas Neural Network.cs 74](#_Toc457477578)

*[Halaman ini sengaja dikosongkan]*

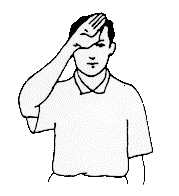
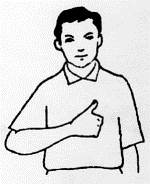
# BAB I PENDAHULUAN

Pada bab ini akan dipaparkan mengenai garis besar tugas akhir yang meliputi latar belakang, tujuan, rumusan, batasan permasalahan, dan manfaat.

## 1.1 Latar Belakang

Bahasa isyarat merupakan media bagi para penderita tuna rungu dan tuna wicara untuk berkomunikasi dengan sekitarnya. Gerakan visual tubuh sangat membantu penderita agar yang ingin disampaikannya lebih mudah dimengerti oleh pasangan komunikasinya [1]. Di Indonesia sendiri, para penderita tuna rungu dan tuna wicara berkomunikasi menggunakan bahasa isyarat yang mengacu pada dua sistem yaitu BISINDO (Berkenalan Dengan Sistem Isyarat Indonesia) dan SIBI (Sistem Isyarat Bahasa Indonesia). BISINDO dikembangkan oleh orang tuna rungu sendiri melalui GERKATIN (Gerakan Kesejahteraan Tuna Rungu Indonesia). SIBI dikembangkan oleh orang normal, bukan-penderita tuna rungu. SIBI sama dengan bahasa isyarat yang digunakan di Amerika yaitu ASL (*American Sign-Language*) [2].

Bahasa isyarat diklasifikasikan dalam dua kategori: statis dan dinamis. Isyarat statis adalah isyarat yang tidak melibatkan suatu pergerakan dalam penyampaiannya [2]. Seperti yang terlihat pada Gambar 1.



(1) (2) (3)

Gambar 1.1 Contoh pemakaian bahasa isyarat statis yang mempunyai arti: (1) baik; (2) demam; (3) maaf

Sedangkan isyarat dinamis adalah isyarat yang melibatkan pergerakan dalam penyampaiannya [2]. Seperti yang terlihat pada Gambar 2.



(1) (2) (3)

Gambar 1.2 Contoh pemakaian bahasa isyarat dinamis yang mempunyai arti: (1) awan; (2) adik; (3) agama

Sebelumnya sudah ada Tugas Akhir yang dibuat oleh Yohanes Aditya Sutanto tentang pengenalan bahasa isyarat Indonesia menggunakan teknologi Kinect 1.0. Tetapi Tugas Akhir tersebut hanya dapat mendeteksi bahasa isyarat statis. Oleh karena itu, dengan kemampuan yang dimiliki oleh Kinect 2.0, munculah ide untuk membuat pengenalan bahasa isyarat Indonesia yang dapat mendeteksi bahasa isyarat dinamis.

Untuk dapat mendeteksi bahasa isyarat dinamis, dibutuhkan pula fitur yang dinamis dalam memproses gerakan yang diberikan. Sehingga pada penelitian ini diusulkan suatu aplikasi yang dapat mengekstraksi fitur dinamis pada bahasa isyarat. Dengan adanya aplikasi tersebut, diharapkan dapat memudahkan komunikasi menggunakan bahasa isyarat dinamis yang digunakan oleh penderita tuna rungu dan tuna wicara dengan mengacu pada Sistem Isyarat Bahasa Indonesia (SIBI).

## 1.2 Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana mengekstraksi fitur dinamis pada gerakan tangan dalam mendeteksi bahasa isyarat?
2. Bagaimana menggunakan fitur dinamis untuk mengenali gerakan yang telah ditentukan oleh pengguna?
3. Bagaimana menggunakan hasil klasifikasi gerakan untuk mendeteksi bahasa isyarat?

## 1.3 Batasan Masalah

Permasalahan yang dibahas dalam tugas akhir ini memiliki beberapa batasan, di antaranya sebagai berikut:

1. Teknologi yang dipakai adalah Kinect 2.0.
2. Gerakan tangan yang dapat dikenali berjumlah 10 gerakan. Yaitu gerakan lurus, berulang, gelombang, kurva, rotasi, ganda, memisah, menyatu, persegi, dan gerakan melingkar.
3. Bahasa isyarat yang dideteksi hanya yang bersifat dinamis.
4. Klasifikasi yang digunakan untuk mengklasifikasi gerakan adalah *Decision Tree*.

## 1.4 Tujuan

Tujuan dari pembuatan tugas akhir ini adalah mengekstraksi fitur dinamis pada gerakan tangan menggunakan Kinect 2.0 untuk mengenali bahasa isyarat Indonesia.

## 1.5 Manfaat

Tugas Akhir ini diharapkan dapat mengenali bahasa isyarat menggunakan anggota tubuh yang bergerak (dinamis) sehingga dapat membantu orang berkebutuhan khusus yaitu tuna rungu dan tuna wicara dalam berkomunikasi dengan orang yang tidak berkebutuhan khusus.

## 1.6 Metodologi

Pembuatan tugas akhir ini dilakukan menggunakan metodologi sebagai berikut:

1. Studi literatur

Pada tahap ini, dicari studi literatur yang relevan untuk dijadikan referensi dalam pengerjaan tugas akhir. Studi literatur dapat diambil dari buku, internet, maupun materi dalam suatu mata kuliah yang berhubungan dengan metode yang akan digunakan.

1. Perancangan perangkat lunak



Gambar 1.3 Alur aplikasi yang akan dibuat

Analisa dimulai dari pengguna yang memberikan masukan berupa gerakan tangan dan posisi gerakan tersebut dari badan ke alat Kinect 2.0. Kemudian masukan tersebut diproses dengan *classifier* yang telah ditentukan yaitu *Decision Tree*. Dan pada akhirnya, keluaran dari proses tersebut ditampilkan menggunakan aplikasi perangkat lunak berbasis desktop.

1. Implementasi dan pembuatan sistem

Pembangunan aplikasi dilakukan dengan menggunakan bahasa pemrograman C#, Integrated Development Environment (IDE) Microsoft Visual Studio dan Kinect for Windows SDK 2.0.

1. Uji coba dan evaluasi

Pengujian akan dilakukan oleh satu orang pengguna. Pengguna tersebut akan diminta untuk melakukan gerakan yang telah ditentukan kemudian dihitung akurasi gerakan tersebut dari aplikasi yang telah dibuat.

1. Penyusunan laporan tugas akhir

Pada tahap ini dilakukan penyusunan laporan yang berisi dasar teori, dokumentasi dari perangkat lunak, dan hasil-hasil yang diperoleh selama pengerjaan tugas akhir.

## 1.7 Sistematika Penulisan

Buku tugas akhir ini terdiri dari beberapa bab, yang dijelaskan sebagai berikut:

**BAB I PENDAHULUAN**

Bab ini berisi latar belakang masalah, rumusan dan batasan masalah, tujuan dan manfaat pembuatan tugas akhir, metodologi yang digunakan, dan sistematika penyusunan tugas akhir.

**BAB II TINJAUAN PUSTAKA**

Bab ini membahas dasar pembuatan dan beberapa teori penunjang yang berhubungan dengan pokok pembahasan yang mendasari pembuatan tugas akhir ini.

**BAB III ANALISIS DAN PERANCANGAN**

Bab ini membahas analisis dari sistem yang dibuat meliputi analisis permasalahan, deskripsi umum perangkat lunak, spesifikasi kebutuhan, dan identifikasi pengguna. Kemudian membahas rancangan dari sistem yang dibuat meliputi rancangan skenario kasus penggunaan, arsitektur, data, dan antarmuka.

**BAB IV IMPLEMENTASI**

Bab ini membahas implementasi dari rancangan sistem yang dilakukan pada tahap perancangan. Penjelasan implementasi meliputi implementasi antarmuka aplikasi dan pembuatan kebutuhan fungsional aplikasi

**BAB V PENGUJIAN DAN EVALUASI**

Bab ini membahas pengujian dari aplikasi yang dibuat dengan melihat keluaran yang dihasilkan oleh aplikasi dan evaluasi untuk mengetahui kemampuan aplikasi.

**BAB VI PENUTUP**

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan serta saran untuk pengembangan aplikasi selanjutnya.

# BAB II DASAR TEORI

Pada bab ini akan dibahas mengenai dasar teori yang menjadi dasar pembuatan tugas akhir ini. Pokok permasalahan yang akan di bahas mengenai teknologi yang mendukung dalam pembuatan tugas akhir seperti Kinect 2.0, Kinect SDK, *Decision Tree*, dan pengetahuan umum mengenai bahasa isyarat.



## 2.1 Tunarungu

Tuna rungu dapat diartikan sebagai keterbatasan yang dimiliki seseorang dalam mendengar sesuatu karena tidak berfungsinya organ pendengaran yang dimilikinya. Ketunarunguan dapat dibedakan menjadi dua kategori yaitu tuli (deaf) dan kurang dapat mendengat (low hearing) [3]. Tuli adalah keadaan dimana organ pendengaran telah mengalami kerusakan yang sangat parah dan mengakibatkan tidak berfungsinya pendengaran. Sedangkan kurang dapat mendengar adalah keadaan dimana organ pendengaran mengalami kerusakan tetapi masih dapat berfungsi untuk mendengar.

## 2.2 Bahasa Isyarat

Bahasa isyarat adalah sarana berkomunikasi bagi penderita tuna rungu. Bahasa isyarat dikembangkan dan memiliki karakteristik sendiri di berbagai negara. Di Indonesia, bahasa isyarat yang digunakan berdasarkan pada Sistem Isyarat Bahasa Indonesia (SIBI). Ada 4 jenis bahasa isyarat dalam SIBI [4] yaitu:

1. Isyarat Pokok: melambangkan sebuah kata atau konsep;

2. Isyarat Tambahan: melambangkan awalan, akhiran, dan

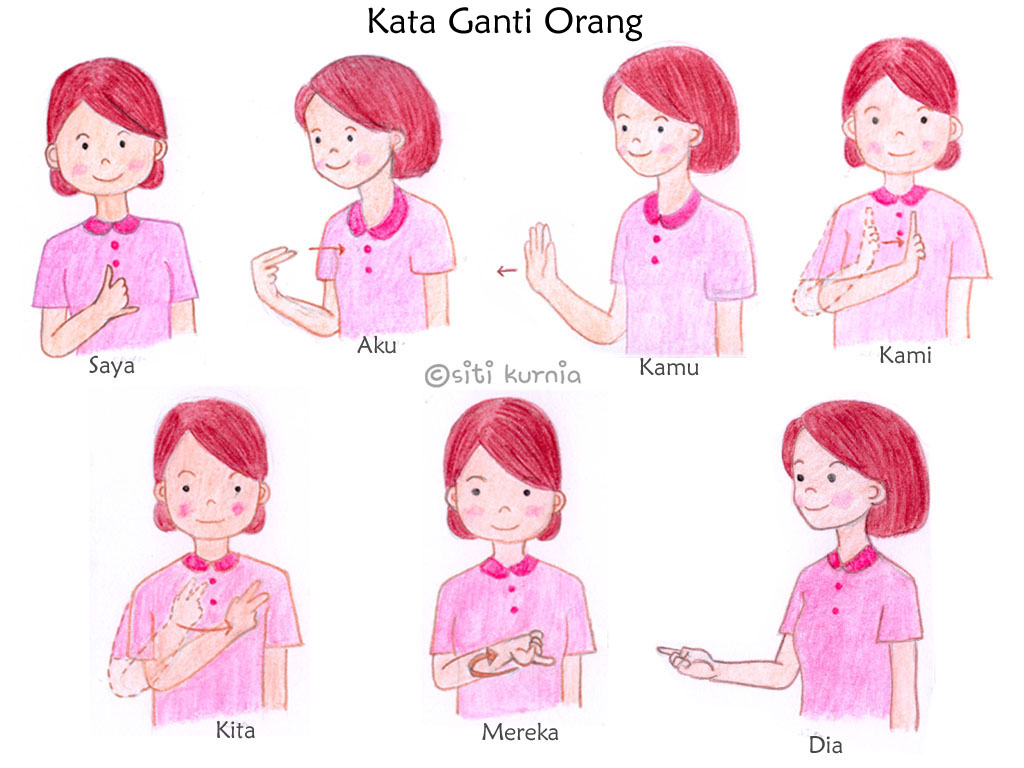
partikel (imbuhan);

3. Isyarat Bentukan: dibentuk dengan menggabungkan

isyarat pokok dan isyarat tambahan;

4. Abjad Jari: dibentuk dengan jari-jari untuk mengeja huruf.

Pada Tugas Akhir ini, isyarat yang akan digunakan adalah isyarat bahasa pokok. Gambar 3 menunjukkan contoh bahasa isyarat yang ada di dalam SIBI.



Gambar 2.1 Contoh bahasa isyarat yang ada di dalam SIBI [5]

## 2.3 Kinect 2.0

Generasi kedua dari Kinect yang dirilis oleh Microsoft pada tahun 2014 adalah versi terbaru Kinect dari yang pertama kali dikeluarkan pada tahun 2010. Perangkat Kinect 2.0 seperti yang terlihat pada Gambar 4, terdapat tiga lensa yaitu kamera RGB yang digunakan untuk menangkap spektrum warna, *infrared emitters* yang memproyeksikan spektrum inframerah dan sensor kedalaman yang menghasilkan gambar mendalam dari seseorang atau objek dengan menganalisis informasi inframerah. Dan sebuah *microphone* array yang dapat menemukan lokasi timbulnya suara. Alhasil, ada enam sumber data yang dihasilkan, termasuk warna, inframerah, kedalaman, indeks tubuh, tubuh, dan suara [6].



Gambar 2.2 Kinect 2.0

## 2.4 Kinect SDK

Kinect SDK adalah pustaka yang dibuat oleh Microsoft untuk pengembangan aplikasi perangkat lunak yang menggunakan Kinect sebagai alat input utama. Kinect SDK dapat diimplementasikan dengan bahasa pemrograman C#, C++, dan JavaScript. Pustaka ini memiliki beberapa fitur diantaranya *skeleton tracking*, *thumb tracking*, *end of hand tracking*, *open/close hand gesture* dan lainnya [7].

## 2.5 Microsoft Visual Studio

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi berbasis *console*, Windows, ataupun berbasis *website* [8].

## 2.6 Decision Tree

Pohon Keputusan atau dikenal dengan *Decision Tree* adalah salah satu metode klasifikasi yang menggunkan representasi suatu struktur pohon yang yang berisi alternatif-alternatif untuk pemecahan suatu masalah. Pohon ini juga menunjukkan faktor-faktor yang memengaruhi hasil alternatif dari keputusan tersebut disertai dengan estimasi hasil akhir bila kita mengambil keputusan tersebut. Peranan pohon keputusan ini adalah sebagai *Decision Support Tool* untuk membantu manusia dalam mengambil suatu keputusan. Manfaat dari *Decision Tree* adalah melakukan *break down* proses pengambilan keputusan yang kompleks menjadi lebih mudah sehingga orang yang mengambil keputusan akan lebih menginterpretasikan solusi dari permasalahan. Konsep yang digunakan oleh decision tree adalah mengubah data menjadi suatu keputusan pohon dan aturan-aturan keputusan (*rule*).

*Decision Tree* menggunakan struktur hierarki untuk pembelajaran *supervised*. Proses dari *Decision Tree* dimulai dari *root node* hingga *leaf node* yang dilakukan secara rekursif. Di mana setiap percabangan menyatakan suatu kondisi yang harus dipenuhi dan pada setiap ujung pohon menyatakan kelas dari suatu data. Pada *Decision Tree* terdiri dari tiga bagian yaitu:

a. Root *Node*

*Node* ini merupakan *node* yang terletak paling atas dari suatu pohon

b. Internal *Node*

*Node* ini merupakan *node* percabangan, hanya terdapat satu input serta mempunyai minimal dua output

c. Leaf *Node*

*Node* ini merupakan *node* akhir, hanya memiliki satu masukan, dan tidak mempunyai keluaran.

1). Algoritma Decision Tree C4.5

Pohon dibangun dengan cara membagi data secara rekursif hingga tiap bagian terdiri dari data yang berasal dari kelas yang sama. Bentuk pemecahan (*split*) yang digunakan untuk membagi data tergantung dari jenis atribut yang digunakan dalam *split*. Algoritma C4.5 dapat menangani data numerik (kontinyu) dan diskrit. *Split* untuk atribut numerik yaitu mengurutkan contoh berdasarkan atribut kontiyu A, kemudian membentuk minimum permulaan (*threshold*) M dari contoh-contoh yang ada dari kelas mayoritas pada setiap partisi yang bersebelahan, lalu menggabungkan partisi-partisi yang bersebelahan tersebut dengan kelas mayoritas yang sama. *Split* untuk atribut diskrit A mempunyai bentuk nilai (A) ε X dimana X ⊂ domain(A).

Jika suatu set data yang mempunyai beberapa pengamatan dengan *missing value* (yaitu *record* dengan beberapa nilai variabel) tidak ada, dan jika jumlah pengamatan terbatas, maka atribut dengan *missing value* dapat diganti dengan nilai rata-rata dari variabel yang bersangkutan.

Dalam melakukan pemisahan obyek dilakukan tes terhadap atribut dengan mengukur tingkat ketidakmurnian pada sebuah *node*. Pada algoritma C4.5 menggunakan rasio perolehan (*gain ratio*). Sebelum menghitung *gain ratio*, perlu menghitung dulu nilai informasi dalam satuan bits dari suatu kumpulan objek. Cara menghitungnya dilakukan dengan menggunakan konsep entropi. Rumus untuk menghitung entropi adalah sebagai berikut:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | (1.0) |

S adalah ruang (data) sampel yang digunakan untuk pelatihan, p+ adalah jumlah yang bersolusi positif atau mendukung pada data sampel untuk kriteria tertentu. Dan p- adalah jumlah yang bersolusi negatif atau tidak mendukung pada data sampel untuk kriteria tertentu. Entropi(S) sama dengan 0, jika semua contoh pada S berada dalam kelas yang sama. Entropi(S) sama dengan 1, jika jumlah contoh positif dan negatif dalam S adalah sama. Entropi(S) lebih dari 0 tetapi kurang dari 1, jika jumlah contoh positif dan negatif dalam S tidak sama.

Kemudian menghitung perolehan informasi dari keluaran data atau variabel dependen y yang dikelompokan berdasarkan atribut A, dinotasikan dengan Gain(y, A). Perolehan informasi Gain(y, A) dari atribut A relatif terhadap keluaran data y adalah:

|  |  |  |
| --- | --- | --- |
|  |  | (1.1) |

Nilai (A) adalah semua nilai yang mungkin dari atribut A, dan yc adalah subjek dari y dimana A mempunyai nilai c. *Term* pertama dalam persamaan diatas adalah emtropi total y dan *term* kedua adalah entropi sesudah dilakukan pemisahan data berdasarkan atribut A.

Untuk menghitung rasio perolehan perlu diketahui suatu *term* baru yang disebut pemisahan informasi (*SplitInfo*). Pemisahan informasi dihitung dengan cara:

|  |  |  |
| --- | --- | --- |
|  |  | (1.2) |

S1 sampai Sc adalah c subset yang dihasilkan dari pemecahan S dengan menggunakan atribut A yang mempunyai sebanyak c nilai. Selanjutnya *gain ratio* dihitung dengan cara:

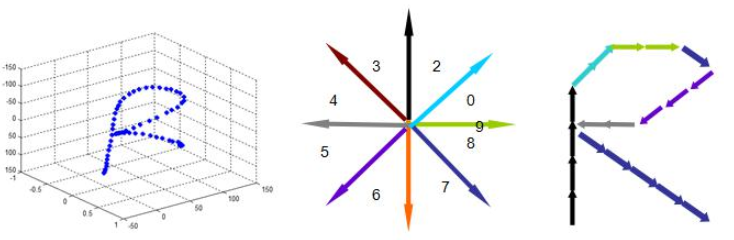
|  |  |  |
| --- | --- | --- |
|  |  | (1.2) |

## 2.7 Ekstraksi Fitur Dinamis dan *Hand Gesture*

Fitur dinamis merupakan fitur gerak atau fitur yang bersifat dinamis yang didapatkan dari deteksi Kinect 2.0 terhadap tangan. Fitur dinamis yang digunakan pada metode ini adalah fitur gerakan tangan atau disebut dengan *hand gesture*.

Sebuah fitur yang baik memainkan peran penting dalam pengenalan. Pada Tugas Akhir yang dibuat ini, terapat 40 *frame* yang akan diproses. Setiap *frame* akan menghasilkan koordinat gerak dinamis yang diproyeksikan ke dalam bidang X0Y yang merupakan prinsip dalam bidang. Kemudian dari hasil setiap frame tersebut, akan dicatat orientasi sudut mutlak yang dapat dilihat pada persamaan (1.0) (1.1) dan (1.2) [6].

|  |  |
| --- | --- |
|  | (2.1) |
|  | (2.2) |
|  | (2.3) |

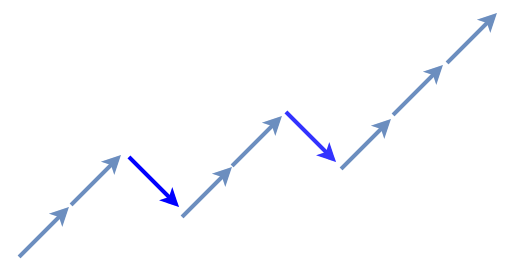


Gambar 2.4 Data Kuantitasi

Setelah mencapai urutan orientasi sudut, sudut dikuantisasi dengan kode 1 sampai 9 (9 merepresentasikan ) seperti Gambar 6. Berikut merupakan contoh data untuk ekstraksi fitur dinamis:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Frame | X | Y |  |  |  | Kuant |
| 1 | 1 | 5 | 0 | 0 | 0 | 0 |
| 2 | 2 | 6 | 1 | 1 | 23 | 1 |
| 3 | 4 | 9 | 2 | 3 | 28 | 1  Akan didapat kuantitasi 1,1,7,1,1,7,1,1,1 (dimulai dari baris ke-2) yang digunakan sebagai fitur dinamis untuk memperoleh pola gerakan tangan. |
| 4 | 6 | 4 | 2 | -5 | 326 | 7 |
| 5 | 7 | 5 | 1 | 1 | 23 | 1 |
| 6 | 9 | 8 | 2 | 3 | 28 | 1 |
| 7 | 7 | 13 | -2 | 5 | 350 | 7 |
| 8 | 10 | 17 | 3 | 4 | 27 | 1 |
| 9 | 12 | 18 | 2 | 1 | 13 | 1 |
| 10 | 13 | 19 | 1 | 1 | 23 | 1 |

Dari contoh data tersebut, akan menghasilkan pola gerakan tangan yang diharapkan menyerupai/sama dengan 10 pola gerakan yang menjadi batasan pada Tugas Akhir ini. Pada contoh data yang diberikan, ekstraksi fitur dinamis akan menghasilkan gerakan menyerupai gerakan Gelombang seperti terlihat pada Gambar 7.

 **=**  

Gambar 2.5 Ekstraksi fitur dinamis dari contoh data

*[Halaman ini sengaja dikosongkan]*

# BAB III ANALISIS DAN PERANCANGAN SISTEM

Bab ini membahas tahap analisis permasalahan dan perancangan dari sistem yang dibangun. Analisis permasalahan membahas permasalahan yang diangkat dalam pengerjaan tugas akhir. Analisis kebutuhan mencantumkan kebutuhan-kebutuhan yang diperlukan oleh sistem. Selanjutnya dibahas mengenai perancangan sistem yang dibuat.



## 3.1 Analisis Perangkat Lunak

Bahasa isyarat merupakan sarana komunikasi untuk penderita tunarungu. Walaupun agak sulit untuk mengartikan isyarat yang diberikan, hal tersebut telah membantu penderita tunarungu untuk berkomunikasi dengan sekitar. Namun masih banyak yang belum mengerti apa arti isyarat bagi yang diberi atau memberi isyarat.

Perangkat lunak ini bertujuan untuk membantu pengguna dalam mempelajari bahasa isyarat Indonesia yang sesuai dengan SIBI. Data isyarat tersebut didapatkan dari *training* data yang dilakukan oleh pengguna. Dengan menggunakan Microsoft Visual Studio dan Kinect 2.0, penulis mengekstraksi *skeleton* yang ditangkap oleh Kinect 2.0 kemudian dikalkulasi lalu menghasilkan sebuah fitur. Setelah selesai dibuat, fitur tersebut diolah menggunakan perangkat lunak WEKA untuk membentuk *Decision Tree* yang akan digunakan sebagai *classifier* dalam menentukan arti bahasa isyarat yang diberikan.

Untuk tahap *testing* data, setelah perangkat lunak mengekstraksi fitur yang didapatkan dari kalkulasi *skeleton* pengguna, fitur tersebut akan diklasifikasikan menggunakan *Decision Tree* yang sebelumnya sudah dibuat kemudian dimasukkan ke dalam perangkat lunak sebagai sebuah kondisi. Setelah seluruh tahap sudah selesai dilakukan, perangkat lunak akan memberikan keluaran berupa arti bahasa isyarat Indonesia yang dimaksud oleh pengguna.

### 3.1.1 Deskripsi Umum Perangkat Lunak

Tugas akhir yang dibangun ini adalah sebuah modul pengenalan bahasa isyarat Indonesia dengan menggunakan teknologi Kinect 2.0. Pengguna utama dalam perangkat lunak ini adalah semua orang yang ingin mempelajari bahasa isyarat Indonesia. Pengguna dapat menggunakan isyarat yang sudah ada di dalam perangkat lunak ataupun memberikan bahasa isyarat baru dengan mengacu pada Sistem Isyarat Bahasa Indonesia (SIBI).

### 3.1.2 Spesifikasi Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak yang akan dibuat ini melibatkan dua hal, yakni kebutuhan fungsional maupun kebutuhan non-fungsional. Dimana masing-masing berhubungan dengan keberhasilan dalam pembuatan tugas akhir ini.

#### 3.1.2.1 Kebutuhan Fungsional Perangkat Lunak

Dalam pengembangan perangkat lunak, terdapat beberapa kebutuhan fungsional yang mendukung jalannya perangkat lunak. Kebutuhan fungsional tersebut adalah sebagai berikut:

1. Mendeteksi *skeleton* pengguna

Perangkat lunak dapat mendeteksi pengguna yang sedang berada di depan Kinect 2.0.

1. Mengekstraksi fitur dinamis *skeleton*

Perangkat lunak dapat mendeteksi bagian-bagian dari *skeleton* pengguna yang akan diekstraksi menjadi fitur dinamis guna melakukan proses klasifikasi data.

1. Menerjemahkan Bahasa Isyarat

Perangkat lunak dapat menerjemahkan bahasa isyarat Indonesia yang dihasilkan melalui proses ekstraksi fitur dinamis *skeleton* pengguna.

#### 3.1.2.2 Kebutuhan Non-Fungsional

Disamping kebutuhan fungsional, terdapat juga beberapa kebutuhan non-fungsional dalam mendukung dan menambah performa perangkat lunak. Kebutuhan non-fungsional tersebut adalah sebagai berikut:

1. Penyesuaian Intensitas Cahaya

Intensitas cahaya merupakan salah satu unsur penting dalam penggunaan sensor di Kinect 2.0. Jika intensitas cahaya rendah, data *skeleton* yang diambil oleh Kinect 2.0 akan menjadi tidak stabil. Oleh karena itu, dalam menggunakan perangkat lunak ini, sebaiknya dilakukan di ruangan yang mempunyai intensitas cahaya yang cukup.

1. Posisi Kinect dengan Pengguna

Dalam mencapai hasil oleh data perangkat lunak (baik itu pengambilan data mapun uji coba data) yang sempurna, jarak optimal Kinect 2.0 dengan pengguna adalah antara 0,6 sampai 1,8 meter.

### 3.1.3 Identifikasi Pengguna

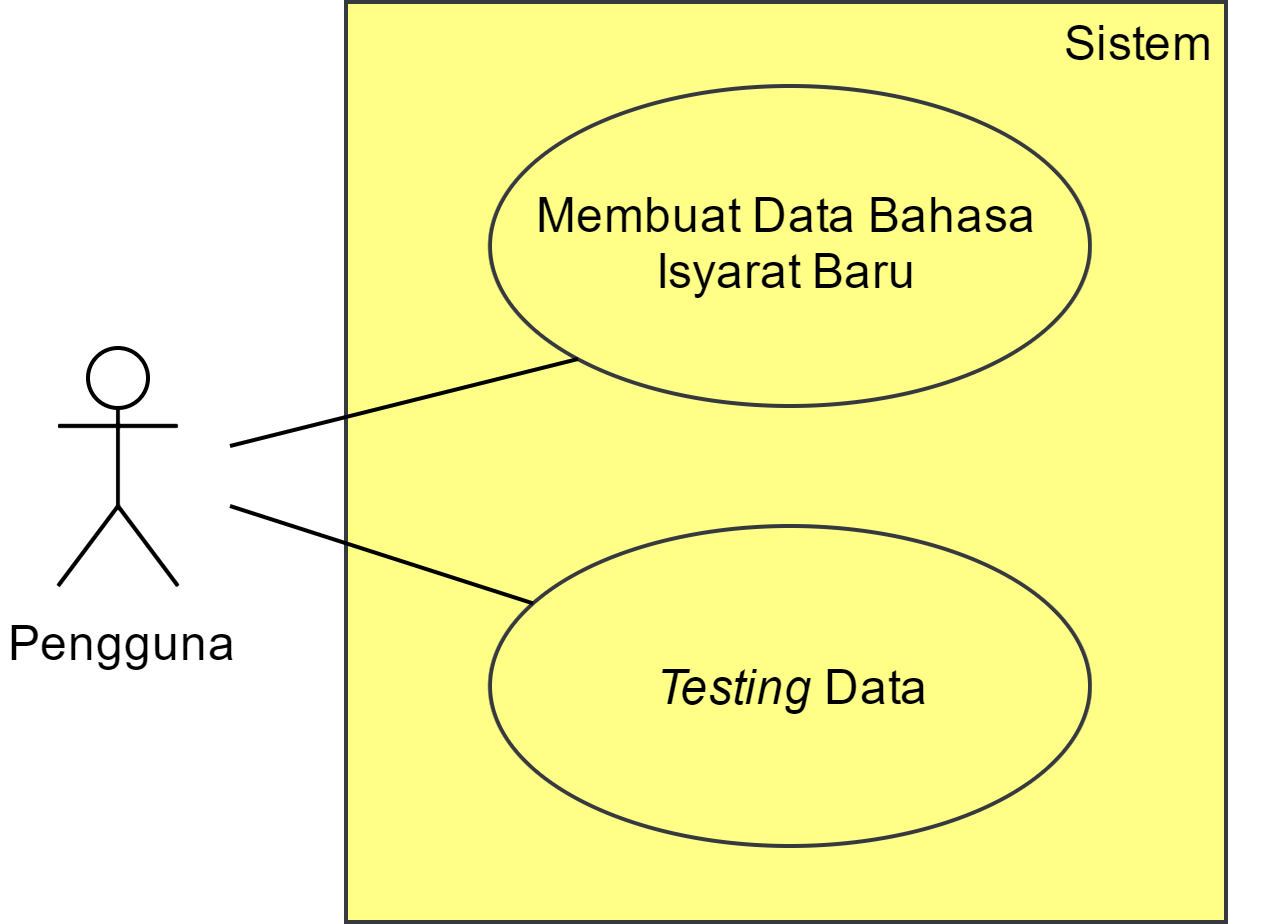
Dalam tugas akhir yang dibangun ini, pengguna yang akan terlibat dalam menjalankan perangkat lunak hanya satu orang saja, yaitu orang yang akan melakukan pengenalan bahasa isyarat Indonesia.

## 3.2 Perancangan Perangkat Lunak

Subbab ini membahas bagaimana rancangan dari tugas akhir ini. Hal yang dibahas meliputi model kasus penggunaan, definisi aktor, definisi kasus penggunaan, arsitektur umum sistem, rancangan antarmuka perangkat lunak, dan rancangan proses perangkat lunak.

### 3.2.1 Model Kasus Penggunaan

Dari hasil analisa deskripsi umum perangkat lunak dan spesifikasi kebutuhan perangkat lunak yang telah dijelaskan, maka model kasus penggunaan untuk perangkat lunak pengenalan bahasa isyarat dapat dilihat pada Gambar 3.1.



Gambar 3.1 Diagram Kasus Penggunaan Perangkat Lunak

### 3.2.2 Definisi Aktor

Aktor yang terdapat dalam sistem aplikasi ini terlihat pada Tabel 3.1.

Tabel 3.1 Definisi Kasus Penggunaan

|  |  |  |
| --- | --- | --- |
| No | Nama | Deskripsi |
| 1 | Pengguna | Merupakan aktor yang bertugas untuk menambahkan data *training* dan melakukan *testing* gerakan isyarat dinamis, seluruh fungsionalitas yang ada di dalam sistem dapat digunakan oleh pengguna. |

### 3.2.3 Definisi Kasus Penggunaan

Pada Gambar 3.1 telah dijelaskan bahwa aktor yang dalam hal ini disebut pengguna mempunyai dua kasus penggunaan, yakni membuat data bahasa isyarat Indonesia yang baru dan melakukan *testing* data. Detail mengenai kasus penggunaan tersebut dapat dilihat pada Tabel 3.2.

Tabel 3.2 Definisi Kasus Penggunaan

|  |  |  |  |
| --- | --- | --- | --- |
| No | Kode Kasus Penggunaan | Nama Kasus Penggunaan | Keterangan |
| 1 | UC-01 | Membuat Data Bahasa Isyarat Baru | Pengguna membuat data bahasa isyarat Indonesia yang baru |
| 2 | UC-02 | *Testing* Data | Pengguna melakukan *testing* data dengan melakukan gerakan bahasa isyarat Indonesia yang tersedia. |

#### 3.2.3.1 Kasus Penggunaan Membuat Data Bahasa Isyarat Baru

Spesifikasi kasus penggunaan membuat data bahasa isyarat baru dapat dilihat pada Tabel 3.3.

Tabel 3.3 Spesifikasi Kasus Penggunaan Membuat Data Bahasa Isyarat Baru

|  |  |
| --- | --- |
| Nama Kasus Penggunaan | Membuat Data Bahasa Isyarat Baru |
| Nomor | UC-01 |
| Deskripsi | Kasus penggunaan aktor untuk membuat data bahasa isyarat baru |
| Aktor | Pengguna |
| Kondisi Awal | Pengguna sudah menjalankan aplikasi dan perangkat Kinect 2.0 telah tersambung |
| Alur Normal | 1. Pengguna memasukan nama bahasa isyarat yang akan dibuat di dalam *textbox* perangkat lunak  2. Pengguna menekan tombol “Create File” di dalam perangkat lunak  3. Perangkat lunak menerima inputan dari Kinect 2.0 dan ketika *skeleton* pengguna ditemukan, perangkat lunak memberikan waktu 5 detik untuk pengguna mempersiapkan gerakan  A1. Kinect 2.0 tidak menemukan *skeleton* pengguna  4. Pengguna melakukan gerakan bahasa isyarat yang akan dibuat  5. Perangkat lunak mengekstrak data *skeleton* pengguna sebanyak 40 data/*frame* untuk dikalkulasi  A2. Kinect 2.0 kehilangan *skeleton* saat mengekstrak data  6. Perangkat lunak menyimpan hasil ekstraksi ke dalam sebuah *file* |
| Alur Alternatif | A1. Kinect 2.0 tidak menemukan *skeleton* pengguna  1. Sistem memberikan notifikasi bahwa *skeleton* pengguna tidak ditemukan  A2. Kinect 2.0 kehilangan *skeleton* saat mengekstrak data  1. Perangkat lunak menghentikan proses kalkulasi  2. Perangkat lunak memberikan notifikasi bahwa *skeleton* hilang |
| Kondisi Akhir | Perangkat lunak akan membuat data set baru |

#### 3.2.3.2 Kasus Penggunaan *Testing* Data

Spesifikasi kasus penggunaan *Testing* Data dapat dilihat pada Tabel 3.4.

Tabel 3.4 Spesifikasi Kasus Penggunaan *Training Dataset*

|  |  |
| --- | --- |
| Nama Kasus Penggunaan | *Testing* Data |
| Nomor | UC-02 |
| Deskripsi | Kasus penggunaan aktor untuk melakukan *testing* data isyarat bahasa Indonesia yang telah dibuat |
| Aktor | Pengguna |
| Kondisi Awal | Pengguna dalam keadaan menjalankan perangkat lunak dan sudah ada *classifier* data isyarat bahasa Indonesia di dalam perangkat lunak |
| Alur Normal | 1. Pengguna menekan tombol “Start Testing” di dalam perangkat lunak  2. Perangkat lunak menerima inputan dari Kinect 2.0 dan ketika *skeleton* pengguna ditemukan, perangkat lunak memberikan waktu 5 detik untuk pengguna mempersiapkan gerakan  A1. Kinect 2.0 tidak menemukan *skeleton* pengguna  3. Pengguna melakukan gerakan bahasa isyarat  4. Perangkat lunak mengekstrak data *skeleton* pengguna sebanyak 40 data/*frame* untuk dikalkulasi  5. Hasil ekstraksi data diklasifikasi menggunakan *Decision Tree* yang sudah dibuat di dalam perangkat lunak  6. Perangkat lunak menampilkan isyarat bahasa Indonesia hasil klasifikasi |
| Alur Alternatif | A1. Kinect 2.0 tidak menemukan *skeleton* pengguna  1. Sistem memberikan notifikasi bahwa *skeleton* pengguna tidak ditemukan |
| Kondisi Akhir | Perangkat lunak memberikan keluaran berupa bahasa isyarat yang dimaksud oleh pengguna |

### 3.2.4 Arsitektur Umum Sistem

Arsitektur umum pada perangkat lunak ini memiliki perangkat tambahan Kinect 2.0 sebagai perangkat masukan. Implementasi aplikasi dibuat menggunakan Microsoft Visual Studio. Arsitektur umum perangkat lunak yang akan dibuat dapat dilihat pada Gambar 3.2.



Gambar 3.2 Arsitektur Umum Sistem

### 3.2.5 Rancangan Antarmuka Perangkat Lunak

Rancangan antarmuka perangkat lunak diperlukan untuk memberikan gambaran umum kepada pengguna bagaimana sistem yang ada dalam perangkat lunak ini berinteraksi dengan pengguna. Selain itu, rancangan ini juga memberikan gambaran bagi pengguna apakah tampilan yang sudah disediakan oleh perangkat lunak mudah untuk dipahami dan digunakan, sehingga akan muncul kesan *user experience* yang baik dan mudah.

Rancangan antarmuka perangkat lunak ini hanya memiliki satu Windows dengan beberapa elemen didalamnya seperti *color view* dan *skeleton view* dari Kinect 2.0, dan kontrol–kontrol lainnya yang sekiranya dapat dipahami oleh pengguna. Rancangan antarmuka perangkat lunak dapat dilihat pada Gambar 3.3.



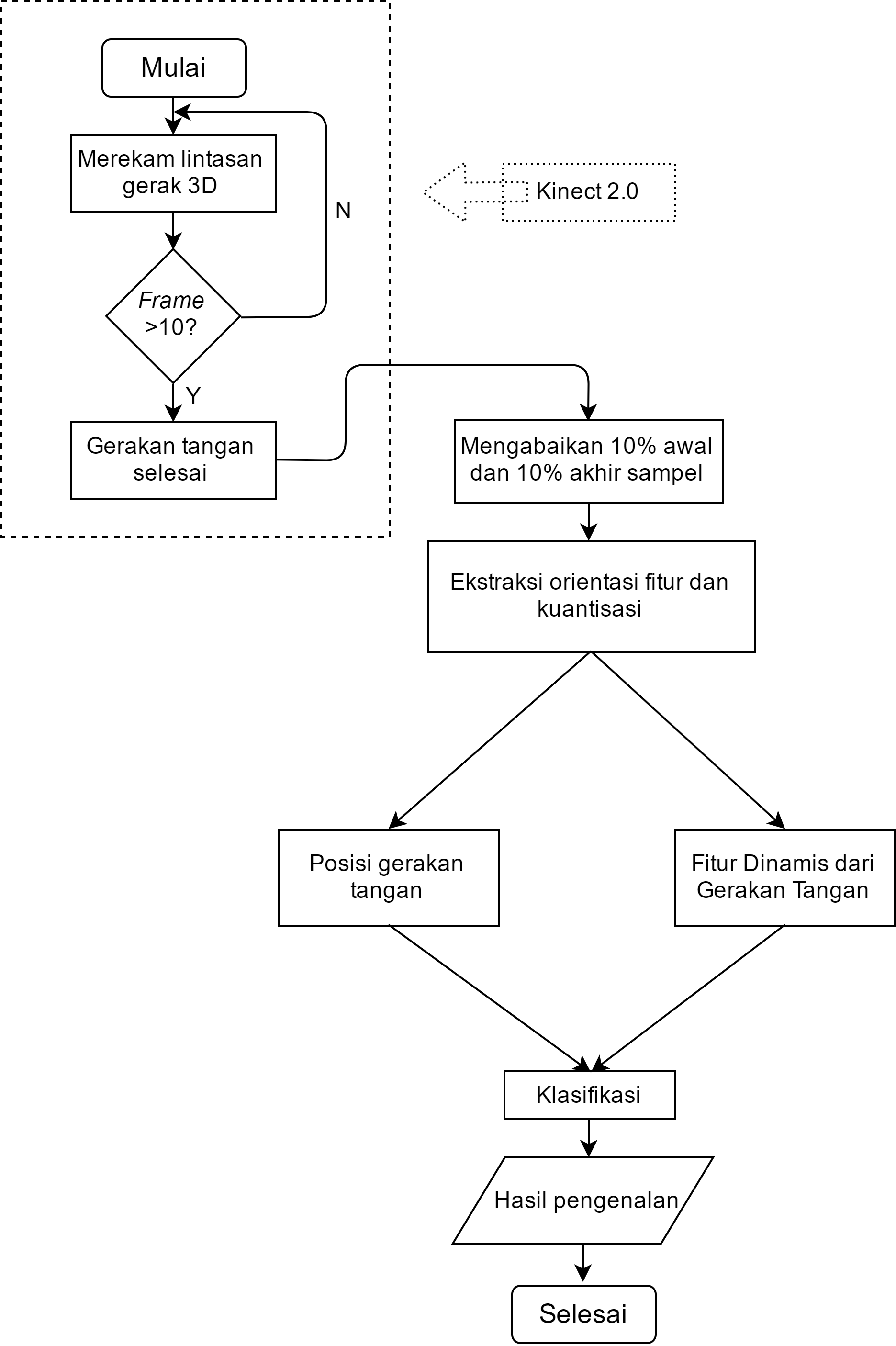
Gambar 3.3 Rancangan Antarmuka Perangkat Lunak

### 3.2.6 Rancangan Proses Perangkat Lunak

Pada rancangan proses perangkat lunak akan dijelaskan mengenai proses yang terjadi dalam sistem untuk memenuhi fungsionalitas yang ada pada perangkat lunak. Proses ini penting agar perangkat lunak dapat berjalan secara baik dan benar.

#### 3.2.6.1 Rancangan Proses Ekstraksi Fitur dan Menentukan Posisi Gerakan Tangan

Proses ekstraksi fitur dinamis dari *skeleton* pengguna merupakan hal yang terpenting dalam pembangunan perangkat lunak ini. Ketika pengguna membuat data bahasa isyarat baru, perangkat lunak akan memberikan waktu 5 detik untuk pengguna mempersiapkan gerakan. Setelah itu perangkat lunak akan mengambil total 40 data/*frame* dan akan dikalkulasi kemudian disimpan ke dalam file berekstensi .csv. Rancangan proses ekstraksi fitur dinamis dan menentukan posisi gerakan tangan dapat dilihat pada Gambar 3.4.



Gambar 3.4 Diagram Alir Proses Ekstraksi Fitur Dinamis dan Menentukan Posisi Gerakan Tangan

Dalam melakukan ekstraksi fitur dinamis dan menentukan posisi gerakan tangan baik untuk membuat data bahasa isyarat baru maupun melakukan *testing* data, penulis mengambil sebanyak empat *skeleton joints* yang dapat diidentifikasi oleh Kinect 2.0. Empat *skeleton joints* tersebut adalah sebagai berikut:

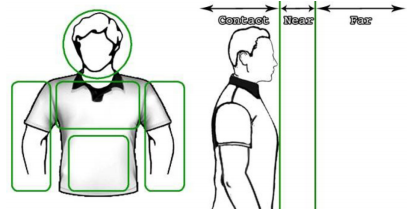
1. Telapak tangan kiri (HandLeft)
2. Telapak tangan kanan (HandRight)
3. Leher (Neck)
4. Bagian tengah tulang belakang (SpineMid)

*Skeleton* telapak tangan kiri dan telapak tangan kanan kemudian diolah menjadi fitur dinamis berdasarkan persamaan (2.1), (2.2), (2.3) dengan mengabaikan 10% data/*frame* awal dan hanya data/*frame* genap yang disimpan dengan harapan agar mendapat fitur dinamis yang lebih akurat. Pembagian atribut data fitur dinamis tersebut dapat dilihat pada Tabel 3.6.

Tabel 3.6 Pembagian Atribut Data Fitur Dinamis

|  |  |
| --- | --- |
| *Skeleton* | Atribut data |
| HandLeft | Atribut ke 0 – 17 |
| HandRight | Atribut ke 18 – 35 |

Sedangkan *skeleton* leher (Neck) dan bagian tengah tulang belakang (SpineMid), akan digunakan sebagai acuan untuk menentukan posisi gerakan yang dibuat oleh pengguna dalam proses pengambilan data. Posisi gerakan tangan tersebut dapat dilihat pada Gambar 3.5.



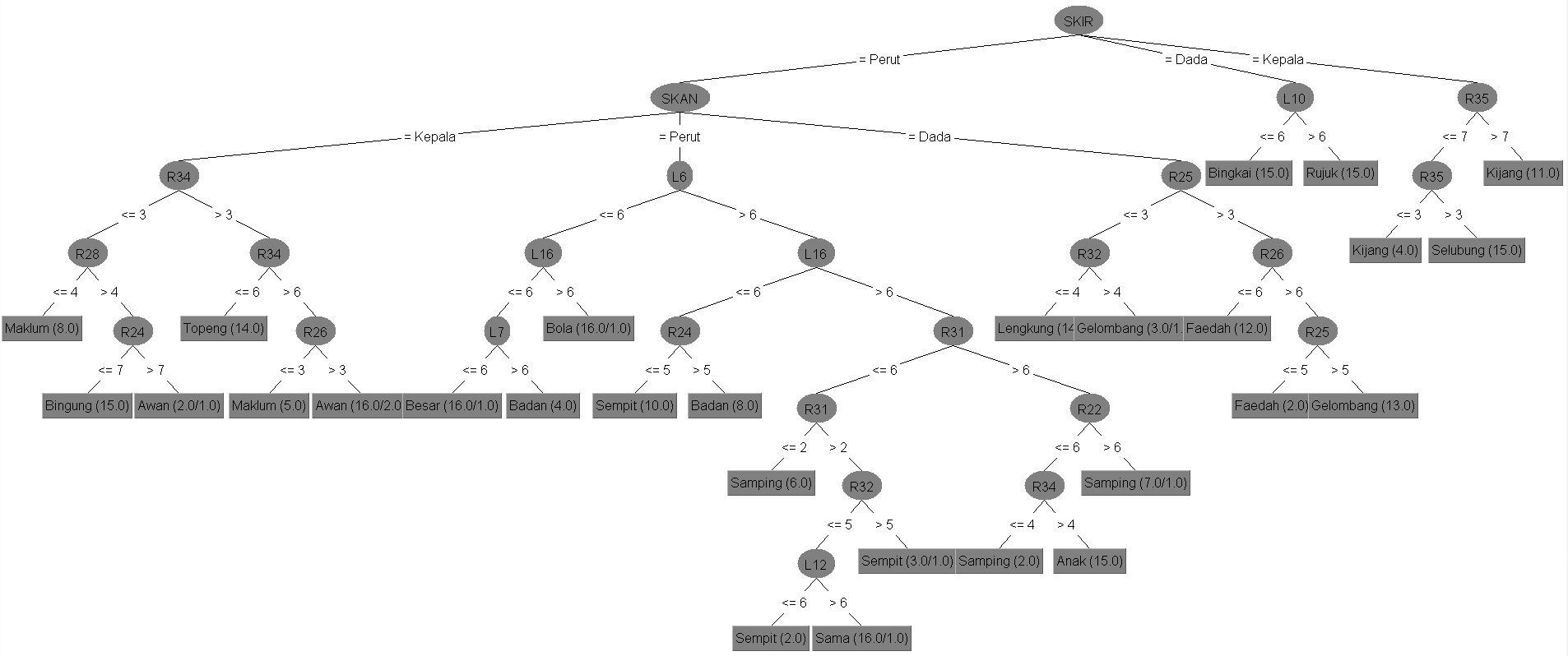
**2**

**1**

Gambar 3.5 Posisi gerakan tangan pada tubuh mengacu pada *skeleton* leher (Neck) dan bagian tengah tulang belakang (SpineMid)

#### 3.2.6.4 Rancangan Proses *Training*

Proses *training* data dilakukan pada perangkat lunak WEKA menggunakan algoritma *Decision Tree* C4.5 (J48) terhadap data *training* yang sudah dibuat. Hasil dari metode tersebut dapat dilihat pada Gambar 3.7. Dari hasil tersebut kemudian dimasukan ke dalam program perangkat lunak yang dibangun untuk mengklasifikasi ketika pengguna melakukan *testing* data.



Gambar 3.7 Hasil Metode *Decision Tree*

#### 3.2.6.5 Rancangan Proses *Testing*

Proses *testing* dilakukan dengan membuat data baru yang dibentuk dari proses ekstraksi fitur. Data tersebut tidak penulis masukan ke dalam sebuah dokumen, melainkan diklasifikasi secara langsung di dalam perangkat lunak menggunakan *Decision Tree* yang sudah dibuat sebelumnya oleh pengguna