PERANCANGAN DAN IMPLEMENTASI APLIKASI PENERJEMAH BAHASA ISYARAT MENJADI SUARA BERBASIS KINECT MENGGUNAKAN METODE *HIDDEN MARKOV MODEL*

DESIGN AND IMPLEMENTATION OF SIGN LANGUAGE TO SPEECH TRANSLATOR APPLICATION BASED ON KINECT USING HIDDEN MARKOV MODEL METHOD

Agustinus[1], Asep Mulyana, ST., MT.[2], Andrew Brian O., ST., MT.[3]

[123]Prodi S1 Sistem Komputer, Fakultas Teknik Elektro, Universitas Telkom University
[1]agustinusandreas1234@gmail.com, [2]asepm267@gmail.com, [3]abosmond@telkomuniversity.ac.id

Abstrak

Suatu isyarat dapat dikenali melalui gerakan tubuh dan gerakan mulut seseorang. Dengan mengenali ciri-ciri khusus pada masing-masing isyarat, maka dapat diterjemahkan. Pada tugas akhir ini dilakukan pengenalan sejumlah kata dalam bahasa isyarat, agar orang-orang dengan kemampuan berbicara secara verbal lebih mudah menerjemahkan isyarat seorang tunarungu.

Masukan secara realtime yang digunakan berupa data gambar (image) pose bahasa isyarat (berdasarkan Kamus Umum Bahasa Isyarat Indonesia) menggunakan perangkat sensor kinect. Data tersebut kemudian diolah sehingga mendapatkan data skeleton. Dari data skeleton kemudian didapatkan skeleton joint. Dalam hal ini menggunakan 6 skeleton joint untuk dijadikan data karakteristik dari setiap isyarat. Berikutnya 6 skeleton joint akan dimodelkan dengan Hidden Markov Model (HMM), dan dilakukan pelatihan sehingga dihasilkan sebuah basis data untuk seluruh HMM. Keluaran dari sistem ini berupa suara yang merupakan terjemahan dari isyarat pada masukan.

Hasil pengujian pada tugas akhir ini mendapat akurasi rata – rata 90%, trendline akurasi untuk berbagai jumlah kata mengalami penurunan jika ditambah isyarat baru ke database, dan waktu komputasi sistem rata-rata 22.8µs.

Kata Kunci: Bahasa Isyarat, *Kinect, Hidden Markov Model*

Abstract

A sign language can be recognized by body movement and mouth movement. By identifying the specific characteristics of each sign language, so can be translated. In this final project there is some sign languages to be recognized, so common people can translate a sign language from the person who has a deaf problem.

Realtime input is image data of sign language pose (based on General Dictionary of Indonesian Sign Language) that use kinect sensor device. Then the data is processed to obtain the skeleton data. After that is obtained many joints of skeleton from the skeleton data. In this case use 6 joints of skeleton as the characteristic of each sign language. And then 6 joints of skeleton will be modeled by Hidden Markov Model (HMM), and go to training process for produce the database. Output of the system is sound of the input sign language translation.

The test result in this final project get the accuracy of system average 90 %, accuracy of trendline in some word decrease if database is added, and time of computation system average is 22.8 µs.

Keyword : Sign Language, Kinect, Hidden Markov Model

1. Pendahuluan

Bahasa merupakan alat komunikasi yang sangat penting bagi kehidupan manusia, karena pada bahasa terdapat proses pertukaran informasi yang dapat menambah pemahaman manusia. Berdasarkan data dari World Health Organization (WHO) bulan Februari 2014 menyebutkan bahwa Lebih dari 5% dari populasi dunia yaitu sekitar 360 juta orang yang menyandang tunarungu (328 juta orang dewasa dan 32 juta anak-anak) [9]. Semakin banyaknya penyandang tunarungu yang terlahir dan seorang tunarungu tidak dapat berbicara secara verbal hanya bisa menggunakan bahasa isyarat, sehingga orang normal tidak dapat berkomunikasi dengan baik dengan penyandang tunarungu. Maka diperlukan sebuah aplikasi untuk dapat menterjemahkan maksud dari bahasa isyarat tersebut.

Microsoft pada tahun 2010 meluncurkan sebuah alat sensor gerakan tubuh yang bernama Kinect. Pada awalnya alat Kinect ini dipergunakan untuk game dengan konsep Natural User Interface, yaitu mengendalikan permainan tanpa menggunakan controller seperti biasa. Namun sekarang ini pemanfaatannya semakin meluas.

Kemampuan dasarnya seperti mendapatkan depth image bisa berkembangkan sehingga mampu mengenali suatu pose tubuh (pose recognition). Namun pose recognition bukanlah fitur bawaan kinect, sehingga diperlukan algoritma pengenalan pola yang bisa mendapat nilai optimum untuk akurasi pengenalan seperti metode Hidden Markov Model.

Untuk mempermudah komunikasi dengan menggunakan bahasa isyarat, maka dibangun sebuah aplikasi penerjemah bahasa isyarat menjadi suara dengan menggunakan metode Hidden Markov Model agar orang normal dapat mengerti maksud dari bahasa isyarat penyandang tunarungu.

2. Dasar Teori

2.1 Bahasa Isyarat Indonesia [2]

Bahasa Isyarat merupakan bahasa yang sering dipakai oleh kaum tunarungu. Bahasa isyarat adalah bahasa yang tidak menggunakan bunyi ucapan manusia atau tulisan dalam sistem perlambangannya [1]. Kaum tunarungu menggunakan bahasa ini mengombinasikan bentuk tangan, orientasi dan gerak tangan, lengan, dan tubuh, serta ekspresi wajah untuk mengungkapkan pikiran mereka.

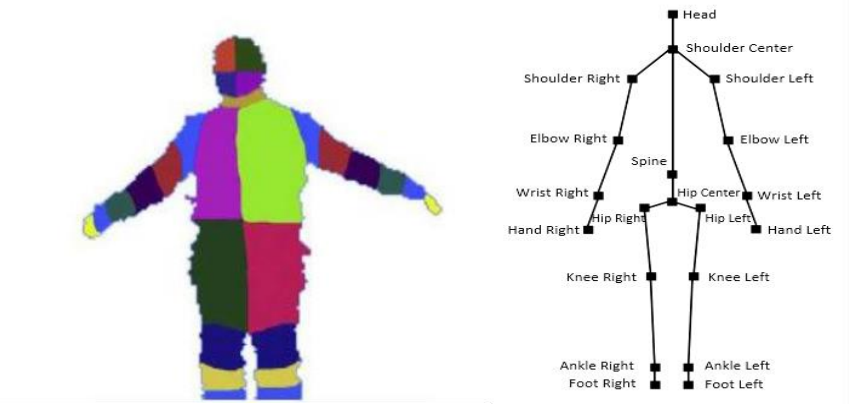
Bahasa Isyarat Indonesia adalah isyarat-isyarat kata yang pada mulanya diambil dari isyarat-isyarat yang disampaikan anak tunarungu yang bisa diterima sebagai kata atau kosakata dalam Bahasa Indonesia termasuk American Sign Language (ASL) atau Bahasa Isyarat Amerika yang diIndonesiakan [2]. Berikut contoh gambar isyarat yang diambil dari Kamus Umum Bahasa Isyarat Indonesia pada gambar 2.1

Gambar 2. 1 Contoh isyarat “terima kasih” (kiri),“cantik: (tengah),“saya” (kanan)

2.2 Kinect [5][6][10]

Skeletal Tracking memungkinakan *Kinect* untuk mengenali *user* dan mengikuti pergerakannnya. Dengan menggunakan kamera inframerah(IR), *Kinect* dapat mengenali sampai dengan enam user dalam jangkauan . Dari jumlah tersebut, dua user dapat dikenali hingga detail. Pemanfaatan *Skeletal Tracking* pada suatu aplikasi dapat memberikan posisi sendi (skeleton joint) dari user yang dikenali dan mengikuti pergerakannya dari waktu ke waktu. *Skeletal Tracking* dioptimalkan untuk mengenali *user* yang berdiri ataupun duduk, dan menghadap Kinect. Dibawah ini adalah gambar yang user yang dikenali.

Kinect SDK menyediakan API(*Application Programming Interface*) untuk memudahkan mengakses seluruh titik sendi. Duapuluh titik sendi yang dapat diakses dan diidentifikasi sesuai dengan nama sendinya. Gambar 2.2 adalah gambar yang merepresentasikan seluruh *skeleton user* yang menghadap *kinect* membentuk dua puluh titik sendi.



Gambar 2. 2 (a) Pemetaan Badan Pemain (kiri) [5], (b) *Skeleton user* (kanan)

2.3 *Hidden Markov Model* [3][4]

Hidden Markov Model merupakan sebuah model statistik dari suatu proses Markov dengan *state* tersembunyi (*hidden*), *state* hanya bisa diamati secara tidak langsung melalui barisan pengamatan. Pada model Markov biasa, *state*-nya bisa langsung diamati sehingga peluang transisi antar *state* menjadi satu-satunya parameter. Pada HMM *state*-nya tidak dapat diamati secara langsung, dimana setiap *state* memiliki distribusi

peluang output yang mungkin muncul sebagai suatu set proses stokastik yang akan membentuk suatu deretan observasi.

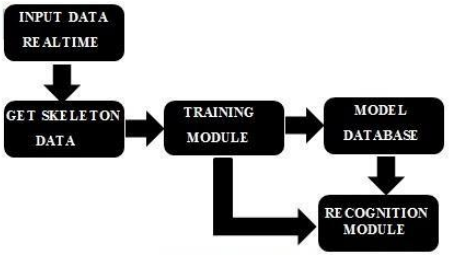
2.3.1 Elemen-elemen dan Tipe HMM

HMM terdiri atas elemen-elemen dasar, yaitu: [3]

a. Banyaknya *state* dalam model (N)
b. Banyaknya simbol pengamatan yang berbeda pada setiap *state* (M)
c. Distribusi probabilitas tansisi antar *state* (A)
   $A = \{aij\}$ menyatakan distribusi probabilitas transisi dari *state* ke-i menuju *state* ke-j
d. Distribusi probabilitas simbol pengamatan (B)
   $B = \{bj\ (k)\}$ adalah distribusi probabilitas simbol observasi ke-k pada *state* ke-j, yang dinyatakan dalam matrik BNxM.
e. Distribusi probabilitas peluang state awal (π)
   $\pi = \{\pi i\}$ menyatakan distribusi probabilitas inisialisasi pada *state* ke-i

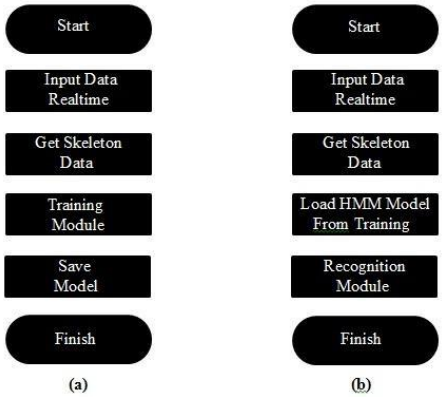
3. Perancangan Sistem

Sistem dirancang dengan tujuan mengenali bahasa isyarat dari input pose bahasa isyarat secara *realtime* untuk diterjemahkan menjadi suara. Pada sistem ini terdapat 4 buah modul dan 1 buah database seperti gambar 3.1.



Gambar 3.1 Blok Diagram Sistem [4]

4 buah modul tersebut adalah Input data realtime, Get skeleton data, Training module, dan Recognition module. Modul-modul tersebut dijelaskan di sub bab 3.1 sampai 3.5. Untuk 1 buah database terdapat model database yang terdiri dari 30 pose isyarat yang telah dimodelkan. Dari blok diagram sistem terbagi menjadi 2 proses yaitu, proses *training* dan proses *testing* seperti gambar 3.2.

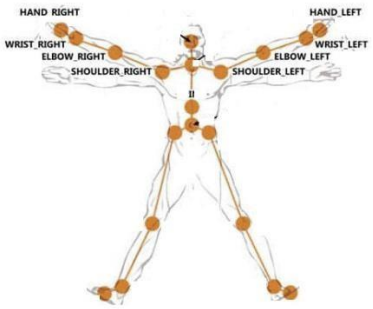


Gambar 3.2 (a)*Flowchart* Proses *Training* (b)*Flowchart Proses* Testing

Pada proses *training* terdapat proses pembentukan nilai pada setiap pose isyarat yang dijadikan *database* kemudian disimpan model masing-masing isyarat ke dalam *database*. Pada proses testing adalah percobaan pengenalan dari masukkan pose tubuh manusia yang akan dikenali oleh aplikasi dan menghasilkan keluaran teks keputusan isyarat yang dikenali juga suara.

3.1 Get Skeleton Data

Kerangka user bisa didapatkan dengan mengaktifkan fitur *depth stream* dan *skeletal tracking* pada sensor kinect yang disediakan oleh Kinect SDK. Dengan mengaktifkan kedua fitur tersebut, aplikasi dapat mengenali objek manusia hingga dua orang dengan masing-masing terdapat 20 titik sendi (*skeleton joint*) seperti gambar 3.3.. Setiap posisi titik sendi mengandung data 3D dengan koordinat x, y, z.



Gambar 3. 1 Skeleton Data

Aplikasi yang dirancang hanya menggunakan satu *user* dengan 8 titik sendi dengan 3 koodinat x, y, z. Titik sendi yang digunakan adalah tangan kanan (*hand right*), tangan kiri (*hand left*), pergelangan tangan kanan (*wrist right*), pergelangan tangan kiri (*wrist left*), siku kanan (*elbow right*), siku kiri (*elbow left*), bahu kanan (*shoulder right*), dan bahu kiri (*shoulder left*). Bahu kanan dan bahu kiri tidak akan dijadikan data pembanding untuk pengenalan gerakan, tetapi digunakan untuk referensi objek karena sifatnya yang kaku. Jadi data pembandingnya hanya ada 6 titik sendi.

3.2 Normalisasi Data

User yang melakukan gerakan bahasa isyarat pada dasarnya bebas melakukannya dimana saja selama dalam jangkauan *Kinect*. Karena posisi *user* tidak pasti maka perlu melakukan normalisasi dari data *skeleton user* yang diperoleh untuk mengembalikan posisi *skeleton* ke tengah sumbu kartesian tiga dimensi dan menyamakan skala ukuran *skeleton*. Untuk mengembalikan posisi *skeleton user* ke posisi normal , maka dilakukan translasi terhadap titik koordinat sendinya dengan persamaan berikut:

$$x_{baru} = x_{lama} - \left(\frac{x_{bahu\ kiri} - x_{bahu\ kanan}}{2}\right) \quad y_{baru} = y_{lama} - \left(\frac{y_{bahu\ kiri} - y_{bahu\ kanan}}{2}\right) \quad (3.1)$$

$$z_{baru} = z_{lama} - \left(\frac{z_{bahu\ kiri} - z_{bahu\ kanan}}{2}\right)$$

Setelah melakukan translasi dilanjutkan dengan dilatasi koordinatnya. Dilatasi perlu dilakukan agar ukuran skeleton yang berbeda-beda menjadi tidak masalah. Persamaan dilatasi adalah sebagai berikut :

$$x_{baru} = \frac{x_{lama}}{\sqrt{(x_{bahu\ kiri} - x_{bahu\ kanan})^2 - (y_{bahu\ kiri} - y_{bahu\ kanan})^2 - (z_{bahu\ kiri} - z_{bahu\ kanan})^2}} \quad (3.2)$$

$$y_{baru} = \frac{y_{lama}}{\sqrt{(x_{bahu\ kiri} - x_{bahu\ kanan})^2 - (y_{bahu\ kiri} - y_{bahu\ kanan})^2 - (z_{bahu\ kiri} - z_{bahu\ kanan})^2}} \quad (3.3)$$

$$z_{baru} = \frac{z_{lama}}{\sqrt{(x_{bahu\ kiri} - x_{bahu\ kanan})^2 - (y_{bahu\ kiri} - y_{bahu\ kanan})^2 - (z_{bahu\ kiri} - z_{bahu\ kanan})^2}} \quad (3.4)$$

3.3 Training Module [4]

Pembuatan model untuk mewakili suatu gerakan isyarat, di mana digunakan algoritma reestimasi untuk memecahkannya yang dilakukan pada *training module*. Untuk *training process* dilakukan dengan 4 proses. Pertama mendapat data *input* kemudian diolah sampai mendapat *skeleton joint* dan setiap *joint* dideteksi berada pada posisi state mana saja. Proses kedua melihat data matriks A yang telah dibentuk. Setelah itu, ketiga dari data *input* yang berupa *skeleton joint* yang telah ditentukan berada di state mana, kemudian dapat dikenali model/deret observasi yang dihasilkan. Yang terakhir, model dan hasil nilai yang didapat kemudian disimpan ke dalam *database* untuk diolah di process *recognition module.*

3.4 Inisialisasi elemen-elemen HMM

Pemodelan HMM terdiri dari 3 matriks probabilitas, yaitu matriks transisi antar state (A), matriks probabilitas pengamatan suatu state (B), dan matriks probabilitas awal state (π). Pada tugas akhir ini jenis HMM yang digunakan merupakan diskrit *ergodic*, dimana parameter-parameter HMM seperti matriks A, B, dan π dibangkitkan secara *random* dengan nilai yang dinormalisasi ke satu. Nilai-nilai matriks A, B, dan π tersebut kemudian akan dilakukan re-estimasi melalui proses pelatihan untuk mendapatkan nilai parameter yang optimal.

Sedangkan untuk elemen-elemen pembentuk HMM yang lain diambil berdasarkan banyaknya state dan jumlah observasi. Banyaknya *state* (N) diambil dari banyaknya *state* pada arsitektur HMM. Pada HMM *state*- nya bersifat *hidden*, dimana pada tugas akhir ini *state* tersembunyinya adalah jenis gerakan isyarat itu sendiri, sedangkan bagian yang akan diobservasi adalah ciri dari gerakan isyarat tersebut.

Matriks A yang digunakan untuk tugas akhir ini sebagai berikut :

$$\begin{bmatrix} 0.4 & 0.25 & 0.05 & 0.1 & 0.15 & 0.05 \\ 0.25 & 0.4 & 0.1 & 0.05 & 0.05 & 0.15 \\ 0.15 & 0.05 & 0.4 & 0.1 & 0.25 & 0.05 \\ 0.05 & 0.15 & 0.1 & 0.4 & 0.05 & 0.25 \\ 0.15 & 0.1 & 0.25 & 0.05 & 0.4 & 0.05 \\ 0.1 & 0.15 & 0.05 & 0.25 & 0.05 & 0.4 \end{bmatrix}$$

Cara menentukan nilai matriks A adalah berdasarkan persamaan notasi :

$$a_{ij} = P[q_{t+1} = S_j | q_t = S_i], \quad 1 \le i, j \le N. \tag{3.5}$$

Nilai-nilai di dalam matriks didapat dengan random terbentuk dari state yang ada dan dinormalisasi ke 1. Nilai 0.4 merupakan nilai yang ada pada posisi baris 1 dan kolom 1, baris 2 kolom 2, sampe baris 6 kolom 6. Nilai penjumlahan di setiap baris di normalisasi menjadi 1, dan nilai 0.4 adalah nilai tertinggi di setiap barisnya. Kemudian nilai tertinggi kedua adalah 0.25 itu ditempatkan pada posisi kolom yg berbeda di setiap baris matriksnya. Nilai 0.25 didapat karena berdasarkan nilai transisi selanjutnya dilihat dari state-state yang telah dibuat. Dan nilai 0.05, 0.1, 0.15, 0.05 berikutnya didapat berdasarkan nilai transisi selanjutnya juga.

Cara mendapatkan nilai B adalah berdasarkan persamaan notasi :

$$b_j(k) = P[v_k \text{ at } t | q_t = S_j], \quad 1 \le j \le N \quad 1 \le k \le M. \tag{3.6}$$

B berupa deret observasi yang merupakan model dari setiap bahasa isyarat.

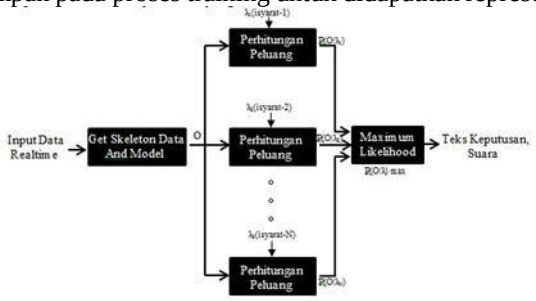
Cara mendapatkan nilai π adalah berdasarkan persamaan notasi :

$$\pi_i = P[q_1 = S_i], \quad 1 \le i \le N. \tag{3.7}$$

Π merupakan nilai initial state dan untuk kasus ini nilainya adalah 1.

3.5 Recognition Module

Masalah kedua pada aplikasi HMM adalah menghitung peluang maksimum suatu deretan observasi yang dibangkitkan oleh model suatu isyarat tertentu yang dilakukan pada *recognition module* (modul pengenalan). Proses pengenalan bahasa isyarat dilakukan pada tiap data video dengan menghitung *likelihood* dari data testing yang akan dikenali terhadap semua model data bahasa isyarat yang telah di-training sebelumnya. Keluaran proses ini memberikan indeks dari model bahasa isyarat referensi yang mempunyai *likelihood* yang paling besar (*maximum likelihood*). Indeks tersebut kemudian dicari dalam database dengan indeks database yang sudah disimpan pada proses training untuk didapatkan representasi teks dan *file* suara yang sesuai seperti gambar 3.5.



Gambar 3.5 Blok diagram sistem pengenalan

3.6 Performansi Sistem

Penilaian performansi sistem dilakukan dengan melihat tingkat akurasi sistem serta waktu yang dibutuhkan sistem dalam melakukan suatu fungsi. Tingkat akurasi menunjukkan persentasi jumlah isyarat dari data uji yang dapat dikenali dengan benar terhadap total data isyarat yang diujikan.

Perhitungan waktu dilakukan untuk menghitung lamanya suatu fungsi dilakukan. Ada 4 fungsi yang dihitung waktunya pada sistem ini, fungsi kuantisasi vektor *training*, fungsi *training* HMM, fungsi kuantisasi vektor *testing*, dan fungsi *training* sistem.

$$\text{Akurasi} = \frac{\text{Jumlah isyarat yang dikenali benar}}{\text{Total data isyarat yang diujikan}} \times 100\ \% \tag{3.8}$$

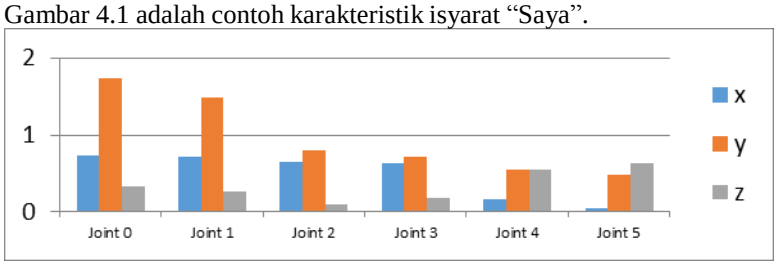
4. Implementasi Dan Pengujian Sistem

4.1 Implementasi

Database berupa data 1 frame yang terdiri dari kumpulan data joint skeleton yang telah direkam menggunakan Kinect Xbox 360 dan disimpan di dalam file .xml. 30 variasi isyarat yang dipraktekan yaitu

Ada	Akur	Ampun	Bahu	Bantal	Baring	Cinta	Dia	Horizontal	Hormat
Jendral	Koper	Kuat	Kubah	Malu	Mau	Pancasila	Perut	Pinggul	Rambut
Saya	Sebal	Topi	Tugu	Tuli	Akhir	Azan	Badak	Golf	Salib

Gambar 4.1 adalah contoh karakteristik isyarat "Saya".

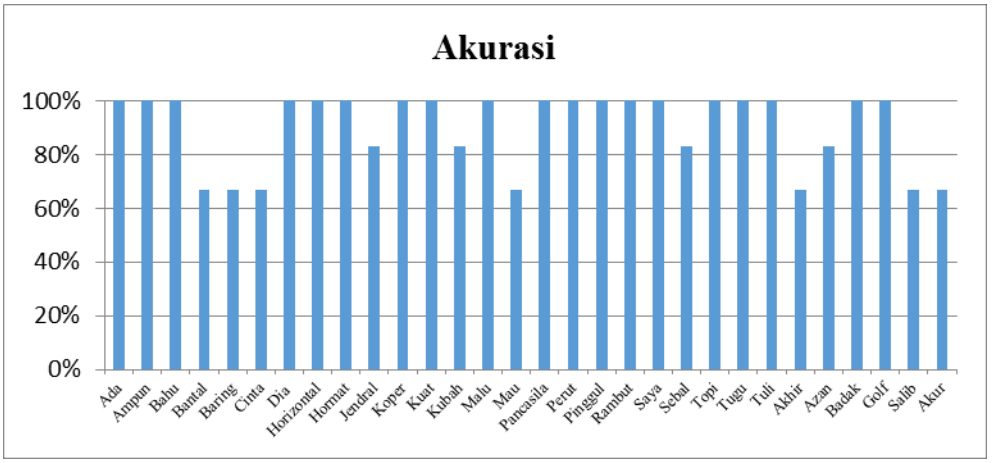


Gambar 4.1 Karakteristik Pose Isyarat "Saya"

Gambar 4.1 merupakan karakteristik pose iyarat "Saya" yang telah tersimpan di file .xml. Setiap pose isyarat terdiri dari 6 joint skeleton, dimana masing-masing joint memiliki nilai posisi x, y, dan z. Nilai x, y, dan z merupakan data *float* yang nilainya ada yang negatif dan positif. Maka dari itu untuk yang bernilai negatif akan dimutlakkan sehingga semua nilai menjadi positif.

Joint 0 = *HandLeft* Joint 1 = *WristLeft* Joint 2 = *ElbowLeft* Joint 3 = *ElbowRight* Joint 4 = *WristRight* Joint 5 = *HandRight*

4.2 Pengujian Akurasi Sistem



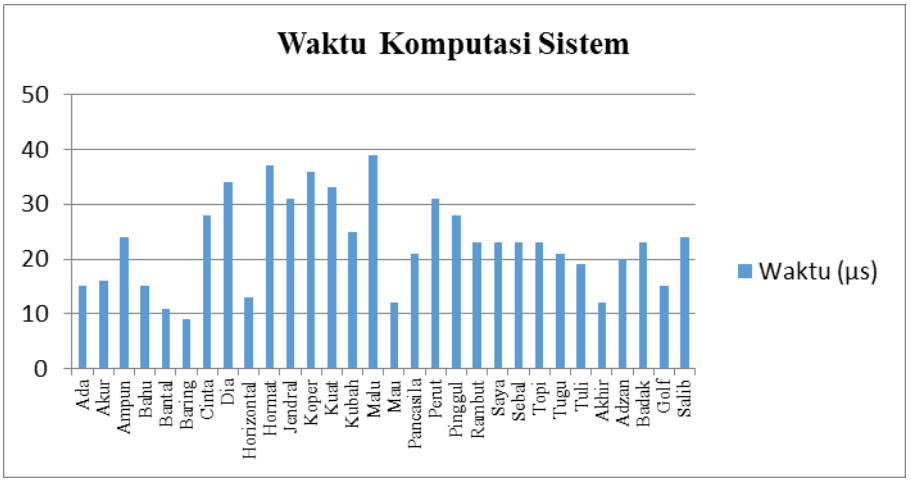
Gambar 4.2 Grafik Akurasi Sistem

Terlihat bahwa terdapat variasi dalam akurasi setiap bahasa isyarat. Bahasa isyarat yang memiliki akurasi tinggi itu karena dapat dikenali dengan mudah dan bahasa isyarat yang memiliki akurasi rendah itu karena sulit dikenali disebabkan oleh overlap(tumpukan tangan kanan dan kiri di posisi yang sama) sehingga sulit untuk mengenali dengan baik.

Gambar 4.3 Trendline akurasi untuk berbagai jumlah kata

Untuk akurasi sistem keseluruhan dari 180 kali percobaan dengan masing-masing tiap isyarat 6 kali, maka didapatkan akurasi rata-rata sebesar 90%. Trendline akurasi untuk berbagai jumlah kata mengalami penurunan jika ditambah isyarat baru ke database.

4.3 Pengujian Waktu Komputasi Sistem



Gambar 4.4 Waktu Komputasi Sistem

Waktu komputasi sistem menghasilkan nilai rata-rata untuk pengenalan yaitu 22.8 µs.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil penelitian dalam Tugas Akhir dapat disimpulkan bahwa :

1. Metode Hidden Markov Model berhasil diterapkan pada aplikasi penerjemah bahasa isyarat menjadi suara menggunakan sensor *Kinect* secara *realtime* dengan akurasi 90%.
2. Waktu komputasi rata-rata untuk memproses pengenalan yaitu 22,8 µs
3. Berdasarkan hasil pengujian beta menghasilkan :
   - presentase bermanfaatnya aplikasi = 87 %
   - presentase kemudahan mencocokkan pose tubuh = 73 %
   - presentase respon sistem = 84 %

5.2 Saran

Untuk pengenbangan tugas akhir ini dimasa yang akan datang, ada beberapa saran yang perlu diperhatikan yaitu sebagai berikut:

1. Mengembangkan aplikasi penerjemah bahasa isyarat seperti pengolahan kalimat isyarat.
2. Mengembangkan aplikasi penerjemah bahasa isyarat hingga ketelitian sampai jari tangan

Daftar Pustaka

[1] Pusat Bahasa. (1988). *Kamus Besar Bahasa Indonesia.* Jakarta: Balai Pustaka.

[2] Imas A. R. Gunawan 1996*Kamus Umum Bahasa Isyarat Indonesia*JakartaLembaga Komunikasi Total Indonesia

[3] Rabiner, L. R. 1989, " A Tutorial on Hidden Markov Models and Selected Application in Speech Recognition", IEEE Journal, Vol. 77. No. 2. pp. 257-286.

[4] Adisti, Ixora. 2010. "Perancangan dan Implementasi Penerjemah Bahasa Isyarat dari Video Menjadi Suara Menggunakan Ekstraksi Ciri dan Hidden Markov Model". Tugas Akhir. Teknik Telekomunikasi. STT Telkom: Bandung.

[5] Webb, Jared, James Ashley. 2012. *Beginning sensor Kinect Programming with the Microsoft sensor Kinect SDK.* Amerika: Apress.

[6] Catuhe, David. 2012. *Programming with the sensor Kinect for Windows Software Development kit.* Washington: Microsoft Press.

[7] Smisek, Jan, Michal Jancosek, Tomas Pajdla. 2011. 3D With *sensor Kinect*. Prague: Czech Technical University.

[8] Kean, Sean, Jonathan Hall, Phoenix Perry. 2010. *Meet the sensor Kinect: An Introduction to Programming Natural User Interfaces.* New York: Technology In Action™.

[9] WHO Media centre, "Deafness and hearing loss," WHO, Februari 2014. [Online]. Available: http://www.who.int/mediacentre/factsheets/fs300/en/. [Diakses 3 Agustus 2014].
[10] A.Jana, Kinect for Windows SDK Programming Guide, Birmingham: Packt Publishing Ltd., 2012.