PENGENALAN INDIVIDU BERDASARKAN GAIT MENGGUNAKAN SENSOR ACCELEROMETER

HUMAN RECOGNITION BASED ON GAIT USING ACCELEROMETER SENSOR

Anggit Syorgaffi[1], Achmad Rizal[2], Sugondo Hadiyoso[3]

[1,2,3]Teknik Telekomunikasi, Fakultas Teknik, Univeristas Telkom

Jl. Telekomnukiasi, Dayeuh Kolot Bandung 40257 Indonesia

[1]anggit.syorgaffi@gmail.com, [2]ach_rizal@yahoo.com, [3]sugondo.hadiyoso@gmail.com

Abstrak

Setiap individu memiliki keunikan tersendiri dalam cara berjalan atau *gait*. Karena itu *gait* dapat digunakan untuk mengenali seorang individu. Sehingga *gait* dapat diimplementasikan sebagai biometrik. *Accelerometer* adalah sensor untuk mengukur dan mendeteksi getaran, ataupun untuk mengukur percepatan, yang juga bergantung pada arah atau orientasi. Sensor *accelerometer* sudah digunakan secara luas dikehidupan sehari-hari, terutama pada *smartphone*. Sehingga dimungkinkan untuk mengukur pergerakan individu saat berjalan menggunakan sensor *accelerometer* yang tertanam pada *smartphone*. Penelitian ini dilakukan pengenalan individu berdasarkan gait dengan menfaatkan sensor *accelerometer* yang tertanam pada *smartphone*. Untuk pengolahan data atau melakukan analisis pengenalan mengimplementasikan metode *Mel-Frequency Cepstral Coefficient* dan *Hidden Markov Model*. Metode *Mel-Frequency Cepstral Coefficient* digunakan untuk melakuakan ekstraksi ciri. Sedangkan metode *Hidden Markov Model* digunakan untuk melakukan klasifikasi. Sinyal *gait* sendiri terdiri dari empat sinyal yaitu sinyal sumbu x, y, z, dan magnitude (m). Penelitian ini menghasilkan tingkat akurasi paling tinggi mencapai 96%, pada model jumlah *states* 2, jumlah MFCC 30, dan ukuran *codebook* sebesar 8. Kombinasi antar sinyal x-y-z-m menghasilkan tingkat akurasi yang paling tinggi. Secara keseluruhan tingkat akurasi sistem mencapai 96%.

Kata kunci : *Gait, Accelerometer, Hidden Markov Model, MFCC*

Abstract

Every person or indiviual has every uniqe way of walking. The manner or style of walking is called gait. Gait can be used recognition of a person or individual. Because gait can be implemented for biometric. Accelerometer is a sensor that can measures and detects vibration, or acceleration, that depend on direction or orientation. Accelerometer sensor has been used in daily Life widely every day, mostly in smartphone. So, It is possible to measure movement of person when he is walking using accelerometer sensor that is embeded in smartphone. In this research, tried to do recognition of individual based on gait using accelerometer sensor that is embeded in smartphone. The gait data is processed and analyzed by implementating Mel-Frequency Cepstral Coefficient method and Hidden Markov Model Method. Mel-Frequency Cepstral Coefficient Method is used to extract feature from gait data. Hidden Markov Model method is used for clasfication. Gait data consist of four signal, axis x signal, axis y signal, axis z signal, and magnitude (m) signal. In this research, produces best accuracy up to 96%, at model consist of 2 states, 30 MFCC, and 8 codebook size. Combination of x-y-z-m signal produces best accuracy among other possible commbinations. Generally, system highest accuracy up to 96%.

Key World : *Gait, Accelerometer, Hidden Markov Model, MFCC*

1 Pendahuluan

. *Gait* merupakan cara berjalan individu. Dalam penjelasan yang lebih spesifik, *gait* merupakan ciri khas cara berjalan individu yang dipengaruhi oleh berat tubuh, panjang kaki, alas kaki yang sedang di pakai, dan postur tubuh, dan hal-hal lain [1]. Penelitian tentang *gait* telah dilakukan sejak dahulu dari segi psikologi. Dalam eksperimen terakhir yang telah dilakukan, subjek manusia dapat mengenali tipe pergerakan dengan melihat *light bulb* yang disematkan pada subjek lain yang sedang berjalan [2].

Pengenalan *gait* adalah dimensi baru dalam autentifikasi biometrik. Dalam satu dekade terakhir, pengenalan *gait* telah menjadi bidang riset yang aktif dan banyak dipelajari. Alasan kepopuleran riset *gait*, karena sifat alami penganalan *gait* yang tidak memerlukan interaksi secara langsung. hal ini memungkinkan melakukan penganalan

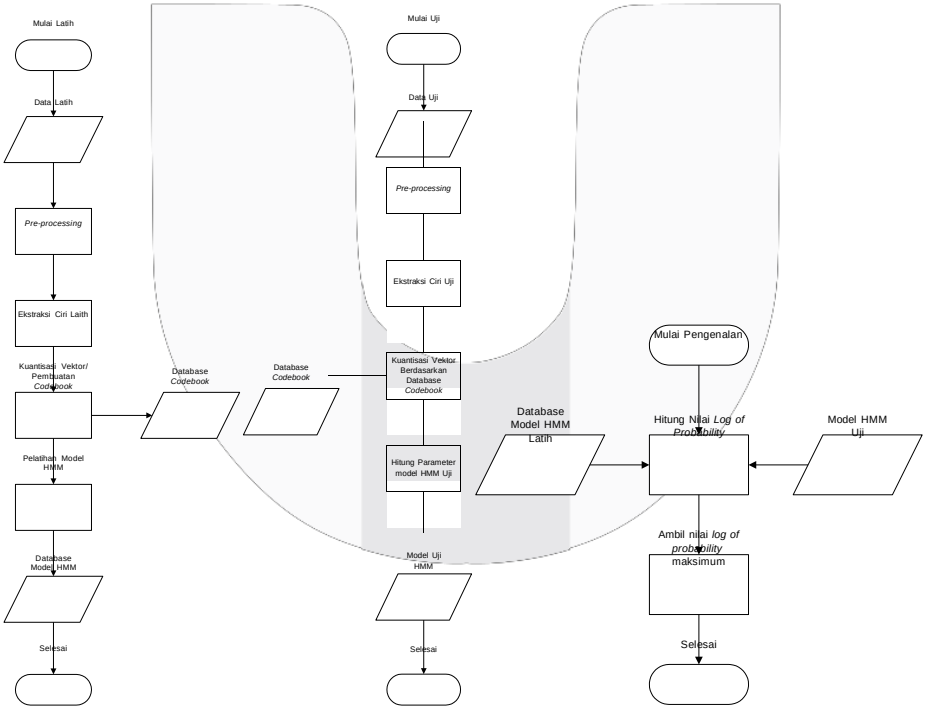
biometrik oleh penggunanya tanpa disadari. Dalam 18 tahun terakhir, sebagian besar penganalan *gait* dilakukan menggunakan pendekatan visual menggunakan video. Pada era yang sama, pengunaan sensor yang dipasang pada lantai untuk melakukan pengenalan gait juga dipelajari. Kemudian muncul kebutuhan akan pengenalan *gait* yang menggunakan sensor yang dapat dipakai atau *wearable-sensor*. Dan pada tahun 2005, untuk pertama kalinya muncul ide untuk memanfaatkan akselerasi sebagai parameter untuk pengenalan *gait.* Akselerasi ini direkam menggunakan sensor *accelerometer* yang dipasang pada tubuh manusia. Sekarang ini, sensor *accelerometer* telah menjadi bagian dari *smartphone*. Oleh karena itu, hal ini memungkinkan untuk melakukan pengenalan *gait* menggunakan sensor *accelerometer* pada *smartphone* sebagai alternatif sistem pengenalan [3].

Untuk itu pada penelitian ini dilakukan identifikasi gait individu dengan memanfaatkan sensor *accelerometer* yang tertanam pada *smartphone*. Dilakukan pengambilan data sensor *accelerometer* dari gait individu dengan memanfaatkan *smartphone*. Data yang didapat kemudian akan diproses menggunakan metode *Mel-Frequency Cepstral Coefficient* untuk melakukan ekstraksi ciri dan metode *Hidden Markov Model* untuk melakukan klasifikasi.

Penelitian ini bertujuan untuk melakukan analisis data gait, untuk dicari cirinya dengan mengimplementasikan *Mel-Frequency Cepstral Coefficient.* Dengan ciri yang didapatkan, dilakukan identifikasi individu dengan memafaatkan metode *Hidden Markov Model.* Selanjutnya akan dianalisis hasil akurasi dan performa sistem pada penelitian ini.

2. Perancangan Sistem

Dalam penelitian ini dirancang suatu sistem yang dapat mendeteksi sinyal *gait* dari sensor *accelerometer* secara *non-realtime*. Terdapat tiga tahap utama pada sistem pengenalan individu ini yaitu *pre-processing*, ekstraksi ciri, proses klasifikasi. Pada *pre-processing* akan dilakukan proses normalisasi, *walk extraction*, interpolasi. Kemudian dilanjutkan dilakukan ekstraksi ciri menggunakan *Mel-Frequency Cepstral Coefficient Extraction* (MFCC). Pada proses klasifikasi akan mengunakan metode *Hidden Markov Model* (HMM). Sistem akan melakukan dua proses yaitu proses latih, proses uji, dan proses pengenalan. Blok perancangan sistem ditunjukkan pada gambar 1.



Gambar 1 Diagram Alir Sistem Pengenalan Gait Menggunakan HMM

Sinyal *gait* ini terdiri dari tiga sinyal yaitu sinyal *accelerometer* pada sumbu x, y, dan z. Sinyal accelerometer x, y, z digunakan untuk menghitung magnitudo vektor (m) dari sinyal *gait*, sesuai persamaan 1.

$$m = \sqrt{x^2 + y^2 + z^2} \qquad (1)$$

Dari persamaan tersebut maka didapat empat sinyal *gait* x, y, z, dan m. Empat sinyal tersebut akan digunakan untuk melakukan pengenalan individu. Pada Tugas Akhir ini klasifikasi akan menggunakan sinyal x, y, z, m secara terpisah dan kombinasi keempat sinyal tersebut x-y-z-m.

2.1 Pre-Processing

1. Normalisasi

Pada proses ini akan dilakukan untuk menghilangkan pengaruh gravitasi pada sinyal *gait*. Hal ini dikarenakan sensor *accelerometer* pada *smartphone* tidak terkalibrasi secara sempurna. Untuk menghilangkan pengaruh tersebut, data sinyal *gait* akan geser mendekati nol. Hal ini dilakukan dengan cara melakukan operasi pengurangan sinyal gait oleh rata-rata sinyal gait ? Ditunjukan oleh persamaan 2 [4].

$$\hat{m}_{?}(?) = \hat{m}_{?}(?) - ?_{?} \quad ? \in \{?, ?, ?\} \qquad (2)$$

2. Walk Extraction

Proses ini dilakukan untuk menghilangkan data pada saat diam atau tidak bergerak pada sinyal *gait*. Pada proses ini diharapkan hanya sinyal *gait* pada waktu bergerak saja yang diambil untuk diproses ke tahap selanjutnya [4]. Proses ini dilakukan dengan cara mencari nilai standar deviasi dari sinyal *gait*. Nilai standar deviasi tersebut dihitung berdasarkan persamaan 3 [5].

$$?_{?}(?) = \left(\frac{1}{1-?}\sum_{?=1}^{?}(?_{?} - ?)^{2^{\frac{1}{2}}}\right) \qquad (3)$$

3. Interpolasi

Pada tahap ini, data sinyal *gait* diinterpolasi secara linier dengan frekuensi sampling tetap. Hal ini dilakukan karena keterbatasan dari *Android Aplication Programming Interface* (API). Nilai akselerasi pada sensor *accelerometer* hanya bisa didapat saat ada perubahan posisi pada sensor, dan kemudian mungkin terkena delay oleh *background System android*. Sehingga data sinyal *gait* yang didapat tidak stabil terhadap waktu dan frekuensi samplingnya tidak tetap. Alasan dilakukan interpolasi untuk menjaga agar tidak banyak nilai data yang hilang [4]. Perhitungan interpolasi secara linier dijelaskan oleh persamaan 4 [6].

$$? = ?_{?} + (?_{?} - ?_{?})\frac{?_{?} - ?_{?}}{?_{?} - ?_{?}} \qquad (4)$$

2.2 Mel Frequency Cepstral Coefficient Extraction

Pada tahap ekstraksi ciri terdapat beberapa tahapan, yaitu:

1. Frame Blocking

Dalam proses ini, sinyal *gait* dibagi ke dalam beberapa *frame*. Saat melakukan *frame blocking*, mungkin saja terjadi *overlap* antar *frame*, hal ini bertujuan agar memperkecil kemungkinan adanya sinyal yang hilang [5]. Pada penelitian ini digunakan ukuran *frame* 1440 ms dengan *Frame shift* 48 ms [4].

2. Windowing

Windowing adalah metode untuk meminimalisasi ketidakberlanjutan sinyal antara akhir *frame* satu ke *frame* selanjutnya. *Windowing* dilakukan pada setiap *frame*. Pada penelitian ini, akan digunakan *window Hamming*. Persamaan *Window Hamming* dituliskan secara matematis sebagai berikut : [5]

$$\hat{m}_{?}(?) = \begin{cases} 0.54 - 0.46\,?\,?\,?\,\dfrac{2\pi ?}{? - 1} & 0 \le ? \le ? - 1 \\ 0; & \text{[illegible]} \end{cases} \qquad (5)$$

3. Fast Fourier Transform (FFT)

FFT merupakan *fast algorithm* dari *Discrete Fourier Transform* (DFT) digunakan untuk mengkonversi setiap *frame* dari domain waktu menjadi domain frekuensi, FFT didefinisikan sesuai persamaan sebagai berikut : [7]

$$X(k) = \sum_{n=0}^{N-1} x(n)\, e^{\frac{-j2\pi kn}{N}} \qquad (6)$$

4. Mel-Frequency Wrapping

Pada proses ini dilakukan konversi untuk setiap data dengan frekuensi *f* diukur dalam Hz, ke dalam skala yang disebut skala 'Mel'. Skala Mel adalah *frequency spacing* yang linier dibawah 1000 Hz dan sebuah *algorithmic spacing* untuk frekuensi diatas 1000 Hz. Persamaan konversi dari skala Hz ke skala Mel dan sebaliknya didefinisikan sebagai berikut : [5]

$$f_{mel} = 2592 * \log_{10}\left(1 + \frac{f}{100}\right) \quad (7)$$

$$f^{-1}(f_{mel}) = 700\left(\exp\left(\frac{f_{mel}}{1125}\right) - 1\right) \quad (8)$$

Skala Mel digunakan untuk menghasilkan *filterbank* yang diinginkan. Proses perhitungan untuk menghasilkan *filterbank* dimulai dengan menetukan *lower* dan *upper frequency* dari sebuah sinyal. Kemudian nilai dari *lower* dan *upper freuquency* dikonversi ke dalam skala Mel. Selanjutnya nilai hasil konversi akan dijadikan rentang nilai untuk dibagi sesuai jumlah *filterbank* yang diinginkan. Setelah dihasilkan *filterbank,* dilakukan perkalian antara *filterbank* dengan spektral daya dari hasil proses FFT untuk menghasilkan energi *filtebank* [7].

Hasil perkalian spektral daya dengan *filterbank* tersebut kemudian dihitung nilai *log* dari masing-masing energi. Setelah didapatkan nilai *log* dari masing-masing energy, kemudian dilakukan perhitungan DCT *(Discrete Cosine Transform)* untuk masing-masing energi tersebut [5].

5. Cepstrum

Proses selanjutnya adalah mengkonversi *Mel Cepstrum* kembali ke domain waktu. Hasil dari proses inilah yang disebut sebagai *Mel-Frequency Cepstral Coefficient*. Representasi cepstral dari spektrum sinyal berasal dari sifat spectrum sinyal untuk analisa *frame* yang ada. MFCC bisa didapat dari persamaan sebagai berikut : [5]

$$c_n = \sqrt{\frac{2}{N}} \sum_{m=1}^{M} S_m \cos\left(\frac{\pi n}{M}(m - 0.5)\right) \quad (9)$$

2.3 Kuantisasi Vektor

Kuantisasi vektor adalah metode *clustering* sinyal. Sinyal dikategorikan kedalam sejumlah *cluster*. Masing-masing *cluster* merepresentasikan data milik suatu populasi tertentu dengan yang mirip atau memilik perbedaan minimum karakteristik spektral. Setiap *cluster* diwakili oleh sebuah titik *centroid* [8].

Pada penelitian ini proses *clustering* menggunakan algoritma *K-Means Clustering*. Metode *K-means* berusaha mengkategorikan data yang ada ke dalam beberapa kelompok. Dengan metode K-means dicari titik-titik *centroid* atau didapat dikatakan sebagai *codebook*. Kemudian dicari *codeword* dari sinyal berdasarkan *codebook* yang telah didapat. *Codebook* dan *codeword* digunakan sebagai masukan proses HMM.

2.4 Klasifikasi Hidden Markov Model

1. Pemodelan Hidden Markov Model

Pada sistem pengenalan sinyal *gait* menggunakan metode klasifikasi HMM, masing-masing individu yang akan dikenali memiliki model individu masing-masing. Sebelum nenetukan model individu, ditentukan jumlah state (N) dan barisan observasi (O) yang akan digunakan Model HMM pada tiap-tiap individu terdiri dari parameter A berukuran $N \times N$ (matriks transisi), B berukuran $N \times M$ (matriks observasi), dan phi berukuran $1 \times N$ (matriks inisialisasi) [5].

2. Penentuan Barisan Observasi

Sebelum memasuki program training dalam *Hidden Makov Model*, suatu ciri dari sinyal atau data latih yang telah kita dapatkan terlebih dahulu di *cluster* atau di kelompokkan menjadi beberapa kelas. Dalam hal ini pengelompokkan data ciri atau data latih tersebut menggunakan *k-means*. Metode *k-means* digunakan untuk membentuk *codebook* dengan ukuran tertentu sesuai jumlah yang diinginkan. *Codebook* ini merupakan nilai acuan untuk mengelompokkan ciri sinyal suara menjadi beberapa kelas berdasarkan nilai terdekat dari *codebook* tersebut. Output dari proses *clustering* ini adalah berupa indeks-indeks barisan observasi [5].

3. Reestimasi Hidden Markov Model

Pada tahap awal pelatihan model individu, ditentukan model *A*, *B*, *phi* awal yang nilai-nilainya ditentukan secara random sesuai jumlah state dan barisan observasi. Selanjutnya, sistem mengestimasi model parameter, $\lambda = (A, B, \pi)$ dari sinyal masukan yang direpresentasikan oleh barisan observasi yang diperoleh dari tahap sebelumnya. Nilai – nilai inisialisasi pada proses pelatihan model kata kemudian di *reestimate* secara berulang agar mendapatkan parameter HMM yang paling akurat, karena parameter awal adalah tebakan awal yang masih belum diuji [9].

4. Proses Pengenalan

Pada proses ini dilakukan pengenalan data uji terhadap model HMM yang telah didapatkan pada proses latih. Sebelum masuk tahap pengenalan, dilakukan proses ekstraksi ciri dari masing-masing input sinyal *gait* dan kemudian mengelompokkannya ke dalam *cluster-cluster* tertentu sesuai ciri spektralnya. Selanjutnya dilakukan ditentukan nilai parameter *A*, *B*, phi dari masing-masing sinyal *gait*. Setelah itu, nilai parameter-parameter tersebut akan dibandingkan ke setiap model atau parameter HMM yang telah didapatkan dari proses HMM latih. Perbandingan itu dilakukan dengan cara mencari nilai *log of probability* dari data uji di semua model HMM yang didapatkan. Hasil *log of probability* yang paling maksimumlah yang nantinya akan menjadi hasil dari proses uji ini.

3. Hasil Dan Analisis

3.1 Analisis Pengaruh Parameter States

Tingkat akurasi sistem dipengaruhi oleh ukuran *states*, seperti ditunjukan oleh gambar 1. Terlihat bahwa semakin banyak jumlah *states* yang digunakan mengurangi tingkat akurasi pengenalan sistem. Penurunan tingkat akurasi terjadi secara signifikan antara jumlah *state* 2 sampai 4. Sedangkan antara jumlah *state* 5 sampai 8 hanya sedikit terjadi naik turun tingkat akurasi sistem. Penurunan tingkat akurasi akibat bertambahnya jumlah *state*, adalah dikarenakan dengan bertambahnya jumlah *state* maka akan memperbesar ukuran matriks transisi antar *state* yang digunakan dalam perhitungan dalam metode HMM. Ukuran matriks transisi yang lebih besar tidak cocok untuk sinyal *gait*. Hal ini dikarenakan informasi yang disajikan oleh sinyal *gait* tidak sedetil atau sebanyak sinyal suara atau gambar. Hal tersebut ditunjukan oleh gambar 1, untuk ukuran MFCC 40 yang berisi data ekstraksi ciri lebih besar dibandingkan MFCC 15, MFCC 40 menghasilkan akurasi yang lebih tinggi dan lebih stabil pada jumlah *state* 3 sampai 7 dibandingkan dengan MFCC 15. Informasi yang kurang detil ini karena keterbatasan sensor *accelerometer* dalam melakukan pengambilan data, yang hanya sekitar 107 sampel/sekon. Jumlah data latih juga mempengaruhi kemampuan metode HMM. Karena variasi data latih yang diberikan kurang menyebabkan kemampuan HMM untuk memprediksi perpindahan *state* menjadi kurang baik. Dengan jumlah *state* yang semakin besar dan kemampuan untuk memprediksi perpindahan *state* yang buruk karena jumlah data latih yang tetap, maka akan banyak terjadi kesalahan prediksi *state* (*error state*). Karena banyak terjadi *error state*, maka kemampuan sistem untuk mengenali individu memburuk yang berpengaruh pada akurasi sistem.

Tabel 1 Pengaruh state terhadap akurasi sistem, diukur pada codebook 8 dan sinyal x-y-z-m

Jumlah State	MFCC 5	MFCC 10	MFCC 15	MFCC 20	MFCC 30	MFCC 40
2	0,78	0,76	0,76	0,56	0,78	0,88
3	0,58	0,62	0,46	0,54	0,3	0,54
4	0,46	0,44	0,56	0,36	0,58	0,3
5	0,38	0,42	0,36	0,44	0,34	0,32
6	0,46	0,28	0,28	0,46	0,34	0,36
7	0,34	0,22	0,24	0,18	0,3	0,38
8	0,26	0,36	0,3	0,32	0,24	0,2

3.2 Analisis Pengaruh Parameter MFCC

Tabel 2 Pengaruh ukuan MFCC terhadap akurasi sistem, diukur pada state 2 dan sinyal x-z

Parameter	MFCC 5	MFCC 10	MFCC 15	MFCC 20	MFCC 30	MFCC 40
Codebook 4	0,4	0,44	0,42	0,42	0,36	0,48
Codebook 8	0,5	0,54	0,58	0,62	0,68	0,66
Codebook 16	0,44	0,68	0,7	0,66	0,64	0,72
Codebook 32	0,48	0,52	0,56	0,68	0,66	0,54

Dari tabel 2, dapat diambil garis besar bahwa dengan bertambahnya jumlah MFCC akan menaikkan tingkat akurasi sistem. Hal tersebut disebabkan semakin besar jumlah MFCC maka semakin banyak variasi nilai ekstraksi ciri. Model HMM yang dihasilkan akan lebih spesifik apabila variasi nilai ekstraksi ciri semakin banyak. Dengan model yang lebih spesifik akan mempermudah sistem dalam membedakan model satu dengan yang lain. Sehingga hal tersebut meningkatkan tingkat akurasi sistem.

Sistem menghasilkan tingkat akurasi tertinggi saat jumlah MFCC adalah 30 buah koefisien. Jumlah MFCC 30 dianggap sebagai jumlah MFCC optimal bagi sistem, karena menghasilkan tingkat akurasi tinggi dengan 30 koefisien. Sehingga waktu komputasi pada saat ekstraksi ciri lebih singkat. Dibandingkan dengan jumlah MFCC 40 yang menghasilkan tingkat akurasi yang hampir mendekati, waktu komputasi MFCC dengan 30 koefisien lebih cepat.

3.3 Analisis Pengaruh Parameter Codebook

Berdasarkan hasil pengujian secara keseluruhan, terlihat bahwa semakin besar *codebook*, maka tingkat akurasi sistem semakin meningkat. Namun, pada *codebook* 64 sampai 128 terjadi penurunan tingkat akurasi sistem yang cukup signifikan. Kondisi ini disebabkan karena sinyal *gait* memiliki rentang data yang cukup kecil atau sempit, sehingga cukup sulit membagi rentang data ke dalam ukuran *codebook* yang cukup besar. Selain itu, dengan ukuran *codebook* yang Besar dan rentang data yang kecil akan menyebabkan jarak antar *codebook* sangat pendek atau kecil. Hal ini menyebabkan pada saat kuantisasi vektor dengan acuan *codebook* menjadi sulit, karena cukup sulit untuk membandingkan data yang hampir mirip.

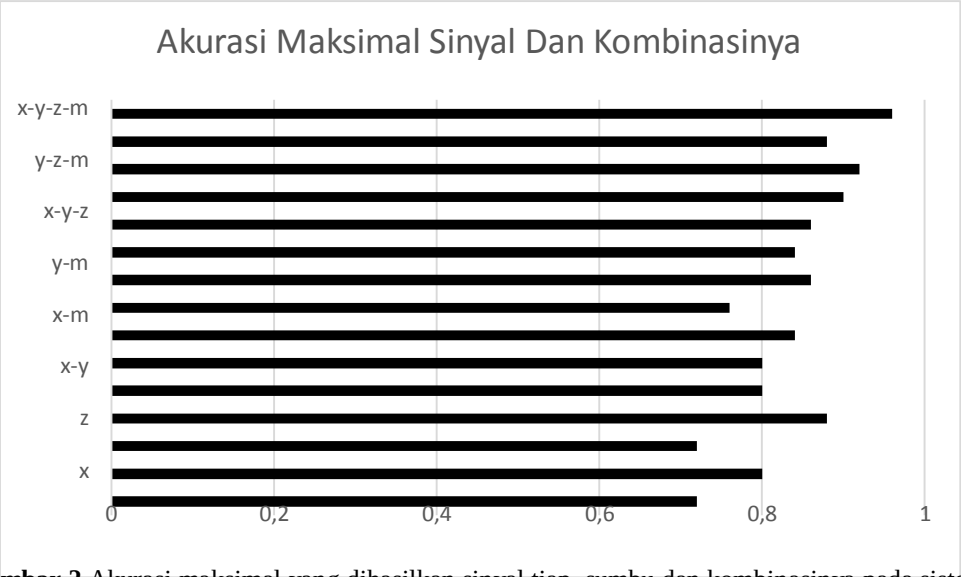
Tabel 3 Pengaruh jumlah codebook terhadap akurasi sistem, diukur pada state 2, 3, 4 dan sinyal x-y-z-m

MFCC		MFCC 15	MFCC 20	MFCC 30
Codebook	4	0,86	0,76	0,78
		0,66	0,64	0,54
		0,48	0,48	0,54
	8	0,86	0,8	0,96
		0,7	0,68	0,58
		0,6	0,6	0,56
	16	0,9	0,82	0,88
		0,52	0,78	0,44
		0,26	0,5	0,34
	32	0,88	0,92	0,86
		0,64	0,44	0,56
		0,2	0,22	0,3
	64	0,52	0,52	0,62
		0,16	0,16	0,22
		0,12	0,1	0,14
	128	0,1	0,1	0,1
		0,1	0,1	0,1
		0,1	0,1	0,1

3.4 Analisis Sinyal yang Diamati

Untuk menganalisis kemampuan dari setiap ciri sinyal yaitu sinyal sumbu x, y, z dan magnitudo (m), maka sinyal masing-masing diuji secara terpisah. Selanjutnya dicari performa terbaik dari kombinasi antar sinyal tersebut. Pengujian mencari pengaruh sinyal antar sumbu yang menghasilkan akurasi terbaik.

Dari keempat jenis sinyal x, y, z, dan m, hanya sinyal y dan m yang layak dipakai untuk mendapatkan akurasi yang cukup tinggi. Kemudian dengan melakukan kombinasi antara dua pasang sinyal didapatkan nilai akurasi yang semakin meningkat, namun masih ada yang kombinasi sinyal yang belum sesuai harapan seperti kombinasi sinyal x-z. Sehingga dilakukan kombinasi sinyal 3 pasang dan didapatkan tingkat akurasi yang tinggi. Pada kombinasi 3 pasang sinyal, kombinasi sinyal x-y-m lah yang paling menonjol. Untuk menyempurnakan nilai akurasi agar lebih tinggi dilakukan kombinasi dari keempat jenis sinyal dan didapatkan nilai akurasi tertinggi sebesar 96%.



Gambar 2 Akurasi maksimal yang dihasilkan sinyal tiap sumbu dan kombinasinya pada sistem

4. Kesimpulan

Nilai akurasi keseluruhan dengan variasi parameter *states*, MFCC dan *codebook* terhadap semua kemungkinan kombinasi sinyal x, y, z, dan m. Menghasilkan nilai akurasi antara 10% sampai 96%. Perubahan jumlah *state* mempengaruhi hasil akurasi sistem, jumlah *state* yang paling optimal untuk sistem adalah 2. Semakin besar jumlah *state*, akurasi sistem memburuk. Karena data latih yang kurang besar atau kurang variasi data. Semakin besar jumlah MFCC yang dipakai, semakin besar pula tingkat keberhasilan sistem dalam mengenali individu. Semakin besar ukuran *codebook*, maka semakin besar tingkat akurasi yang dicapai oleh sistem. Tetapi untuk ukuran *codebook* lebih dari 32 akurasi menurun untuk sistem pengenalan *gait.* Pada Tugas Akhir ini, kombinasi sinyal yang paling optimal adalah kombinasi 4 pasang sinyal x-y-z-m. Nilai parameter yang paling optimal pada pengujian adalah jumlah *state* 2, jumlah MFCC 30, dan ukuran *codebook* 8.

DAFTAR PUSTAKA

[1] L. Lee and W. Grimson, "Gait analysis for recognition and classification," in *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on 20-21 May*, Washington, DC, 2002.

[2] C. BenAbdelkader, R. Cutler dan L. Davis, "Gait Recognition using Image Self-Similarity," *Eurasip Journal on Applied SIgnal Processing*, 2004.

[3] M. Muaaz dan C. NIckel, "Influence of different walking speeds and surfaces on accelerometer-based biometric gait recognition," *Telecommunications and Signal Processing (TSP), 2012 35th International Conference on 3-4 July* , pp. 508 - 512 , 2012.

[4] C. Nickel, J. Buchmann dan C. Busch, Accelerometer-based Biometric Gait Recognition for Authentication on Smartphones, Dissertation, Darmstadt: Technischen Universitat Darmstadt, 2012.

[5] N. K. R. Intan, Analisis Dan Simulasi Sistem Penerjemah Kata Berbahasa Bali Ke Bahasa Inggris Berbasis Speech To Text Secara Real Time Menggunakan Metode Klasifikasi Hidden Markov Model, Tugas Akhir, Bandung: Universitas Telkom, 2014.

[6] E. Meijering, "A Chronology of Interpolation : From Ancient Astronomy to Modern Signal and Image Processing," *Proceedings of the IEEE*, vol. 90, no. 3, pp. 319-342, 2002.

[7] P. Mahesha dan D. Vinod, "Vector Quantization and MFCC based Classification of Dysfluencies in Stuttered Speech," *Bonfring International Journal of Man Machine Interface*, vol. II, no. 3, 2012.

[8] V. Ayuningtias, Pengkategorian Hasil Pencarian Dokumen Dengan Clustering, Tugas Akhir, Bandung: Universitas Telkom, 2008.

[9] A. Buono, "Perluasan Metode Mfcc 1d ke 2d Sebagai Ekstraksi Ciri Pada Sistem Identifikasi Pembicara Menggunakan Hidden Markov Model (HMM), Skripsi," Institut Pertanian Bogor, Bogor, 2009.

[10] D. Post, Gait Analysis Review, Aerospace & Mechanical Engineering, University of Notre Dame, 2006.

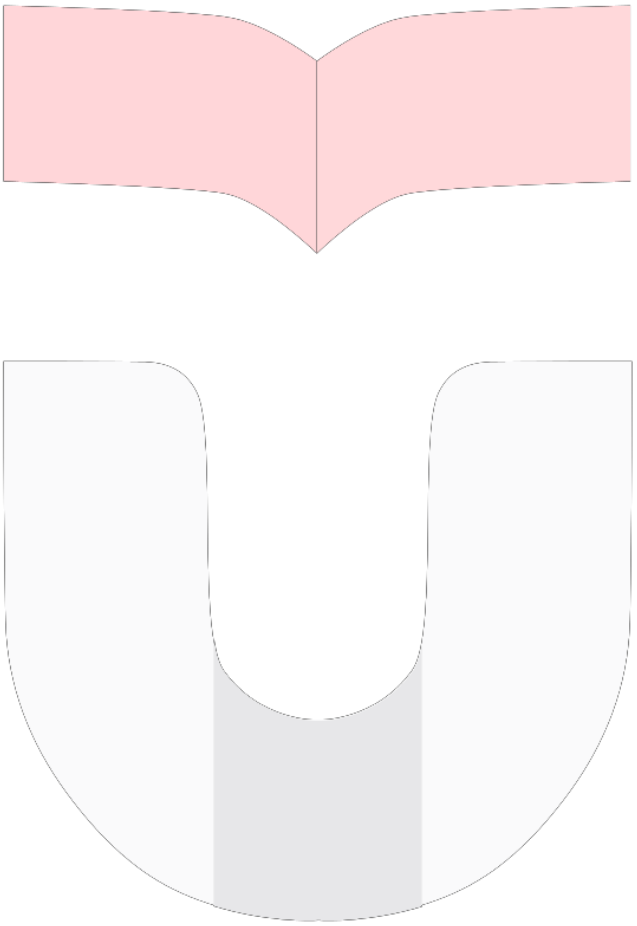
[11] R. Wiryadinata, "Prinsip kerja sensor accelerometer," WordPress, 29 April 2009. [Online]. Available: http://wiryadinata.web.id/?p=22. [Diakses 14 Mei 2014].

[12] M. F. Nowlan dan Savvides, "Human Identification via Gait Recognition Using Accelerometer Gyro Forces," *CPSC-536-Networked Embedded Systems and Sensor Networks*, 2009.

[13] Y. Mingjing , Z. Huiru, W. Haiying, M. Sally dan D. Newell, "iGAIT: An interactive accelerometer based gait analysis," *Computer Methods And Program In Biomedicine* , vol. I, no. 08, pp. 715-723, 2012.

[14] M. F. Kushidayati, Database Transkrip Akord Instrumen Tunggal Menggunakan Metode Enhanced Pitch Class Profile (EPCP), Skripsi, Surabaya: Institut Teknologi Sepuluh Nopember, 2010.

[15] L. Xinguang, C. Jiahua dan L. Zhenjiang, "English Sentence Recognition Based on HMM and Clustering," *American Journal of Computational Mathematics*, no. 3, pp. 37-42, 2013,.

[16] J. Trdine dan P. Dymarski, Hidden Markov Models, Theory and Applications, Rijeka, Croatia: InTech, 2011.

[17] L. Rabiner dan H. J. Biing, Fundamental Of Speech Recognition, Mexico: Prentice-hall International, 1993.