

Makalah Seminar Tugas Akhir

SISTEM NAVIGASI PERSONAL BERBASIS SENSOR MEMS (*Micro Electro-Mechanical System*)

Laras Dwi Kawuri¹, Iwan Setiawan², Darjat²
Jurusan Teknik Elektro, Fakultas Teknik, Universitas Diponegoro
Jl. Prof. Sudharto, Tembalang, Semarang, Indonesia

Abstrak

Sistem navigasi merupakan suatu cara yang digunakan untuk menentukan posisi dan arah perjalanan dari keadaan awal. Saat ini, muncul berbagai penelitian untuk menciptakan sistem navigasi yang diterapkan pada manusia. Beberapa sensor dengan tingkat akurasi yang cukup memadai dalam mengenali pergerakan manusia dibutuhkan untuk mendapatkan informasi posisi dan arah perjalanan. Sensor percepatan dengan teknologi *Micro Electro Mechanical System (MEMS)* dapat menjadi solusi alternatif untuk mengestimasi pergerakan manusia tersebut, dengan menggunakan metode *Pedestrian Dead Reckoning (PDR)*.

Melihat kenyataan tersebut, maka pada tugas akhir ini akan dibuat suatu sistem navigasi personal bagi pejalan kaki dengan menggunakan sensor yang berteknologi MEMS. Sistem ini akan mendeteksi jumlah langkah kaki dengan menggunakan sensor *accelerometer H48C*, kemudian panjang langkah kaki setiap step diketahui melalui nilai input rata-rata panjang satu langkah kaki. Kompas digital *CMPS03* akan digunakan untuk mengetahui arah pergerakan terhadap medan magnet bumi, yang diupdate setiap terdeteksi satu langkah kaki.

Kata kunci: sistem navigasi, PDR, accelerometer, kompas digital, deteksi step.

I. PENDAHULUAN

1.1 Latar Belakang

Penggunaan *Global Positioning System (GPS)* telah berkembang sebagai alat yang tepat sebagai sistem navigasi diluar ruangan (*outdoor*). Ada beberapa kelemahan atau keterbatasan GPS, terutama dalam wilayah perkotaan, diantaranya adalah sinyal yang hilang karena terhalang oleh bangunan atau pergerakan manusia, dan adanya pelemahan sinyal. GPS juga tidak tepat bila digunakan didalam ruangan (*indoor*), padahal suatu sistem navigasi yang diimplementasikan pada manusia diharapkan dapat digunakan dimana saja, *indoor* maupun *outdoor*.

Kemajuan dalam bidang komputasi, komunikasi, dan kemampuan sensor bermunculan sebagai prasarana utama dalam hal *mobile computing*. Ratusan bahkan ribuan sensor dalam ukuran kecil, dengan harga terjangkau, dan sumber daya yang kecil dapat dengan cepat digunakan untuk memonitoring keberadaan seseorang. Hal tersebut merupakan syarat yang diperlukan untuk membuat suatu sistem navigasi yang portabel. Salah satunya, dalam beberapa tahun terakhir ini muncul penelitian mengenai sistem berbasis *Accelerometer* berteknologi MEMS (*Micro Electro Mechanical Systems*) yang digunakan sebagai sistem navigasi personal.

Oleh karena itu pada tugas akhir ini akan dikembangkan salah satu pengaplikasian sensor berteknologi MEMS yaitu *Accelerometer H48C* dan kompas digital yang akan dipasang di sepatu pejalan kaki. Informasi yang akan didapatkan

dari sistem ini adalah jumlah langkah dan posisi pejalan kaki. Dengan aplikasi ini diharapkan dapat digunakan sebagai alternatif sistem navigasi personal bagi para pejalan kaki atau untuk pengembangan lebih lanjut dapat digunakan untuk mengkompensasi beberapa kelemahan sistem navigasi dengan menggunakan GPS.

1.2 Tujuan

Tujuan yang ingin dicapai dalam Tugas Akhir ini adalah merancang dan membuat sistem instrumentasi *accelerometer* dan kompas digital sebagai salah satu komponen navigasi personal untuk mendeteksi langkah kaki manusia serta mengetahui posisi dan jarak dari posisi awal berbasis Mikrokontroler *ATMega 8535*.

1.3 Pembatasan Masalah

Dalam pembuatan Tugas Akhir ini penulis membatasi permasalahan sebagai berikut :

1. Sensor *accelerometer* yang digunakan memiliki 3 derajat kebebasan.
2. Pembuatan sistem navigasi ini menggunakan sensor *accelerometer* sebagai pendeteksi langkah kaki dan kompas digital untuk mengetahui posisi.
3. Dalam pengujian sistem pendeteksi langkah kaki ini hanya digunakan untuk penulis dan dalam medan yang keadaan tanahnya rata.
4. Langkah kaki yang terdeteksi hanya untuk langkah kaki maju dan dalam kecepatan normal.

1) Mahasiswa Jurusan Teknik Elektro UNDIP

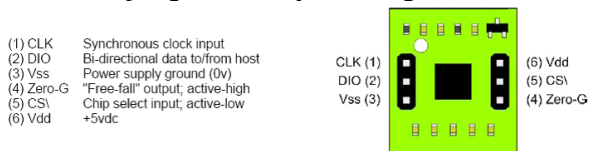
2) Staf Pengajar Jurusan Teknik Elektro UNDIP

5. Tidak membahas secara mendetail rangkaian di dalam sensor *accelerometer*.
6. Sistem monitoring yang digunakan adalah laptop dengan bantuan program Microsoft Visual C#.

II. DASAR TEORI

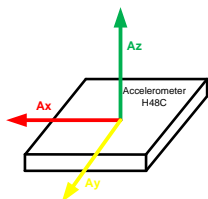
2.1 Accelerometer Hitachi H48C

Hitachi H48C adalah modul sensor accelerometer terintegrasi yang dapat merasakan gaya gravitasi dalam tiga buah sumbu angular (X, Y, dan Z). Chip ini berbahan silicon menggunakan teknologi MEMS (*Micro Electro-Mechanical System*) yang memungkinkannya untuk digunakan secara langsung tanpa perlu dikalibrasi. Pada modul ini terdapat regulator tegangan untuk mensuplai tegangan 3.3 volt untuk chip H48C, 12-bit ADC jenis MCP3204 untuk membaca tegangan keluaran dari chip H48C dan pengkodisi sinyal analog.



Gambar 1 Konfigurasi pin modul H48C

Data dari ketiga *axis* bisa didapatkan dengan cara melihat *channel* pada ADC internal yang digunakan untuk menyimpan data tegangan masing-masing sumbu (*axis*). Pada modul sensor ini diketahui bahwa data dari masing-masing sumbu X, Y, dan Z terdapat pada *channel* 0, 1, dan 2. Selain ketiga sumbu tersebut pada modul ini juga terdapat data tegangan referensi (Vref), terdapat pada channel 3, yang digunakan sebagai perbandingan oleh masing-masing sumbu untuk mendapatkan nilai G.



Gambar 2 Orientasi arah percepatan *accelerometer*

Dari data sumbu (*axis*) dan Vref tadi bisa didapatkan. Besar nilai G dengan menggunakan persamaan dibawah ini:

$$G = ((axis - Vref) / 4095) \times (3,3 / 0,3663)$$

Dimana nilai *axis* dan Vref didapatkan dari pembacaan data ADC, 4095 merupakan output maksimal dari 12 bit ADC, 3.3 adalah tegangan suplai H48C, dan 0,3663 adalah output

tegangan untuk 1G (ketika beroperasi pada tegangan 3.3V). Persamaan tersebut dapat disederhanakan menjadi:

$$G = (axis - Vref) \times 0.0022$$

2.2 Pedestrian Dead Reckoning (PDR)

Pedestrian navigation merupakan suatu proses untuk mengetahui posisi dan mengikuti jejak pejalan kaki. Dengan mengetahui langkah seseorang setiap waktu dan memahami mekanisme pejalan kaki, dapat mengurangi pertambahan nilai error dalam menentukan posisi seseorang. Dalam Pedestrian Dead Reckoning, sensor percepatan digunakan untuk mendeteksi terjadinya langkah (*step*) kaki, kemudian memperkirakan posisi dalam koordinat X dan Y berdasarkan input panjang langkah kaki dan orientasi arah yang terbaca oleh kompas digital.

Ada tiga parameter penting yang diterapkan dalam pedestrian dead reckoning ini yaitu:

1. Deteksi langkah (*step*) kaki
2. Panjang langkah kaki
3. Orientasi perubahan arah setiap langkah kaki.

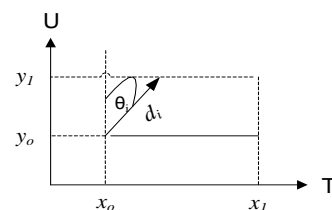
Untuk panjang langkah d_i dengan arah membentuk sudut θ_i seperti pada gambar 5 maka perpindahan ke arah timur, Δx_i adalah

$$\Delta x_i = d_i \sin \theta_i \dots\dots\dots (1)$$

Maka menggunakan cara yang sama, perpindahan dalam arah utara adalah

$$\Delta y_i = d_i \cos \theta_i \dots\dots\dots (2)$$

Dengan θ_i adalah sudut yang terbaca oleh kompas digital, karena pembacaan kompas digital yang digunakan absolut terhadap arah utara bumi atau sumbu Y positif koordinat kartesian.



Gambar 3 Orientasi arah pergerakan dengan membentuk sudut θ_i

Maka untuk sejumlah n langkah yang terdeteksi, posisi akhir dengan titik referensi $[x_0, y_0]$ adalah

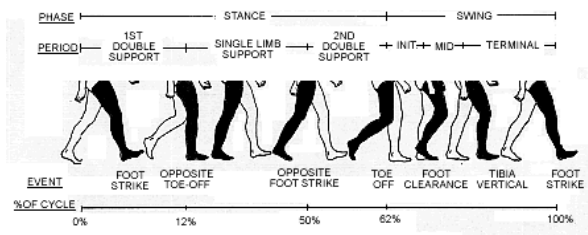
$$x_n = x_0 + \sum_{i=1}^{i=n} d_i \sin \theta_i \dots\dots\dots (3)$$

$$y_n = y_0 + \sum_{i=1}^{i=n} d_i \cos \theta_i \dots\dots\dots (4)$$

Bentuk persamaan tersebut menjadi dasar dari sistem navigasi bagi pejalan kaki menggunakan prinsip dead reckoning.

2.3 Identifikasi Gaya Berjalan

Berdasarkan cara manusia berjalan menunjukkan bahwa pola pergerakan manusia selama berjalan memiliki siklus, berulang, dan konsisten. Hal tersebut dapat diilustrasikan pada Gambar 4. Gaya berjalan manusia terdiri dari dua fase utama yaitu fase berdiri (*stance*), ketika salah satu kaki diatas tanah dan fase berayun (*swing*). Telapak kaki menyentuh tanah selama fase *stance* hingga ujung jari terakhir menyentuh tanah (*toe-off*) terjadi, sehingga fase *swing* dimulai sejak ujung jari kaki meninggalkan tanah. Fase *swing* berakhir ketika telapak kaki menapak tanah kembali dan siap untuk memulai fase *stance* berikutnya.



Gambar 4 Siklus normal orang berjalan

2.4 Deteksi Langkah Kaki

Cara termudah untuk mendeteksi event deteksi langkah kaki dengan cara ini adalah dengan mengetahui nilai percepatan pada setiap fase ataupun kondisi seseorang saat berjalan kemudian membandingkannya dengan nilai *threshold* (Ω) yang telah ditentukan, yang dapat dijelaskan sebagai berikut:

1. *Accelerometer* yang digunakan dalam tugas akhir ini, mengukur nilai percepatan dalam tiga sumbu, yaitu sumbu x, y, dan z, yang masing-masing disimbolkan dalam A_x , A_y , dan A_z . Dari ketiga nilai percepatan tersebut, dapat dicari nilai magnitudenya dengan cara

$$A = \sqrt{(A_x)^2 + (A_y)^2 + (A_z)^2} \dots\dots\dots(1)$$

2. Nilai output magnitude yang ditampilkan oleh *accelerometer* dipengaruhi pula oleh nilai dari percepatan gravitasi, seperti ditunjukkan pada persamaan berikut

$$a_o(t) = a_f(t) + a_g(t) \dots\dots\dots(2)$$

dimana a_o adalah nilai output *accelerometer*, a_f menunjukkan nilai magnitude percepatan kaki, dan a_g nilai dari percepatan gravitasi. Percepatan gravitasi memiliki ketetapan nilai $9,8 \text{ m/s}^2$ ($1g$). Pada saat kaki berada pada fase *stance*, percepatan kaki bernilai nol ($a_f(t)=0$). Sehingga persamaan sebelumnya dapat disederhanakan menjadi

$$a_o(t) = 1g \dots\dots\dots(3)$$

Idealnya, nilai magnitude bernilai sama dengan percepatan gravitasi bila kaki berada pada fase *stance*, sehingga

$$a_o(t) = a_g(t) = A = 1 \dots\dots\dots(4)$$

3. Seseorang mengalami proses melangkah bila terjadi perpindahan fase dari fase *stance* ke fase *swing*. Maka, berdasarkan persamaan (1) hingga persamaan (4), langkah kaki seseorang dapat terdeteksi apabila terjadi perubahan nilai yang tiba tiba dari $A \approx 1$ menjadi:

$$A > 1g$$

2.5 Variance Threshold

Nilai *variance*, s_n^2 dari sejumlah n sampel data dapat diketahui dengan

$$s_n^2 = \frac{1}{n} \sum_{i=j}^{i=j+n} (x_i - \bar{x}_j)^2$$

Dimana, \bar{x}_j adalah nilai rata-rata dari n sampel data

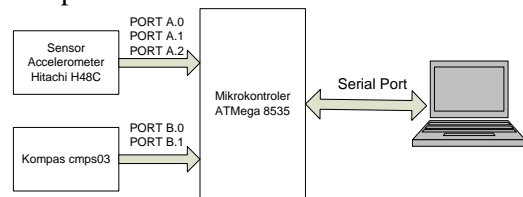
$$\bar{x}_j = \frac{1}{n} \sum_{i=j}^{i=j+n} x_i$$

Dalam penerapan yang tepat, nilai *variance* dari magnitude sangat bermanfaat untuk mendeteksi event terdeteksinya langkah kaki, dengan perubahan nilai *variance* berubah secara tiba-tiba misalnya pada saat tumit terangkat. Dengan perhitungan nilai *variance* yang relatif stabil membuat nilai *drift* atau *bias* dari output sensor menjadi tidak sensitif. Oleh karena itu dengan logika yang sama seperti dijelaskan pada subbab deteksi langkah kaki, pendeteksian langkah kaki dengan menggunakan nilai ambang (*threshold*) *variance* sebagai pembanding menjadi lebih akurat dibandingkan dengan menggunakan nilai ambang magnitude saja.

III. PERANCANGAN ALAT

3.1 Perancangan Perangkat Keras (*Hardware*)

Perancangan perangkat keras sistem navigasi personal pada tugas akhir ini dapat dilihat pada Gambar 5.



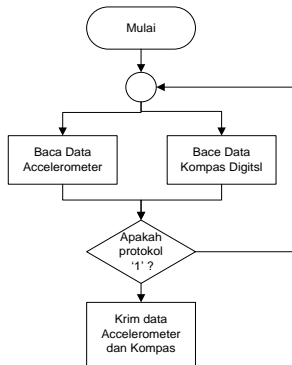
Gambar 5 Diagram blok perancangan perangkat keras.

Penjelasan diagram blok sistem pada Gambar 5 dapat dijelaskan sebagai berikut:

1. Sensor *accelerometer* H48C digunakan untuk mengetahui nilai percepatan kaki saat berjalan kemudian akan dikirim ke Mikrokontroller.
2. Kompas CMPS03 digunakan untuk menentukan arah terhadap medan magnet bumi setiap langkah kaki yang terdeteksi
3. Mikrokontroller ATmega 8535 sebagai otak dari sistem ini yang fungsinya menerima data dari sensor dan mengirim data ke laptop
4. Laptop berfungsi sebagai pengolah data melalui program Visual C# 2008.

3.2 Perancangan Perangkat Lunak Pada Mikrokontroller

Mikrokontroller ATmega 8535 digunakan untuk mengakses sensor dan mengirim data sensor ke laptop. Secara umum diagram alir pemrograman pada mikrokontroller adalah sebagai berikut

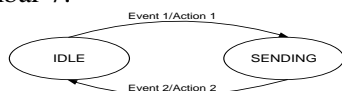


Gambar 6 Diagram alir program utama pada mikrokontroller

Pengiriman data-data sensor dilakukan oleh mikrokontroller melalui komunikasi serial dengan laptop. Teknik komunikasi serial yang digunakan dalam tugas akhir ini adalah master-slave, dimana *master* adalah *device* yang memberikan protokol, sedangkan *slave* adalah *device* yang menerima protokol. Proses komunikasi serial ini adalah *master* mengirimkan data sebagai protokol terlebih dahulu, kemudian data tersebut diterima oleh *slave*, selanjutnya berdasarkan protokol yang diterima, *slave* akan mengirimkan data yang diinginkan *master* atau akan menerima data dari *master*.

3.2.1 Penerimaan Data Serial pada Mikrokontroller

Perancangan penerimaan data serial menggunakan diagram *state* yang diperlihatkan pada Gambar 7.



Gambar 7 Diagram *state* pengiriman data pada mikrokontroller

3.2.2 Pengiriman Data Serial pada Mikrokontroller

Fungsi pengiriman data serial dipanggil setiap 45 ms sekali dengan menggunakan interrupt timer1. Register yang digunakan untuk menset nilai timer adalah register TCNT, register TCNT sendiri dibagi menjadi dua yaitu TCNT 1H dan TCNT 1L. Rumus yang digunakan adalah:

$$TCNT = (1 + 0xFFFF) - (\text{waktu} * (XTAL / \text{prescaler}))$$

dimana:

waktu : waktu yang diinginkan

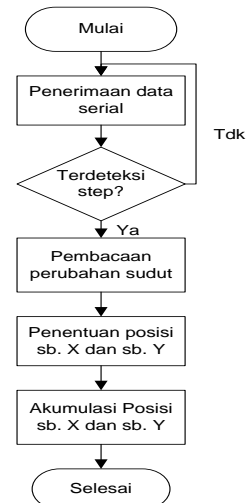
XTAL : frekuensi xtal yang digunakan

prescaler : nilai prescaler

Nilai dari TCNT yang didapat dari rumus bernilai 16 bit (4 angka heksadesimal), 2 angka di depan dimasukkan ke TCNT1H dan 2 angka yang di belakang dimasukkan ke TCNT1L.

3.3 Perancangan Perangkat Lunak pada Laptop

Tampilan program pada laptop dibuat menggunakan bahasa pemrograman Visual C# 2008. Tugas-tugas pada program ditangani oleh fungsi-fungsi *event handler*. Fungsi-fungsi tersebut dipanggil jika ada *event* tertentu saat program dijalankan, sehingga semua program berjalan jika ada *event* tertentu yang dijadwalkan melalui Timer.

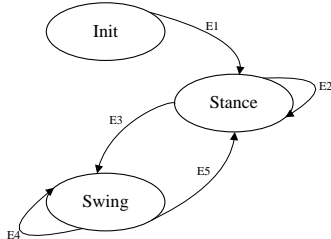


Gambar 8 Flowchart program utama pada laptop

3.3.1 Perancangan Program Pendeteksi Langkah Kaki

Pendeteksian langkah kaki yang digunakan adalah dengan menentukan nilai ambang (*threshold*) dari *variance* magnitudenya. Nilai *threshold variance* ini kemudian dikondisikan ke dalam nilai nol (0) dan satu (1). Logika 0 dikondisikan untuk kaki yang

mengalami fase *stance* sedangkan logika 1 untuk fase *swing*. Langkah kaki akan terdeteksi apabila ada perubahan dari logika satu ke logika nol dengan lima sampel data pada fase *stance* berjumlah lima. Pencuplikan lima sampel data dilakukan dengan menerapkan metode *moving variance*. Berikut ini adalah perancangan program pendeteksi langkah kaki yang digambarkan dalam diagram state.



Gambar 9 Diagram state pendeteksi langkah kaki

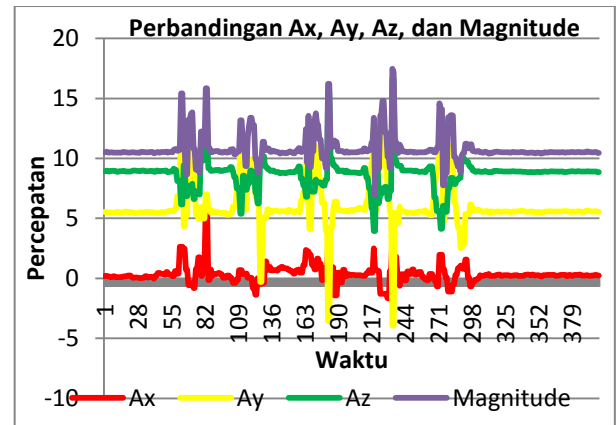
Dalam perancangan sistem ini, nilai *variance* akan berlogika 0 atau berada pada fase *stance* jika nilai varians yang sesungguhnya (*var*) bernilai kurang dari atau sama dengan 0.02, sedangkan akan berlogika 1 atau berada pada fase *swing* bila nilai varians yang sesungguhnya tidak sama dengan 0.02. Nilai ambang tersebut ditentukan melalui pengamatan beberapa data percobaan saat kaki melangkah.

3.3.2 Perancangan Program Posisi dan Orientasi Arah Langkah Kaki

Program posisi dan orientasi arah langkah kaki akan mulai bekerja apabila terdeteksi ada step. Perubahan orientasi arah langkah kaki diketahui melalui pembacaan nilai kompas digital. Nilai tersebut akan dianggap sebagai nilai θ_i karena pembacaan kompas digital yang digunakan absolut terhadap arah utara bumi atau sumbu Y positif koordinat kartesian. Kemudian dari nilai orientasi arah terhadap sumbu Y positif koordinat kartesian maka akan dapat diketahui posisi dalam arah sumbu X dan sumbu Y sesuai dengan rumus (1) – (4).

IV. PENGUJIAN DAN ANALISIS

4.1 Pengujian Output Sensor Accelerometer saat Kaki Melangkah



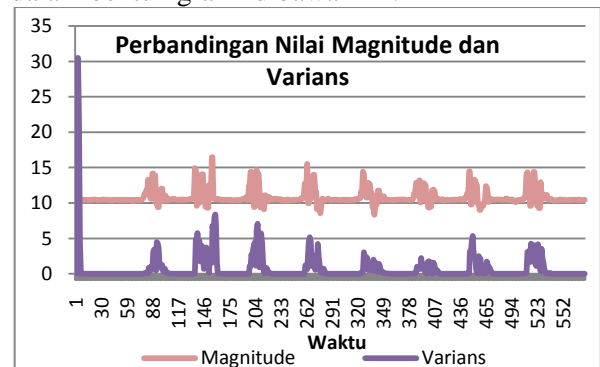
Gambar 10 Grafik output accelerometer

Gambar diatas menunjukkan nilai keluaran *accelerometer* saat kaki melangkah. Fase *stance* ditunjukkan saat grafik berada pada nilai tertentu yang relatif stabil sehingga hampir membentuk garis horizontal, kemudian fase *swing* ditunjukkan saat grafik membentuk gelombang-gelombang.

Dari gambar terlihat bahwa percepatan dalam arah sumbu Y dan Z memiliki tingkat sensitifitas yang lebih tinggi dibandingkan dengan sumbu X. Hal tersebut terlihat dari kemampuan sumbu tersebut dalam membedakan fase *stance* dan *swing* dengan jelas. Semua sumbu tersebut memiliki nilai berubah-ubah saat fase *swing*. Oleh karena itu, nilai magnitude digunakan dalam mendeteksi langkah kaki karena saat kaki tidak bergerak atau sedang berada pada fase *stance*, berapapun nilai percepatan setiap sumbu nilai magnitudenya pasti mendekati $1g \approx 10 \text{ m/s}^2$. Hal ini bertujuan agar memudahkan dalam memahami algoritma pendeteksi langkah kaki yang digunakan.

4.2 Pengujian Jumlah Langkah Kaki yang Terdeteksi

Pengujian ini dilakukan dalam keadaan berjalan dengan sistem terpasang di punggung kaki, kemudian data-data sensor diproses untuk mendeteksi langkah kaki yang terjadi. Berikut ini adalah hasil data pengujian yang ditampilkan dalam bentuk grafik dibawah ini:



Gambar 11 Nilai magnitude dan varians untuk 8 step

Seperti terlihat pada gambar diatas, dengan menggunakan nilai magnitudo saja sebenarnya sudah dapat terlihat perbedaan saat fase *stance* dan fase *swing* sehingga jumlah langkah kaki dapat terdeteksi. Akan tetapi untuk lebih memudahkan dalam memahami algoritma pendeteksian langkah kaki dan kemudahan dalam *programming*, maka dilakukan normalisasi terhadap nilai magnitudo dengan cara mencari nilai variannya. Sehingga besaran nilai magnitudo tidak perlu dipedulikan karena dengan penggunaan variansi nilai-nilai tersebut akan otomatis dikondisikan pada dua keadaan, yaitu keadaan yang mendekati nilai nol (fase *stance*) dan nilai yang lebih dari nol (fase *swing*). Oleh karena itu, penggunaan metode *variance threshold* menjadi metode yang tepat dan mudah digunakan untuk mendeteksi jumlah langkah kaki.

Berdasarkan hasil pengamatan data dari beberapa percobaan dapat disimpulkan:

Varians $\leq 0,02 \longrightarrow$ Fase *Stance*

Varians $> 0,02 \longrightarrow$ Fase *Swing*

4.3 Pengujian Posisi dan Orientasi Perubahan Arah

Pengujian posisi dan orientasi perubahan arah ini dilakukan dalam dua lintasan yaitu lintasan yang lurus dan lintasan berbentuk persegi.

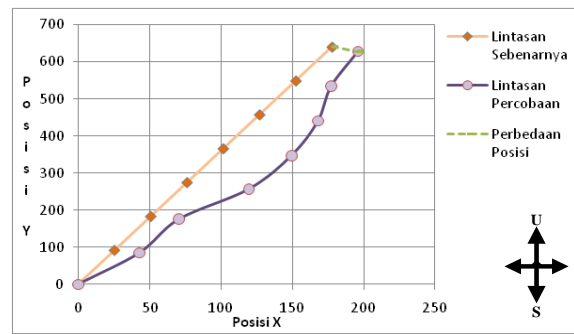
4.3.1 Lintasan Lurus

Pada percobaan ini orientasi arah pejalan kaki saat sebelum melangkah adalah $15,529^\circ$. Berikut ini adalah data yang didapatkan dari hasil percobaan kemudian dibandingkan dengan data perhitungan yang seharusnya:

Tabel 1 Perbandingan orientasi arah dan posisi antara perhitungan dan percobaan lintasan lurus

Step	Sdt_Perc (°)	Posisi Sb. X (cm)		Posisi Sb. Y (cm)	
		Seharusnya	Perc	Seharusnya	Perc
1	26.82	25.433	42.868	91.532	84.778
2	16.94	50.866	70.55	183.06	175.65
3	31.06	76.299	119.56	274.59	257.04
4	18.35	101.73	149.48	366.13	347.20
5	11.29	127.16	168.08	457.66	440.36
6	5.65	152.59	177.43	549.19	534.90
7	11.29	178.03	196.03	640.72	628.06

Berdasarkan data diatas maka dapat dibuat pemetaan dalam suatu koordinat kartesian seperti ditunjukkan gambar berikut ini:



Gambar 12 Pemetaan pada lintasan lurus

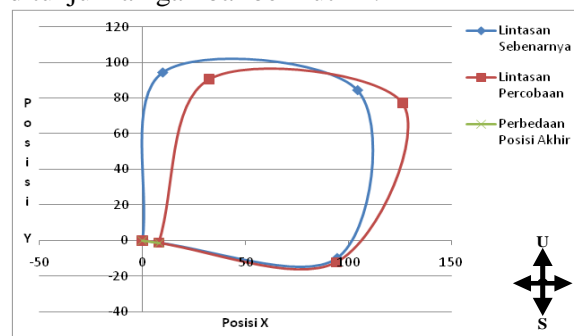
4.3.2 Lintasan Persegi

Pada percobaan ini orientasi arah pejalan kaki saat sebelum melangkah adalah 96° . Berikut ini adalah data yang didapatkan dari hasil percobaan kemudian dibandingkan dengan data perhitungan yang seharusnya:

Tabel 2 Perbandingan orientasi arah dan posisi antara perhitungan dan percobaan lintasan persegi

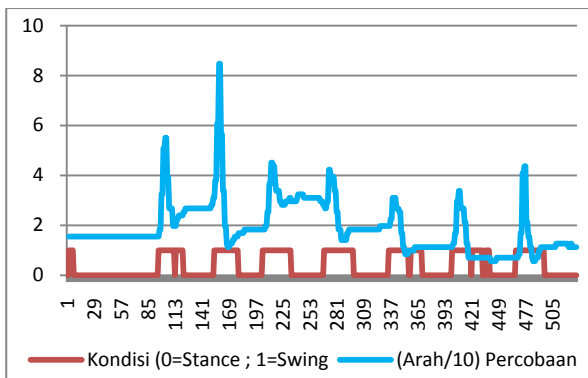
Step	Sdt_seharusnya (°)	Sdt_Perc (°)	Posisi Sb. X (cm)		Posisi Sb. Y (cm)	
			Seharusnya	Perc	Seharusnya	Perc
2	6	19.76	104.4	126.3	84.55	77.15
3	276	278.1	9.93	32.28	94.48	90.56
4	186	194.8	0	7.977	0	-1.27

Berdasarkan data diatas maka dapat dibuat pemetaan dalam suatu koordinat kartesian seperti ditunjukkan gambar berikut ini:



Gambar 13 Pemetaan lintasan persegi 95cm x 95cm

Dari kedua pengujian diatas dapat disimpulkan bahwa sistem ini sudah dapat mengikuti bentuk lintasan yang seharusnya walaupun tidak tepat sesuai dengan koordinat yang seharusnya. Penyebab utama hal tersebut adalah kemampuan kompas dalam membaca orientasi arah seperti dapat diilustrasikan oleh gambar berikut ini:



Gambar 14 Pembacaan data kompas saat terjadi fase *stance* dan *swing*

Gambar 14 menunjukkan pembacaan data kompas untuk lintasan yang lurus. Seharusnya ketika kompas membaca orientasi arah pejalan kaki pada fase *stance* nilai yang terbaca tidak berbeda jauh dengan pembacaan orientasi arah saat sebelum melangkah. Hal ini bisa terjadi karena saat seseorang berjalan terjadi pergerakan yang dinamis dalam waktu yang cepat tetapi tidak diimbangi dengan waktu komputasi kompas yang cepat pula. Akibatnya, kompas belum berada pada kondisi yang *steady* tetapi sudah harus mengirim data orientasi arah. Selain itu, pembacaan data kompas ini juga dipengaruhi oleh *tilt* yang mengakibatkan pembacaan data kompas terjadi saat keadaan kaki tidak benar-benar dalam keadaan yang diam dan rata terhadap tanah.

V. PENUTUP

5.1 Kesimpulan

Berdasarkan pengujian dan analisis yang telah dilakukan, maka dapat disimpulkan beberapa hal sebagai berikut:

1. Jumlah langkah dapat terdeteksi dengan menggunakan metode *variance threshold*, dengan ketentuan nilai varians $\leq 0,02$ untuk fase *stance* dan varians $> 0,02$ untuk fase *swing*.
2. Algoritma pendeteksian langkah yang digunakan sudah dapat mendeteksi jumlah langkah sesuai keadaan yang sebenarnya pada kondisi kecepatan normal.
3. Pembacaan data kompas dalam keadaan statis dan tidak ada gangguan dari luar sudah menunjukkan nilai pembacaan arah yang benar. Akan tetapi, bila kompas mengalami pergerakan yang dinamis pembacaannya masih kurang tepat. Hal ini dikarenakan pergerakan terjadi dalam waktu yang cepat tetapi tidak diimbangi dengan waktu komputasi kompas yang cepat pula. Akibatnya, kompas belum berada pada

kondisi yang *steady* tetapi sudah harus mengirim data orientasi arah.

4. Hasil pemetaan pada lintasan lurus dan persegi sudah bisa mengikuti pola yang sebenarnya walaupun tidak tepat pada posisi koordinat yang sebenarnya

5.2 Saran

Untuk pengembangan sistem lebih lanjut, maka ada beberapa saran yang dapat dilakukan yaitu sebagai berikut:

1. Kompas digital yang digunakan dapat diganti dengan kompas lain yang memiliki waktu komputasi yang jauh lebih cepat daripada yang digunakan dalam tugas akhir ini.
2. Pengolahan data-data dari *accelerometer* dan kompas dilakukan dalam mikrokontroler, sehingga data yang dikirim ke laptop sudah berupa data posisi.
3. Untuk fleksibilitas dalam penggunaan terutama saat berjalan, maka sistem komunikasi data kabel serial dapat diganti dengan komunikasi *wireless*.
4. Tampilan visual pada C# dapat dikembangkan dengan menampilkan pemetaan posisi X dan posisi Y secara *real time*.

DAFTAR PUSTAKA

- [1] Stirling, Ross Grote, *Development of Pedestrian Navigation System Using Shoe Mounted*, Thesis Teknik Mesin Universitas Alberta, Kanada, 2004.
- [2] Abadi, Muslim, *Rancang Bangun Alat Pengukur Langkah Kaki dengan Sensor Accelerometer dan Fasilitas Wireless 2,4 GHz*, Proyek Akhir Teknik Telekomunikasi Politeknik Negeri Surabaya Institut Teknologi Sepuluh November, Surabaya, 2010.
- [3] Ojeda, Lauro dan Johann Borenstein., *Non-GPS Navigation for Security Personnel and First Responders*, Journal of Navigation, University of Michigan, USA, 2007
- [4] Joni, I Made dan Budi Raharjo., *Pemrograman C dan Implementasinya*, Penerbit Informatika, Bandung, 2006.
- [5] Hartanto, Budi, *Memahami Visual C# Secara Mudah*, Penerbit Andi, Yogyakarta, 2008.
- [6] Andrejasic, Matej, *MEMS Accelerometer*, Faculty for Mathematics and Physics Department of Physics, University of Ljubljana, 2008.

- [7] Soebhakti, Hendawan, *Digital Compass CMPS03*, <http://www.hendawan.files.wordpress.com/2009/.../digital-compass-application3.pdf>, Desember 2008
- [8] -----, *Atmega8535 Data Sheet*, <http://www.alldatasheet.com/datasheet-pdf/pdf/164169/ATMEL/ATMEGA8535.html>
- [9] -----, Komunikasi USART AVR, <http://www.PayZtronics.blogspot.com>, Agustus 2009.
- [10] -----, *H48C Data Sheet – Parallax Home*, <http://www.parallax.com/dl/docs/prod/acc/H48CDatasheet.pdf>, Juli 2007.
- [11] -----, Pengenalan C#, <http://www.openit.net>, Januari 2010.
- [12] -----, Rata-rata Bergerak Tunggal (Single Moving Average), <http://exponensial.wordpress.com/2010/10/26/rata-rata-bergerak-tunggal-single-moving-average/>
- [13] -----, Bermain dengan Timer 1 AVR, <http://www.avrku.com/2009/02/bermain-dengan-timer-1-avr.html>
- [14] -----, Mikroteknologi, <http://id.wikipedia.org/wiki/Mikroteknologi>



Laras Dwi Kawuri (L2F 006 061) Lahir di Jakarta, 14 November 1988. Mahasiswa Teknik Elektro Reguler 2006, Konsentrasi Kontrol dan Automatik, Universitas Diponegoro.

Mengetahui dan mengesahkan,

Dosen Pembimbing I

Dosen Pembimbing II

Iwan Setiawan, S.T.,M.T.

Darjat, S.T.,M.T.

NIP.197309262000121001

NIP.197206061999031001

Tanggal: _____

Tanggal: _____